Fusion pour la séparation de sources audio

Xabier Jaureguiberry

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUT OF TECHNOLOGY

TELECOM ParisTech

2015-ENST-0030

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal & Images »

présentée et soutenue publiquement par

Xabier JAUREGUIBERRY

le 16 juin 2015

Fusion pour la séparation de sources audio

Fusion for audio source separation

Directeur de thèse: **Gaël RICHARD**
Co-encadrement de la thèse: **Emmanuel VINCENT**

Jury

M. Laurent GIRIN, Professeur, GIPSA-lab, Grenoble-INP	Rapporteur
M. Jérôme IDIER, Directeur de Recherche, IRCCyN, École Centrale de Nantes	Rapporteur
M. Jean-Luc ZARADER, Professeur, ISIR, Université Pierre et Marie Curie	Président
M. Pierre LEVEAU, Docteur, Directeur de la recherche, Audionamix	Examinateur
M. Jonathan LE ROUX, Docteur, Mitsubishi Electric Research Laboratories	Examinateur

TELECOM ParisTech
école de l'Institut Mines-Télécom - membre de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

T H È S E

Résumé

La séparation aveugle de sources audio dans le cas sous-déterminé est un problème mathématique complexe dont il est aujourd'hui possible d'obtenir une solution satisfaisante pour certaines applications industrielles, à condition de sélectionner la méthode la plus adaptée au problème posé et de savoir paramétrer celle-ci soigneusement. Afin d'automatiser cette étape de sélection déterminante, nous proposons dans cette thèse de recourir au principe de fusion, très populaire dans le domaine de la classification mais encore peu exploité en séparation de sources. L'idée est simple : il s'agit, pour un problème donné, de sélectionner plusieurs méthodes de résolution plutôt qu'une seule et de les combiner afin d'en améliorer la solution.

Pour cela, nous introduisons un cadre général de fusion qui consiste à formuler l'estimée d'une source comme la combinaison de plusieurs estimées de cette même source données par différents algorithmes de séparation, chaque estimée étant pondérée par un coefficient de fusion. Ces coefficients peuvent notamment être appris sur un ensemble d'apprentissage représentatif du problème posé par minimisation d'une fonction de coût liée à l'objectif de séparation. Pour aller plus loin, nous proposons également deux approches permettant d'adapter les coefficients de fusion au signal à séparer. La première formule la fusion de modèles de factorisation en matrices non-négatives (NMF) dans un cadre bayésien, à la manière du moyennage bayésien de modèles. La deuxième exploite la puissance d'apprentissage des réseaux de neurones profonds afin de déterminer des coefficients de fusion variant en temps.

Toutes ces approches ont été évaluées sur deux corpus distincts : l'un dédié au rehaussement de la parole, l'autre dédié à l'extraction de voix chantée. Quelle que soit l'approche considérée, nos résultats montrent l'intérêt systématique de la fusion par rapport à la simple sélection, la fusion adaptative par réseau de neurones se révélant être la plus performante.

Abstract

Underdetermined blind source separation is a complex mathematical problem that can be satisfyingly resolved for some practical applications, providing that the right separation method has been selected and carefully tuned. In order to automate this selection process, we propose in this thesis to resort to the principle of fusion which has been widely used in the related field of classification yet is still marginally exploited in source separation. Fusion consists in combining several methods to solve a given problem instead of selecting a unique one.

To do so, we introduce a general fusion framework in which a source estimate is expressed as a linear combination of estimates of this same source given by different separation algorithms, each source estimate being weighted by a fusion coefficient. For a given task, fusion coefficients can then be learned on a representative training dataset by minimizing a cost function related to the separation objective. To go further, we also propose two ways to adapt the fusion coefficients to the mixture to be separated. The first one expresses the fusion of several non-negative matrix factorization (NMF) models in a Bayesian fashion similar to Bayesian model averaging. The second one aims at learning time-varying fusion coefficients thanks to deep neural networks.

All proposed methods have been evaluated on two distinct corpora. The first one is dedicated to speech enhancement while the other deals with singing voice extraction. Experimental results show that fusion always outperform simple selection in all considered cases, best results being obtained by adaptive time-varying fusion with neural networks.

Table des matières

Chapitre 1

Introduction

Sommaire

Les applications audio du traitement de signal ont le pouvoir de fasciner le grand public. J'en tiens pour preuve les nombreuses fois où l'on m'a demandé de raconter mon travail. Si l'évocation du moindre terme mathématique tend à effrayer et désintéresser instantanément le curieux, il semble plus aisé de trouver des exemples concrets d'applications qui touchent ce dernier, voire le passionnent. Dans ces situations, l'un des exemples dont j'ai le plus usé pour expliquer mon travail et susciter l'intérêt est incarné par le logiciel de reconnaissance de musique *Shazam*[1] [Wang, 2006]. En effet, rares sont ceux n'ayant pas entendu parlé de cet outil bien pratique et les avis sont généralement très élogieux quant à ses performances. Développé à l'origine pour les téléphones intelligents, son objectif est simple : il permet d'identifier un titre musical diffusé, par exemple, à la radio grâce à l'enregistrement d'un court extrait par le biais du microphone du téléphone. Si l'idée sous-jacente paraît évidente, le défi technique relevé pour rendre cette technologie fiable et robuste n'en est pas moins remarquable. Pour l'utilisateur non averti, la technologie semble relever de procédés magiques mais pour le chercheur aguerri, elle reflète le fruit d'années de travail dans le domaine de la recherche d'information musicale (*MIR* pour *Music Information Retrieval* en anglais) aboutissant finalement au lancement d'un produit grand public aujourd'hui très populaire.

Dans le domaine connexe de la séparation de sources, qui forme le cœur de notre étude, nombreuses sont les applications qui pourraient susciter un engouement comparable à celui rencontré par *Shazam*. Par exemple, il pourrait un jour devenir possible de générer automatiquement un accompagnement de karaoké à partir de n'importe quel morceau de musique. Pour les musiciens, la séparation de sources pourrait aider à l'édition automatique de partitions de musique, à partir d'un enregistrement studio ou même *à la volée* pour la transcription de musique *live*. Dans le domaine médical, elle pourrait participer également à l'amélioration des prothèses auditives qui ont pour défaut bien connu d'amplifier tous les sons de façon équivalente, alors qu'il serait plutôt souhaitable de ne rehausser que les sons utiles, et en premier lieu la voix. Les applications envisageables n'ont de limite que notre imagination.

En revanche, pour que ces applications fantasmées deviennent un jour réalité, la recherche en séparation de sources doit encore progresser, et en particulier, il devient souhaitable d'automatiser les procédés de séparation de sources. En effet, la littérature dans le domaine présente tant de diversité dans les méthodes de séparation proposées que, pour résoudre un problème posé en pratique, il est fort probable qu'une ou plusieurs méthodes déjà existantes puissent être employées pour réaliser la tâche considérée, au prix toutefois d'un effort potentiellement important d'adaptation. C'est ainsi qu'a été initialement pensé le sujet de cette thèse.

1.1 Contexte : sélection automatique de modèles pour la séparation de sources

J'ai débuté cette thèse alors que je travaillais encore pour une petite entreprise innovante française nommée *Audionamix*[2]. À l'époque, l'entreprise avait pour principal objectif de fournir à des acteurs majeurs des industries du cinéma, de la télévision et de la musique des services de séparation de sources. Pour ce faire, l'entreprise disposait de plusieurs méthodes de séparation et modèles de sources. En réponse à une demande d'un client, le travail de séparation se décomposait naturellement en trois étapes. Dans un premier temps, il s'agissait de prendre connaissance du problème posé ainsi que des données et informations contextuelles fournies par le client afin de choisir les modèles de séparation adéquats. Une fois les modèles sélectionnés, la deuxième étape consistait à paramétrer ces modèles pour les adapter au mieux au problème posé. En effet, la plupart des modèles utilisés possédaient un certain nombre de degrés de liberté, appelés *hyperparamètres*,

1. http://www.shazam.com/
2. http://www.audionamix.com/

permettant de configurer le modèle en fonction de l'objectif convoité. Enfin, une fois la séparation effectuée par les modèles choisis et paramétrés, une qualité de séparation satisfaisante n'était souvent obtenue qu'au prix d'une ultime étape de nettoyage à la main, souvent coûteuse en temps malgré l'efficacité des procédés de séparation employés. Bien que des travaux aient été déjà menés afin que ces trois étapes puissent être réalisées par des ingénieurs du son qualifiés, il n'était pas rare que les demandes des clients soient en partie traitées par l'équipe de recherche, seule à même de régler très finement les modèles de séparation. Pour y remédier, il m'a été proposé d'étudier des moyens permettant de sélectionner et paramétrer automatiquement les modèles de séparation en fonction de la tâche considérée.

D'un point de vue théorique, ce problème a été très largement étudié dans la littérature sous le nom de *sélection de modèles*. Comme son nom l'indique, la sélection de modèles concerne l'étude des moyens permettant de sélectionner le modèle le plus adapté à la résolution d'un problème donné. Mes premiers travaux de thèse ont donc été menés dans cette direction. Plus précisément, nous avions retenu un des modèles utilisés régulièrement par les ingénieurs du son (la factorisation en matrices non-négatives, que nous introduirons dès le prochain chapitre) et identifié l'ensemble des hyperparamètres ayant une influence non-négligeable sur la qualité de séparation. Étant donnée une tâche de séparation, notre objectif devenait alors de déterminer la meilleure combinaison de valeurs de ces hyperparamètres. Pour ce faire, avec mes encadrants et collègues de l'époque, nous avions envisagé de recourir aux métaheuristiques [Dréo et al., 2003], et notamment aux algorithmes génétiques. Ces algorithmes d'optimisation ont la particularité d'être capables de parcourir des espaces de très grande dimension, tels que ceux formés par l'ensemble des valeurs possibles des hyperparamètres étudiés. Toutefois, cet axe de recherche a été rapidement abandonné, pour deux raisons principales. En premier lieu, les tests préliminaires que nous avons effectués n'ont montré qu'une très faible amélioration de la séparation (quelques centièmes de décibels seulement) au prix de plus d'un mois de calculs intensifs, amenuisant de ce fait l'espoir qu'une configuration optimale d'un modèle puisse être ainsi découverte. Par ailleurs, notre vision du problème posé s'est vue modifiée par le simple questionnement suivant : pourquoi choisir ?

En effet, pourquoi ne retenir qu'un modèle pour résoudre un problème donné alors que nous disposons potentiellement de plusieurs modèles distincts permettant de résoudre ce même problème ? Et pour un modèle donné, pourquoi ne retenir qu'un unique paramétrage alors que différentes combinaisons d'hyperparamètres peuvent éventuellement donner des performances équivalentes ? Ces questions trouvent également une réponse générale dans la littérature sous le nom de *fusion*. La fusion peut être vue comme une extension du principe de sélection de modèles où, au lieu de ne sélectionner qu'un modèle pour résoudre un problème, il est proposé d'en retenir plusieurs et de les combiner. La fusion s'attache alors à l'étude des moyens permettant de combiner plusieurs modèles. Si en séparation de sources ce principe semble peu exploité, il a été à l'origine de contributions majeures dans des domaines connexes tels que la classification ou la reconnaissance de parole. À ce titre, il nous a semblé judicieux de réorienter nos recherches dans cette voie et de privilégier cet axe-là dans ce mémoire.

1.2 Contributions : fusion de modèles pour la séparation de sources

Bien que nous ayons à l'origine considéré le problème de sélection de modèles comme une solution à part entière pour automatiser les procédures de séparation de sources, nous avons donc finalement privilégié d'étendre notre étude au principe plus général de fusion. Nous proposons ci-après un bref résumé des contributions en ce sens, contributions qui seront ensuite détaillées dans les prochains chapitres. Notons dès à présent que toutes ces contributions feront ici l'objet d'une évaluation sur deux corpus distincts, l'un concernant une tâche de rehaussement de la parole,

l'autre concernant une tâche d'extraction de voix chantée. Nous présenterons ces corpus dans le chapitre 3.

Contribution 1 : cadre général de fusion pour la séparation de sources Notre principale contribution est un cadre de fusion dédié à la séparation de sources. Nous avons voulu ce cadre le plus général possible afin de pouvoir exploiter toute la diversité des approches de séparation proposées dans la littérature. De ce fait, notre cadre de fusion, qui sera présenté en détail dans le chapitre 3, ne fait que très peu d'hypothèses sur la nature du problème de séparation considéré et les méthodes de séparation à fusionner. Concrètement, le cadre de fusion proposé revient à combiner linéairement plusieurs estimées d'une même source, chaque estimée étant pondérée par un coefficient de fusion. En pratique, le problème revient donc à estimer ces coefficients de fusion.

Le cadre proposé a fait l'objet d'une première publication dans une version simplifiée au sein des actes d'une conférence internationale [Jaureguiberry et al., 2013]. Une version plus complète a également été soumise pour publication dans une revue internationale [Jaureguiberry et al., 2015].

À partir de ce cadre théorique de fusion, nous avons pu dériver plusieurs cas particuliers de fusion dont nous avons mesuré le potentiel dans un cadre oracle. Les contributions suivantes concernent la mise en pratique de ces cas particuliers.

Contribution 2 : fusion statique Le premier cas de fusion étudié consiste à déterminer les coefficients de fusion indépendamment du signal à séparer, d'où le terme de *fusion statique*. La fusion statique peut être constante sur tout le signal (*fusion statique invariante*) ou constante par bande de fréquences (*fusion statique variant en fréquence*). Plusieurs approches ont été proposées selon que les coefficients de fusion statique sont obtenus avec ou sans étape d'apprentissage. Quel que soit le cas, nous avons montré que la fusion statique était une alternative intéressante à la simple sélection.

La fusion statique sera l'objet du chapitre 4 de ce mémoire. Les premiers résultats ont été publiés dans [Jaureguiberry et al., 2013] et étendus dans [Jaureguiberry et al., 2015].

Contribution 3 : fusion adaptative bayésienne Afin d'adapter les coefficients de fusion au signal à séparer (*fusion adaptative*), nous avons proposé une inteprétation bayésienne de notre cadre de fusion pour le cas particulier de la fusion de modèles de factorisation en matrices non-négatives (*NMF*). Ces modèles, présentés en détail dans le chapitre 2, sont particulièrement sensibles au choix de leur ordre, aussi appelé *nombre de composantes*. Notre étude, qui sera l'objet du chapitre 5, a donc porté sur la fusion de modèles de NMF de différents ordres. Cette contribution peut être détaillée en trois sous-contributions.

Contribution 3.1 : fusion par moyennage bayésien de modèles En respectant le cadre général de fusion proposé, la formulation bayésienne de la fusion adaptative se trouve être équivalente au principe de moyennage bayésien de modèles. Nous avons donc étudié en pratique l'application du moyennage bayésien de modèles à des modèles de NMF d'ordres différents et montré qu'en théorie le coefficient de fusion attaché à un modèle pouvait être interprété comme la probabilité *a posteriori* de ce modèle. Notre étude a notamment fait l'objet d'une publication dans les actes d'une conférence internationale [Jaureguiberry et al., 2014b]. Nous avons ainsi montré que, tel quel, le moyennage bayésien de modèles était inopérant et revenait à effectuer une sélection plutôt qu'une fusion, l'un des modèles ayant une probabilité *a posteriori* égale à un et tous les autres ayant une probabilité *a posteriori* nulle.

Contribution 3.2 : contrôle de l'entropie de la distribution *a posteriori* des modèles Afin de rendre effective la fusion par moyennage bayésien de modèles, nous avons proposé l'introduction d'un paramètre de contrôle de l'entropie de la distribution *a posteriori* des modèles. Grâce à ce paramètre, la fusion par moyennage bayésien devient effective. L'influence de ce paramètre a d'abord été validée par des expériences sur des données synthétiques, dont les résultats ont été publiés dans les actes d'une conférence internationale [Jaureguiberry et al., 2014b], puis par des expériences sur des données réelles, dont les résultats ont également été publiés dans les actes d'une conférence internationale [Jaureguiberry et al., 2014a]. Nous avons ainsi montré que la fusion bayésienne obtenait de meilleurs résultats que la simple sélection bayésienne. Ces résultats sont toutefois à nuancer au vu des performances obtenues par des approches plus simples telles que les approches par fusion statique (contribution 2).

Contribution 3.3 : modèle génératif de NMF à ordre multiple Sur la base de notre étude du moyennage bayésien de modèles de NMF, nous avons proposé un modèle génératif pour la NMF modélisant son ordre au moyen d'une variable aléatoire. Nous avons montré que ce modèle permettait d'estimer plusieurs NMFs d'ordres différents conjointement et de les fusionner à la manière du moyennage bayésien de modèles. Les performances ainsi obtenues sont comparables au cas de la fusion par moyennage bayésien de modèles, pour un gain de temps de calcul très important. Nous avons montré que l'introduction du paramètre de contrôle de l'entropie de la distribution *a posteriori* des nombres de composantes permettait, dans ce cas également, de rendre effective la fusion des NMFs. Ce modèle génératif a été présenté dans notre publication [Jaureguiberry et al., 2014a].

Contribution 4 : fusion adaptative variant en temps par réseaux de neurones Nous avons enfin étudié la possibilité d'adapter, en pratique, les coefficients de fusion au niveau de chaque trame temporelle du signal. Ce cas de fusion, nommé *fusion adaptative variant en temps*, peut être vu comme un problème de régression. Nous suggérons sa résolution par l'emploi de réseaux de neurones profonds. Moyennant la constitution d'un ensemble d'apprentissage représentatif de la tâche de séparation à réaliser, nous montrons que cette approche de fusion est celle menant aux meilleures performances. Nous montrons également que les fonctions de coût usuellement utilisées pour l'apprentissage peuvent être remplacées par des fonctions liées à l'objectif de séparation de sources afin d'améliorer la qualité de séparation. Cette proposition constitue le cœur de l'article soumis pour publication dans une revue internationale [Jaureguiberry et al., 2015]. Nous introduirons également ces travaux dans le chapitre 6 de ce mémoire.

1.3 Structure du document

Ce mémoire est organisé en plusieurs chapitres. Dans un premier temps, nous introduirons dans le chapitre 2 l'état de l'art de la séparation de sources, de la sélection et de la fusion de modèles. Ce chapitre permettra d'introduire notre travail, de le situer par rapport à la littérature et de motiver les approches de fusion que nous proposerons dans les chapitres suivants.

Les chapitres 3, 4, 5 et 6 seront consacrés à la description des travaux que nous avons menés durant cette thèse, chaque chapitre étant en lien direct avec l'une des contributions listées ci-dessus. En particulier, dans le chapitre 3, nous introduirons le cadre général de fusion que nous avons mis au point (contribution 1), les corpus que nous avons exploités et nous évaluerons également les performances potentielles de notre cadre de fusion sur ces corpus. Les chapitres suivants seront dédiés à l'étude pratique de différents cas particuliers de notre cadre de fusion. Dans le chapitre 4, nous étudierons des moyens simples pour déterminer des coefficients de fusion statique, avec ou sans apprentissage (contribution 2). Le chapitre 5 sera consacré à l'étude de la fusion adaptative

bayésienne de NMFs (contribution 3). Dans le chapitre 6, nous détaillerons notre proposition de fusion adaptative variant en temps basée sur l'exploitation de réseaux de neurones (contribution 4). Enfin, nous conclurons nos travaux et donnerons quelques perspectives visant à les prolonger dans le chapitre 7.

Chapitre 2

État de l'art

Sommaire

Nous proposons dans ce chapitre d'introduire notre travail en dressant un état de l'art des techniques de notre champ de recherche. Notre travail se situe à la croisée de deux grands domaines de recherche voisins mais ayant chacun une communauté active de chercheurs relativement distincte. Ainsi, les états de l'art relatifs à ces deux domaines seront traités séparément.

Nous proposerons dans la partie 2.1 de présenter le problème de séparation de sources ainsi que les principales solutions proposées dans la littérature. La partie 2.2 sera plus particulièrement dédiée aux deux cas particuliers de séparation de sources que nous exploiterons tout au long de ce manuscrit afin d'évaluer nos méthodes de fusion. Dans la partie 2.3, nous dresserons ensuite un rapide état de l'art des méthodes de sélection et de fusion de modèles, d'un point de vue théorique d'abord, puis dans l'optique de notre objectif de séparation de sources. Le contenu de ce chapitre formera donc la base de nos travaux présentés dans les chapitres suivants, travaux qui ont pour but d'appliquer les grands principes de fusion introduits dans la partie 2.3 aux méthodes de séparation de sources présentées dans la partie 2.1 et qui seront évalués sur les applications spécifiques décrites dans la partie 2.2.

2.1 Séparation de sources sous-déterminée

La séparation de sources a pour objectif d'estimer divers signaux à partir de l'observation de leur mélange. Le mélange, que nous noterons $\mathbf{x}(t)$ tout au long de ce manuscrit, est généralement représenté sous la forme d'un signal temporel à plusieurs canaux (ou *multicanal*). Ainsi, il est couramment représenté comme une fonction vectorielle d'une variable de temps t telle que

$$\forall t, \quad \mathbf{x}(t) = [x_1(t), ..., x_i(t), ..., x_I(t)]^\mathsf{T}, \tag{2.1}$$

où $x_i(t)$ représente la valeur du $i^{\text{ième}}$ canal du mélange à l'instant t. À l'ère du numérique, la variable de temps t est généralement à valeurs discrètes et on parle alors de signal *échantillonné*, chaque composante $x_i(t)$ représentant alors un *échantillon* du mélange. Lorsque le mélange n'est composé que d'un seul canal, on parle de mélange *monophonique* ($I = 1$) par opposition à un mélange *stéréophonique* qui en comporte deux ($I = 2$). Au delà, les mélanges sont simplement qualifiés de multicanaux.

Historiquement, le format monophonique est resté le seul format d'enregistrement et de diffusion audio jusqu'à la première moitié du XX^e siècle. Depuis, le format stéréophonique est devenu le standard de distribution des contenus audio, aussi bien sur internet qu'à la télévision ou à la radio. Par rapport au format monophonique, le format stéréophonique ajoute au rendu sonore une impression d'espace, enrichissant ainsi l'expérience de l'auditeur, lui donnant l'illusion que les sons qui composent le mélange proviennent de différentes positions, ponctuelles ou étendues, dans l'espace. Cette impression d'espace est obtenue en restituant simplement chaque canal du mélange sur un canal de reproduction distinct comme le font la plupart des systèmes de reproduction domestiques actuels. Par exemple, les casques audio (ou les enceintes stéréo) reproduisent le canal gauche sur l'écouteur gauche (ou l'enceinte gauche) et le canal droit sur l'écouteur droit (ou l'enceinte droite). Notons toutefois que de nombreux paramètres tels que la position de l'auditeur ou le type de dispositif utilisé pour la reproduction peuvent altérer le rendu de l'espace sonore.

Au cinéma, les formats de diffusion ont souvent à l'origine plus de deux canaux. En particulier, le format *5.1* composé de $I = 5$ canaux est depuis les années 70 le format traditionnellement utilisé pour la restitution sonore en salle de cinéma. Pour une utilisation domestique, ce format présente toutefois l'inconvénient de requérir un système de restitution dédié, formé d'au moins cinq enceintes (plus une pour la diffusion des basses fréquences). Récemment, avec l'essor de l'image et du son 3D, de nouveaux formats font leur apparition et le nombre de canaux qu'ils emploient ne cesse de croître.

2.1.1 Formalisation du problème

Indépendamment du nombre de canaux, un mélange audio est comme son nom l'indique le résultat de l'assemblage de plusieurs objets sonores, nommés sources. Cet assemblage peut être réalisé de différentes façons selon, par exemple, le type de production (musique, cinéma, *etc.*), ou la nature des sources à mélanger (sources monophoniques, sources multicanales). Selon le cas, le terme de *source* et l'objectif même de la séparation de sources peuvent alors prendre des sens très différents. Il convient donc dès à présent de définir ce que nous nommerons *source* tout au long de ce document.

Définition d'une source : le modèle de mélange linéaire

Pour illustrer nos propos, prenons pour exemple la production d'un morceau de musique actuelle. Comme le montre la figure 2.1, une telle production est généralement réalisée en plusieurs temps, menant de l'enregistrement des instruments jusqu'au mélange audio $\mathbf{x}(t)$ qui sera distribué puis restitué sur un système de reproduction adapté. Une description détaillée d'un tel processus peut être trouvée dans [Sturmel et al., 2012]. Nous ne donnons ici qu'un exemple illustratif pour la suite de nos travaux.

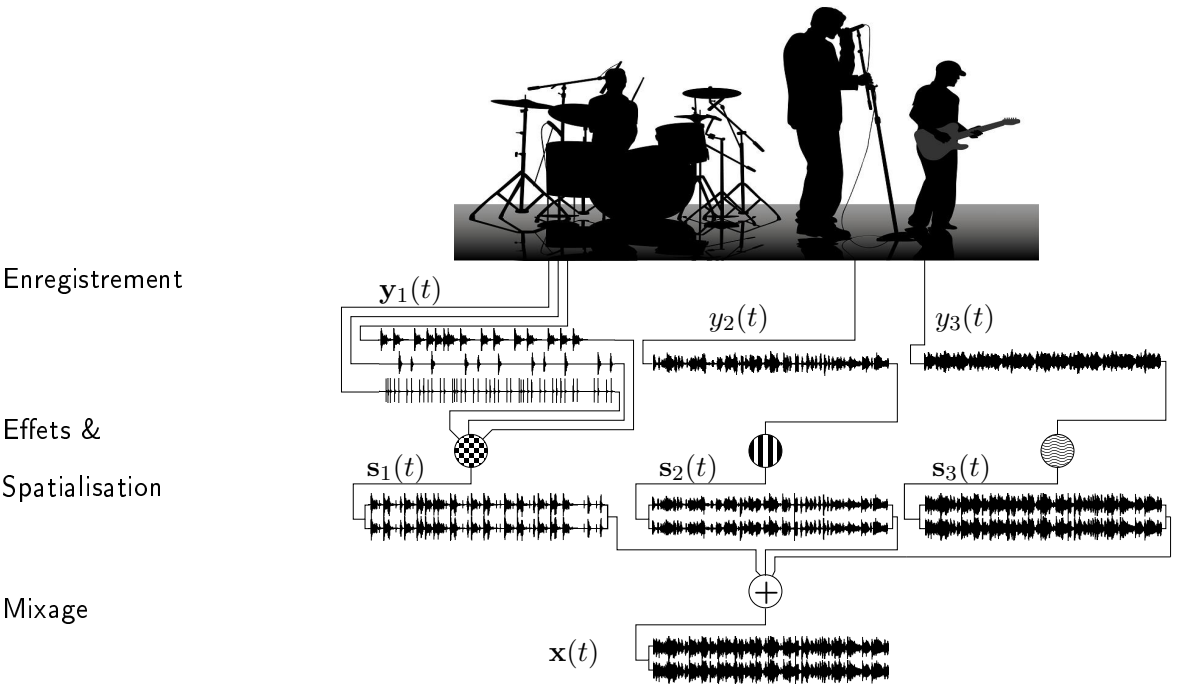


Figure 2.1 – Exemple de production d'un morceau de musique actuelle.

La première étape consiste généralement à enregistrer les instruments composant le morceau à l'aide de microphones, menant à l'acquisition de J signaux $\mathbf{y}_j(t)$. Chaque signal ainsi enregistré peut n'être relatif qu'à un seul instrument, comme c'est le cas dans notre exemple pour la voix ou la guitare, ou à plusieurs comme c'est le cas pour la batterie. Selon le cas, le signal $\mathbf{y}_j(t)$ pourra donc être à un ou plusieurs canaux.

La deuxième étape consiste généralement à transformer ces signaux à l'aide d'effets (compression, distorsion, filtrage, *etc.*). Ces effets, qui peuvent être appliqués à chaque signal $\mathbf{y}_j(t)$ ou à tous les signaux, participent entièrement à la dimension artistique de la création.

Les signaux ainsi obtenus sont une dernière fois transformés afin de donner à chacun une position, possiblement changeante, dans l'espace sonore créé par les canaux du mélange $\mathbf{x}(t)$. En

d'autres termes, cette dernière étape permet de transformer chaque signal $\mathbf{y}_j(t)$ en son image spatiale $\mathbf{s}_j(t)$ de même dimension que le mélange $\mathbf{x}(t)$. Le mélange est alors finalement obtenu par simple addition des J images spatiales $\mathbf{s}_j(t)$ selon l'équation dite de *mélange linéaire* [Cardoso, 1998] :

$$\forall t, \quad \mathbf{x}(t) = \sum_{j=1}^{J} \mathbf{s}_j(t). \tag{2.2}$$

Bien qu'elle ne soit pas représentée sur la figure 2.1, le mélange $\mathbf{x}(t)$ est souvent transformé lors d'une ultime étape dite de *mastering*, visant principalement à homogénéiser une production en vue de sa diffusion. Dans notre domaine d'application, le *mastering* est toutefois généralement négligé ou contourné [Sturmel et al., 2012].

La description de ce schéma classique de production d'un morceau de musique actuelle nous montre les multiples définitions que peut prendre le terme de *source*, et par extension, les différents objectifs que l'on peut entendre par *séparation de sources*. En effet, tous les signaux ci-dessus définis peuvent être qualifiés de sources, que ce soient les signaux acquis $\mathbf{y}_j(t)$, les signaux transformés ou les signaux spatialisés $\mathbf{s}_j(t)$.

Dans ce travail, nous ne retiendrons pourtant qu'une seule de ces définitions, à savoir celle donnée par le modèle de mélange linéaire (2.2). Ce modèle constitue en effet le paradigme le plus employé aujourd'hui en séparation de sources [Vincent et al., 2010a]. Tout au long de ce travail, nous admettrons donc que l'objectif de la séparation de sources consiste à estimer les J images spatiales des sources $\mathbf{s}_j(t)$ qui composent le mélange $\mathbf{x}(t)$ selon l'équation de mélange linéaire (2.2). Le problème de séparation de sources peut être défini de façon équivalente dans le domaine temps-fréquence selon

$$\forall f, n, \quad \mathbf{x}_{fn} = \sum_{j=1}^{J} \mathbf{s}_{j,fn}, \tag{2.3}$$

où $\mathbf{x}_{fn}$ représente la valeur de la Transformée de Fourier à Court-Terme (TFCT) du mélange $\mathbf{x}(t)$ au point temps-fréquence (f, n), avec f l'indice de fréquence et n l'indice de trame. De la même manière, $\mathbf{s}_{j,fn}$ représente la TFCT de la source $\mathbf{s}_j(t)$.

Cas particulier : le modèle de mélange convolutif

D'autres modèles de mélange ont été proposés dans la littérature et certains sont des cas particuliers du modèle de mélange linéaire. En reprenant l'exemple de la figure 2.1, la plupart des effets appliqués aux signaux enregistrés $\mathbf{y}_j(t)$ pour obtenir les sources $\mathbf{s}_j(t)$ peuvent être représentés par une opération de filtrage.

Supposant que chaque signal $\mathbf{y}_j(t)$ est composé de R_j canaux tels que

$$\forall t, \quad \mathbf{y}_j(t) = \left[y_{j1}(t), ..., y_{jr}(t), ..., y_{jR_j}(t)\right]^\mathsf{T}, \tag{2.4}$$

l'équation de mélange linéaire (2.3) peut être réécrite grâce à l'introduction de la matrice dite de *mélange*, notée $\mathbf{A}_{fn}$, selon

$$\forall f, n, \quad \mathbf{x}_{fn} = \mathbf{A}_{fn}\mathbf{y}_{fn} \tag{2.5}$$

où $\mathbf{y}_{fn}$ désigne le vecteur de longueur $R = \sum_{j=1}^{J} R_j$ composé des J TFCT des signaux $\mathbf{y}_j(t)$. La matrice de mélange $\mathbf{A}_{fn}$ est donc de taille $I \times R$ et est obtenue par concaténation de J matrices $\mathbf{A}_{j,fn}$, de taille $I \times R_j$ chacune, de sorte que

$$\mathbf{A}_{fn} = [\mathbf{A}_{1,fn}, ..., \mathbf{A}_{j,fn}, ..., \mathbf{A}_{J,fn}]. \tag{2.6}$$

L'équation de mélange peut alors être réécrite selon

$$\forall f, n, \quad \mathbf{x}_{fn} = \sum_{j=1}^{J} \mathbf{A}_{j,fn} \mathbf{y}_{j,fn}. \tag{2.7}$$

La matrice $\mathbf{A}_{j,fn}$ modélise donc la manière dont le $j^{\text{ième}}$ signal $\mathbf{y}_{j,fn}$ a été réparti sur les I canaux du mélange $\mathbf{x}_{fn}$ de sorte que $\mathbf{s}_{j,fn} = \mathbf{A}_{j,fn}\mathbf{y}_{j,fn}$. Ainsi, il est important de noter que si les équations (2.2) et (2.3) sont strictement équivalentes, l'introduction des matrices de mélange $\mathbf{A}_{j,fn}$ font des équations (2.5) et (2.7) une approximation de l'équation de mélange initiale (2.2).

Toutefois, ce formalisme modélise bien de nombreuses techniques de mixage traditionnelles, dont deux cas ont été particulièrement étudiés. Lorsque la matrice de mélange est indépendante de la trame n et de la fréquence f, la matrice de mélange s'écrit $\forall f, n, \mathbf{A}_{fn} = \mathbf{A}$ et le modèle obtenu est dit *linéaire instantané*. Plus généralement, lorsque la matrice de mélange ne dépend que de la fréquence f, le mélange est alors qualifié de *convolutif*. La matrice de mélange est alors simplement notée $\mathbf{A}_f$.

Séparation sous-déterminée et séparation sur-déterminée

La complexité du problème de séparation de sources, qu'il soit défini par l'équation de mélange linéaire (2.2) ou l'équation (2.5), dépend principalement du nombre de canaux I du mélange et du nombre de sources R à estimer. Lorsque nous disposons de plus de canaux qu'il n'y a de sources à estimer ($I > R$), le problème est dit *sur-déterminé* (ou *déterminé* lorsque $I = R$). À l'inverse, lorsqu'il y a plus de sources à estimer que de canaux ($I < R$), le problème est qualifié de *sous-déterminé*.

Dans cette thèse, nous ne nous intéresserons qu'au cas le plus complexe : le cas sous-déterminé. En effet, en adoptant le modèle (2.5), dans les cas déterminé et sur-déterminé, l'estimation des J sources $\mathbf{s}_{fn}$ qui composent le mélange peut être simplement menée par l'estimation de la matrice de mélange $\mathbf{A}_{fn}$. Une fois estimée, dans le cas déterminé, l'inversion de cette matrice suffit à recouvrir les composantes $\mathbf{y}_{fn}$ et donc les sources $\mathbf{s}_{fn}$. La méthode est similaire dans le cas sur-déterminé. Des méthodes de réduction de dimension telles que l'Analyse en Composantes Principales (PCA pour *Principal Component Analysis*) [Jolliffe, 2002] sont simplement employées au préalable. Pour estimer la matrice de mélange, l'Analyse en Composantes Indépendantes (ICA pour *Independent Component Analysis*) [Hyvärinen et al., 2004] a été très largement étudiée.

En revanche, dans le cas sous-déterminé, la connaissance de la matrice de mélange ne suffit pas à estimer les sources et des méthodes plus avancées doivent être employées. Nous en donnons un aperçu dans la partie suivante. Le problème sous-déterminé est plus complexe mais il répond aussi à des enjeux industriels très concrets, tels que j'ai pu en rencontrer dans mes expériences professionnelles passées. Il m'a donc semblé plus naturel de porter mon attention sur ce cas.

2.1.2 Méthodes pour la séparation de sources sous-déterminée

Comme nous l'avons introduit dans la partie précédente, nous nous intéressons ici aux méthodes de séparation de sources dans le cas sous-déterminé ($R > I$). Elles peuvent être réparties en trois classes principales dont nous donnerons un bref aperçu ci-après : les méthodes psychoacoustiques, les méthodes spatiales et les méthodes spectrales [Vincent, 2006]. D'autres méthodes dites *hybrides* peuvent emprunter à ces trois grandes classes certains de leurs principes.

Méthodes psychoacoustiques

L'analyse de scène auditive computationnelle (souvent désignée par l'acronyme *CASA*, pour *Computational Auditory Scene Analysis*) a pour objectif d'identifier les objets sonores qui com-

posent une scène (ici, notre mélange $\mathbf{x}(t)$) et de les grouper selon des critères psychoacoustiques (inspirés du fonctionnement du système auditif humain) [Bregman, 1994; Ellis, 1996; Godsmark et Brown, 1999]. Plutôt que d'opérer les groupements dans le domaine temporel, les techniques de type CASA emploient traditionnellement une représentation temps-fréquence du mélange inspirée du fonctionnement de la cochlée et nommé *cochléogramme*. Le cochléogramme est obtenu à l'aide d'un banc de filtres. Le calcul des corrélations entre chaque sortie du banc de filtres permet alors de construire une représentation tridimensionnelle nommée *corrélogramme* [Slaney et Lyon, 1990].

L'identification des objets sonores et leur regroupement en sources sont principalement réalisés à partir de ce corrélogramme. Ainsi, deux objets sonores pourront être groupés au sein d'une même source si, par exemple, ils sont en relation harmonique, leurs attaques sont simultanées, l'évolution de leurs amplitudes est corrélée, *etc*. Certains critères peuvent être basés sur un apprentissage préalable des caractéristiques d'une source sur des extraits isolés de cette source [Eggink et Brown, 2003; Kashino et al., 1995; Kinoshita et al., 1999]. Bien qu'originellement les méthodes de type CASA ont été développées pour la séparation de mélanges monocanaux, il a été proposé d'intégrer au processus de groupement des critères basés sur l'estimation de la localisation des sons et leur proximité dans l'espace sonore [Nakatani, 2002; Sakuraba et Okuno, 2003]. Une fois les objets sonores identifiés groupés en sources, ces dernières sont généralement extraites par masquage temps-fréquence binaire, dont nous verrons le principe dans la partie 2.1.4.

Méthodes spatiales

Les méthodes spatiales peuvent être utilisées lorsque le mélange possède plus d'un canal ($I > 1$). Comme nous l'avons indiqué plus tôt, lorsque le problème de séparation de sources est sur-déterminé, les méthodes les plus employées sont l'analyse en composantes principales (PCA) et l'analyse en composantes indépendantes (ICA). Lorsque le problème est sous-déterminé, les méthodes spatiales supposent généralement que les sources à séparer sont disjointes dans le plan temps-fréquence et l'objectif de la séparation revient souvent à estimer un masque temps-fréquence binaire pour chaque source.

Pour ce faire, certaines méthodes ont directement été dérivées de l'ICA dans le cas sur-déterminé [Gribonval et Lesage, 2006; Plumbley et al., 2010]. Ces méthodes, réunies sous le terme de *Sparse Component Analysis* (SCA), supposent qu'il existe un domaine transformé dans lequel les sources à séparer sont parcimonieuses. L'objectif de la séparation consiste alors à trouver une telle transformation, à estimer dans ce domaine la matrice de mélange puis à estimer les sources par inversion de la matrice de mélange estimée. La transformée en ondelettes ou la TFCT sont des choix classiques de transformation. L'estimation de la matrice de mélange est elle menée à l'aide de la traditionnelle ICA ou par *clustering* de type *K-means*.

Une autre catégorie de méthodes spatiales regroupe les méthodes basées sur le calcul de la Différence d'Intensité Interaurale, de la Différence de Temps Interaurale [Roman et al., 2003; Viste et Evangelista, 2003] et de la Différence de Phase Interaurale [Mandel et al., 2010]. Ces grandeurs ont en commun qu'elles permettent d'identifier la source prépondérante en chaque point temps-fréquence par estimation de sa direction d'arrivée. De façon similaire, la célèbre méthode *DUET* (pour *Degenerate Underdetermined Estimation Technique*) [Jourjine et al., 2000; Rickard et Yilmaz, 2002] s'appuie sur la détermination d'un couple *atténuation-délai* calculé à partir du simple rapport des TFCTs des canaux gauche et droit d'un mélange stéréo. Les sources sont alors identifiées par clustering des points temps-fréquence selon la valeur de ces couples *atténuation-délai*.

Méthodes spectrales

Lorsque le mélange ne possède qu'un seul canal ($I = 1$), aucune des méthodes spatiales présentées ci-dessus n'est applicable. Les méthodes dites spectrales constituent alors une alternative aux méthodes psychoacoustiques. Comme leur nom l'indique, les méthodes spectrales s'attachent à discriminer les sources entre elles par leurs caractéristiques spectrales. Généralement, il s'agit de décomposer une représentation temps-fréquence du mélange (souvent, le spectrogramme d'amplitude $|\mathbf{X}|$ ou le spectrogramme de puissance $|\mathbf{X}|^{\cdot 2}$ du mélange) comme la somme de spectres de base pondérés par des coefficients d'activation variant au cours du temps. Les sources peuvent être alors identifiées selon des critères d'indépendance, de non-négativité ou de parcimonie sur tout ou partie de la décomposition.

Les premières méthodes spectrales peuvent être regroupées sous le terme d'*analyse en sous-espaces indépendants* (*ISA* pour *Independent Subspace Analysis*). L'ISA peut être vue comme une extension de l'ICA [Theis, 2006] où, au lieu de décomposer le mélange $\mathbf{x}$ en un certain nombre de composantes unidimensionnelles indépendantes, il s'agit de décomposer ce même mélange en un certain nombre de composantes puis de former à partir de ces composantes des groupes indépendants (d'où le terme de sous-espace indépendant). L'idée d'ISA a été initialement proposée dans [Hyvarinen, 1999] et a été popularisée par [Casey et Westner, 2000]. La décomposition en composantes peut être préalablement obtenue par *ICA multidimensionnelle* [Cardoso, 1998]. Le regroupement en sous-espaces indépendants est lui obtenu en maximisant soit l'indépendance des bases de la décomposition [Casey et Westner, 2000; Smaragdis, 2001], soit l'indépendance des activations temporelles [FitzGerald, 2004], soit l'indépendance conjointe des deux [Stone et al., 1999]. Comme les spectrogrammes d'amplitude et de puissance sont par définition à valeurs positives, il a également été proposé de contraindre les bases et activations à être elles aussi à valeurs positives. De ce fait, la décomposition obtenue est purement additive et il est alors plus aisé de lui donner un sens physique. L'ICA avec contrainte de non-négativité a notamment été étudiée dans [Plumbley et Oja, 2004].

D'autres méthodes ont été proposées exploitant également cette idée de décomposition en termes non-négatifs mais abandonnant l'indépendance comme critère d'identification des sources. C'est le cas notamment des méthodes basées sur la factorisation en matrices non-négatives [Smaragdis et Brown, 2003] (*NMF* pour *Nonnegative Matrix Factorization* en anglais). Ces méthodes, très populaires, feront à ce titre l'objet d'une présentation plus détaillée dans la partie 2.1.3.

Une troisième catégorie de méthodes spectrales exploite, de façon similaire à la SCA, une contrainte de parcimonie pour identifier les sources. Cette fois, ces méthodes dites par *codage parcimonieux* (ou *sparse coding* en anglais) supposent qu'il existe une base dans laquelle les sources sont parcimonieuses. Dans cette base, chaque source est donc représentée par un petit nombre seulement de coefficients d'activation non-nuls. De nombreuses méthodes par codage parcimonieux utilisent également un critère de non-négativité [Benaroya et al., 2003; Hoyer, 2002; Virtanen, 2003]. Lorsque les bases de la représentation parcimonieuse sont construites à partir de caractéristiques morphologiques des sources à séparer, les méthodes peuvent être également identifiées par le terme d'*analyse en composantes morphologiques* (*MCA* pour *Morphological Component Analysis* en anglais) [Gribonval et Lesage, 2006; Starck et al., 2005].

Enfin, les récentes avancées en apprentissage profond, dont nous verrons les grands principes dans le chapitre 6, ont fait naître quelques propositions de méthodes de séparation de sources par réseau de neurones. Qu'ils soient proactifs ou rétroactifs, les réseaux profonds proposés sont utilisés afin de déterminer soit des masques temps-fréquence idéaux [Huang et al., 2014b; Narayanan et Wang, 2013; Weninger et al., 2014], soit les spectres d'amplitude des sources [Grais et al., 2014; Huang et al., 2014c].

Méthodes hybrides

Comme nous l'avons indiqué en introduction, certaines méthodes de séparation de sources sous-déterminée peuvent emprunter leur principe à plusieurs des trois grands ensembles de méthodes ci-dessus présentés. C'est le cas notamment des méthodes spectrales qui, si elles ont été initialement développées pour des mélanges monocanaux, peuvent elles aussi tirer partie de la diversité spatiale dans le cas de mélanges multicanaux en étant associées à des méthodes spatiales. Sans être exhaustifs, nous pouvons citer, par exemple, SCA et MCA qui ont été associées sous le terme de MCA multicanale (*Multichannel MCA*) dans [Bobin et al., 2005]. De la même manière, la NMF multicanale a été proposée dans [Ozerov et Févotte, 2010; Vincent, 2006].

2.1.3 Factorisation en matrices non-négatives (NMF)

Nous proposons dans cette partie d'introduire la factorisation en matrices non-négatives (*NMF* pour *Nonnegative Matrix Factorization* en anglais) qui est, dans le cas sous-déterminé, l'un des modèles spectraux les plus employés en séparation de sources. Cet engouement se justifie à la fois par la simplicité de la décomposition proposée ainsi que par la contrainte de non-négativité qui permet une interprétation physique de la factorisation obtenue. En effet, la publication à l'origine de la popularité de la NMF [Lee et Seung, 1999] a montré que la factorisation d'images de visages permettait d'extraire les différentes parties qui composent un visage, à savoir le nez, la bouche, les yeux, *etc*. De nombreuses applications à d'autres types de signaux et d'autres formes de factorisation non-négatives ont alors été dérivées de cette application. L'ouvrage [Cichocki et al., 2009] en donne une description détaillée.

Dans ce chapitre, nous nous intéresserons principalement à la NMF originelle, appliquée à la factorisation de spectrogrammes, dans le cas monocanal.

Principe de la NMF déterministe

A l'origine, la NMF a pour objectif d'approximer une matrice non-négative $\mathbf{M}$ de taille $F \times N$ par une autre matrice non-négative $\mathbf{V}$ résultant du produit de deux matrices également non-négatives $\mathbf{W}$ et $\mathbf{H}$ de tailles respectives $F \times K$ et $K \times N$ de sorte que

$$\mathbf{M} \approx \mathbf{V} = \mathbf{WH}. \tag{2.8}$$

En notant $\mathbf{V} = \{v_{fn}\}$, $\mathbf{W} = \{w_{fk}\}$ et $\mathbf{H} = \{h_{kn}\}$, la NMF s'écrit

$$\forall f, n, \quad v_{fn} = \sum_{k=1}^{K} w_{fk} h_{kn}. \tag{2.9}$$

L'ordre K de la NMF est généralement choisi tel que $K(F+N) \ll FN$ de sorte que la NMF peut être vue comme une méthode de réduction de la dimension de la matrice $\mathbf{V}$. La matrice $\mathbf{W}$ est généralement nommée *dictionnaire* alors que $\mathbf{H}$ est souvent désignée par le terme de *matrice d'activations*.

Pour la séparation de sources, il a été proposé d'appliquer la NMF à la décomposition de spectrogrammes d'amplitude $|\mathbf{X}|$ ou à la décomposition de spectrogrammes de puissance $|\mathbf{X}|^{.2}$, où $\mathbf{X}$ désigne la TFCT du mélange $\mathbf{x}(t)$. La notation $|\mathbf{X}|$ désigne la matrice formée des valeurs absolues des coefficients de la matrice $\mathbf{X}$ et $|\mathbf{X}|^{.p}$ représente son exponentiation terme à terme. Il a également été envisagé dans [Hennequin, 2010; Smaragdis et al., 2009] de factoriser le spectrogramme de puissance élevé à une puissance non-entière $p \in [1, 2]$ de sorte que la matrice à factoriser se trouve être $|\mathbf{X}|^{.p}$. Quel que soit p, le spectrogramme $|\mathbf{X}|^{.p}$ est bien à valeurs positives

ou nulles et peut donc être approché par le produit des matrices $\mathbf{W}$ et $\mathbf{H}$ elles aussi non-négatives, tel que

$$|\mathbf{X}|^{\cdot p} \approx \mathbf{WH}. \tag{2.10}$$

La figure 2.2 illustre le principe de la NMF pour la décomposition du spectrogramme de notes jouées au piano. Le spectrogramme du mélange (ici, pour $p = 1$) est représenté dans le quart en haut à gauche. Il est composé de trois évènements harmoniques distincts. Ces évènements correspondent à des notes jouées par le piano. Une première note est d'abord jouée à l'instant $t = 0$ s, puis une deuxième note est jouée à $t = 15$ s. Enfin, ces deux notes sont jouées simultanément à l'instant $t = 30$ s. Le dictionnaire $\mathbf{W}$ et la matrice d'activation $\mathbf{H}$ obtenus par NMF avec $K = 2$ composantes sont également tracés sur la figure (en bas à gauche pour $\mathbf{H}$ et en haut à droite pour $\mathbf{W}$). Nous constatons que la décomposition obtenue fait sens. En effet, les deux spectres $\mathbf{w}_k = [w_{1k}, ..., w_{fk}, ..., w_{Fk}]^\intercal$ formant les colonnes du dictionnaire $\mathbf{W} = [\mathbf{w}_1 \mathbf{w}_2]$ représentent chacun le spectre moyen d'une des deux notes jouées dans le mélange. De leur côté, les lignes $\mathbf{h}_k = [h_{k1}, ..., h_{kn}, ..., h_{kN}]$ de la matrice d'activation $\mathbf{H} = [\mathbf{h}_1 \mathbf{h}_2]^\intercal$ indiquent à quels instants les spectres du dictionnaire sont actifs et avec quelle amplitude.

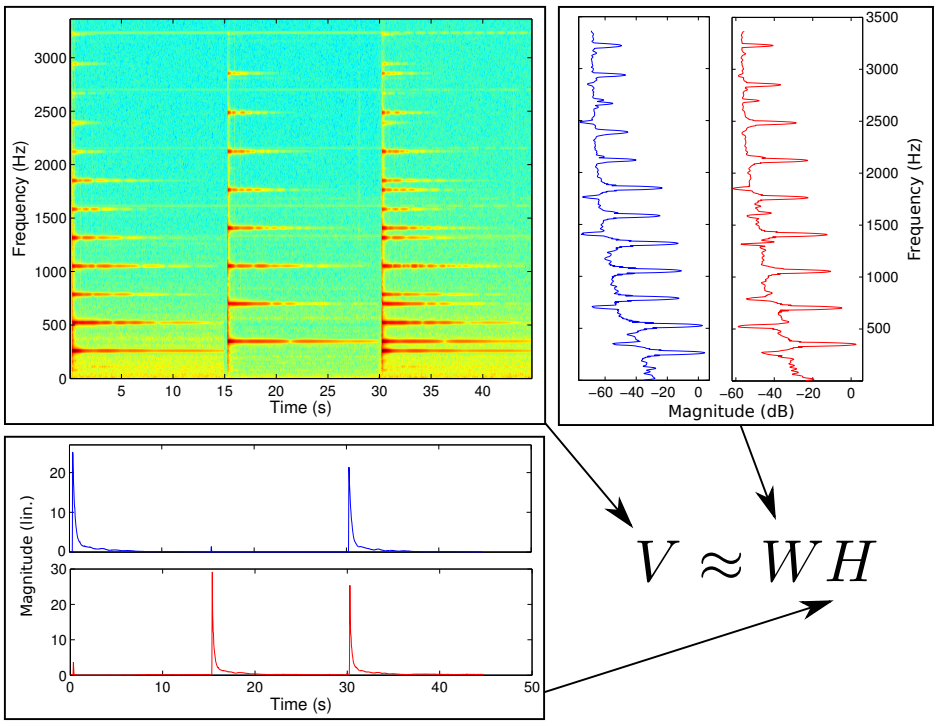


FIGURE 2.2 – Exemple de factorisation en matrices non-négatives d'un spectrogramme.

Nous comprenons donc que la NMF appliquée au spectrogramme $|\mathbf{X}|^{\cdot p}$ revient à approcher le spectrogramme par la somme de K composantes, notées ci-après $\mathbf{V}_k$, résultant chacune du produit du spectre de base $\mathbf{w}_k$ et de l'activation correspondante $\mathbf{h}_k$, tel que

$$|\mathbf{X}|^{\cdot p} \approx \sum_{k=1}^{K} \mathbf{V}_k = \sum_{k=1}^{K} \mathbf{w}_k \mathbf{h}_k. \tag{2.11}$$

Chacune des composantes $\mathbf{V}_k$ est donc homogène au spectrogramme $|\mathbf{X}|^{\cdot p}$.

NMF déterministe par minimisation d'une fonction de coût

La NMF est généralement formulée comme un problème de minimisation d'une fonction de coût entre l'observation $|\mathbf{X}|^{\cdot p}$ et le modèle $\mathbf{V} = \mathbf{WH}$:

$$\underset{\mathbf{W},\mathbf{H}}{\operatorname{argmin}} \quad \mathcal{D}(|\mathbf{X}|^{\cdot p}|\mathbf{WH}). \tag{2.12}$$

La fonction de coût utilisée est le plus souvent une *divergence* dite *séparable* car vérifiant

$$\mathcal{D}(|\mathbf{X}|^{\cdot p}|\mathbf{WH}) = \sum_{f=1}^{F}\sum_{n=1}^{N} d\left(|x_{fn}|^p \middle| \sum_{k=1}^{K} w_{fk}h_{kn}\right). \tag{2.13}$$

De nombreuses divergences ont été utilisées dans la littérature. En particulier, la distance euclidienne a été originellement utilisée dans [Lee et Seung, 1999]. Deux autres divergences ont été largement utilisées depuis dans les applications audio : la divergence de Kullback-Leibler (KL) introduite dans [Kullback et Leibler, 1951] et la divergence d'Itakura-Saito (IS) introduite dans [Itakura et Saito, 1968]. Ces trois distances sont toutefois des cas particuliers ou limites d'une classe plus large de divergences nommées β-*divergences* [Févotte et Idier, 2011] et définies par

$$\forall \beta \in \mathbb{R} \setminus \{0,1\}, \quad \forall x, y \in \mathbb{R}, \quad d_\beta(x|y) = \frac{1}{\beta(\beta-1)}(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}). \tag{2.14}$$

La distance euclidienne correspond au cas $\beta = 2$. La divergence de KL correspond au cas limite $\beta \to 1$ et la divergence d'IS à $\beta \to 0$.

Dans la suite de ce rapport, nous utiliserons systématiquement la divergence d'IS qui est définie par

$$d_{\mathsf{IS}}(x|y) = \frac{x}{y} - \log\frac{x}{y} - 1. \tag{2.15}$$

Il a été montré dans [Bertin et al., 2009b; Févotte et al., 2009] que cette divergence est bien adaptée à la NMF de spectrogrammes audio. En effet, contrairement aux autres β-divergences, la divergence d'IS est la seule à être invariante par changement d'échelle. Cette propriété signifie que

$$\forall x, y, \quad \forall \lambda \in \mathbb{R}, \quad d_{\mathsf{IS}}(\lambda x|\lambda y) = d_{\mathsf{IS}}(x|y). \tag{2.16}$$

De ce fait, la fonction de coût (2.13) pénalisera d'autant une mauvaise approximation pour un point temps-fréquence de faible amplitude ($|x_{fn}|^p$ petit) que pour un points temps-fréquence d'amplitude plus grande. Si l'on prend pour exemple la production d'une note de piano, on comprend bien que cette propriété est désirable pour notre objectif de séparation de sources. En effet, une note de piano se décompose grossièrement en deux événements : l'attaque, correspondant à l'instant où le marteau vient frapper la corde, et la phase de chute (*decay* en anglais) qui est le résultat de la vibration de la corde et de son amplification par la table d'harmonie et qui dure jusqu'à son extinction naturelle ou l'extinction volontairement provoquée par l'étouffoir (partie du mécanisme du piano moderne). Sur un spectrogramme, l'attaque se traduit par une excitation large bande de faible puissance alors que la phase de chute est caractérisée par une excitation harmonique, donc localisée en fréquence et de plus forte puissance. Pour autant, ces deux événements participent entièrement au timbre de la note et ont donc la même importance au sens de notre objectif de séparation.

Algorithmes d'optimisation

De nombreux algorithmes ont été proposés afin de résoudre le problème de minimisation (2.13), quelle que soit la divergence choisie [Cichocki et al., 2009]. Par exemple, les méthodes standards telles que l'algorithme de descente de gradient [Lin, 2007; Wang et Zou, 2008] et l'algorithme de Newton peuvent être employées [Zdunek et Cichocki, 2007], à condition de considérer leurs variantes *projetées* afin de respecter la contrainte de non-négativité de la factorisation. D'autres méthodes permettent d'implicitement préserver cette contrainte. C'est le cas notamment des algorithmes à mises à jour multiplicatives, originellement employés dans [Lee et Seung, 1999] et que nous allons ici présenter.

Comme leur nom l'indique, ces algorithmes permettent d'écrire les mises à jour des paramètres NMF $\mathbf{W}$ et $\mathbf{H}$ sous forme multiplicative. La dérivation de ces règles de mise à jour peut se faire à l'aide d'une approche par Majoration/Minimisation telle que décrite dans [Févotte et Idier, 2011; Kompass, 2007]. Ces algorithmes peuvent être également dérivés de façon simple mais heuristique. Pour ce faire, il convient simplement de dériver la fonction de coût (2.13) par rapport à chacun des paramètres NMF θ (un coefficient du dictionnaire w_{fk} ou un coefficient de la matrice d'activation h_{kn}) et d'écrire cette dérivée comme la différence de deux termes $P(\theta)$ et $Q(\theta)$ tous deux positifs ou nuls, tel que

$$\frac{\partial \mathcal{D}(|\mathbf{X}|^{\cdot p}|\mathbf{WH})}{\partial \theta} = P(\theta) - Q(\theta). \tag{2.17}$$

La mise à jour multiplicative du paramètre θ s'écrit alors

$$\theta \leftarrow \theta \frac{P(\theta)}{Q(\theta)}. \tag{2.18}$$

En appliquant ce principe à chacun des paramètres de la NMF et en reprenant les notations introduites dans la formulation de la NMF pour la séparation de sources (2.22), les mises à jours des paramètres de la NMF relatifs à la source j peuvent s'écrire, pour une β-divergence quelconque, sous la forme matricielle suivante :

$$\mathbf{W}_j \leftarrow \mathbf{W}_j \circ \frac{\left(\mathbf{V}^{\cdot(\beta-2)} \circ |\mathbf{X}|^{\cdot p}\right) \mathbf{H}_j^\intercal}{\mathbf{V}^{\cdot(\beta-1)} \mathbf{H}_j^\intercal}, \tag{2.19}$$

$$\mathbf{H}_j \leftarrow \mathbf{H}_j \circ \frac{\mathbf{W}_j^\intercal \left(\mathbf{V}^{\cdot(\beta-2)} \circ |\mathbf{X}|^{\cdot p}\right)}{\mathbf{W}_j^\intercal \mathbf{V}^{\cdot(\beta-1)}}, \tag{2.20}$$

où la multiplication (notée $\circ$), la division et l'exponentiation sont effectuées terme à terme.

Application à la séparation de sources

Supposons à présent un mélange $x(t)$ monocanal ($I = 1$) défini selon le modèle linéaire de l'équation (2.2). Dans le domaine temps fréquence, la TFCT du mélange $\mathbf{X} = \{x_{fn}\}_{n=1..N}^{f=1..F}$ s'exprime comme la somme des J TFCT des sources $\mathbf{S}_j = \{s_{j,fn}\}_{n=1..N}^{f=1..F}$ qui composent le mélange selon

$$\mathbf{X} = \sum_{j=1}^{J} \mathbf{S}_j. \tag{2.21}$$

Comme nous l'avons évoqué dans la partie précédente, la NMF consiste à approcher le spectrogramme du mélange $|\mathbf{X}|^{\cdot p}$ comme la somme de K composantes $\mathbf{V}_k$ homogènes au spectrogramme du mélange. La séparation des sources composant le mélange est alors menée en associant chacune

des K composantes à l'une des J sources du mélange. De ce fait, la factorisation du spectrogramme du mélange (2.10) peut se réécrire en termes relatifs à chacune des sources selon

$$|\mathbf{X}|^{\cdot p} \approx \mathbf{WH} = \sum_{j=1}^{J} \mathbf{W}_j \mathbf{H}_j, \tag{2.22}$$

où $\mathbf{W}$ (respectivement $\mathbf{H}$) est composée des colonnes (resp., des lignes) des J matrices $\mathbf{W}_j$ (resp. $\mathbf{H}_j$). Le dictionnaire $\mathbf{W}_j$ est composé de K_j spectres de base décrivant la $j^{ième}$ source et $\mathbf{H}_j$ représente les activations correspondant à ces spectres. Le spectrogramme $\mathbf{V}_j = \mathbf{W}_j\mathbf{H}_j$ est alors traditionnellement identifié au spectrogramme $|\mathbf{S}_j|^p$ de la $j^{ième}$ source.

De ce fait, dans la suite de ce rapport, nous parlerons de *modèle NMF* car chacune des sources qui composent le mélange est décrite par une NMF propre modélisant son spectrogramme $|\mathbf{S}_j|^{\cdot p}$. Afin d'obtenir une factorisation dans laquelle les composantes $\mathbf{V}_k$ peuvent être attribuées aux différentes sources, il convient d'introduire de l'information *a priori* afin de favoriser une décomposition faisant sens pour l'application de séparation de sources désirée. Nous verrons certaines méthodes permettant cela plus tard dans cette partie.

Pour le moment, nous voudrions revenir sur l'approximation faite par l'utilisation de modèles NMF pour la séparation de sources. Comme nous l'avons indiqué plus tôt, la NMF $\mathbf{V}_j$ est identifiée au spectrogramme $|\mathbf{S}_j|^{\cdot p}$ de la $j^{ième}$ source composant le mélange. Cela revient donc à décomposer le spectrogramme du mélange $|\mathbf{X}|^{\cdot p}$ comme la somme des spectrogrammes des sources selon

$$|\mathbf{X}|^{\cdot p} \approx \sum_{j=1}^{J} |\mathbf{S}_j|^{\cdot p}. \tag{2.23}$$

Bien entendu, puisque nous supposons que le mélange des sources est linéaire suivant (2.21), l'approximation ainsi formulée ne peut être exacte, et ce quel que soit p. Des travaux dans [Hennequin, 2010; Smaragdis et al., 2009] ont proposé de déterminer quel exposant p vérifiait au mieux l'approximation (2.23) et ont apporté quelques pistes de réflexion en ce sens. Pour notre travail, nous retiendrons que cette approximation ne posera problème que pour les points temps-fréquence où plusieurs sources ont une énergie significative et comparable. D'après l'étude menée dans [Parvaix et Girin, 2011] sur des signaux musicaux, de tels points temps-fréquence sont marginaux puisque la plupart des points temps-fréquence du mélange ne contiennent qu'une source ayant une énergie significative.

Pour réaliser notre objectif de séparation, il est indispensable de reconstruire les sources estimées dans le domaine temporel $s_j(t)$ à partir de l'approximation de leurs spectrogrammes $|\mathbf{S}|^{\cdot p}$ par NMF. Cette phase est généralement réalisée à partir du principe de masquage temps-fréquence qui sera plus généralement décrit dans la partie 2.1.4.

En pratique, pour la NMF, cela revient à calculer les coefficients $\tilde{s}_{j,fn}$ de la TFCT de la j^{ime} source estimée selon

$$\tilde{s}_{j,fn} = \frac{v_{j,fn}}{\sum_{j'=1}^{J} v_{j',fn}} x_{fn}, \tag{2.24}$$

où $v_{j,fn} = \sum_{k=1}^{K_j} w_{j,fk} h_{j,kn}$ est l'estimée du spectrogramme de la source j par NMF.

Prise en compte d'informations *a priori*

Comme nous l'avons évoqué plus tôt, afin de pouvoir identifier les diverses sources parmi les composantes de la NMF, il convient généralement soit de prendre en compte des informations ou données auxiliaires, souvent qualifiées d'information *a priori*, soit d'introduire des connaissances *a priori* sur les sources à séparer.

Lorsque des informations *a priori* sont disponibles, la NMF est dite *supervisée*. Généralement, cela revient à initialiser certains de ses paramètres à l'aide de ces informations auxiliaires. Par exemple, il a été proposé dans [Ewert et Müller, 2012; Hennequin et al., 2011] d'initialiser le dictionnaire $\mathbf{W}_j$ d'une source à partir de la partition musicale de cette source. Le dictionnaire $\mathbf{W}_j$ peut être également appris à partir d'un ensemble d'apprentissage représentatif de la source à séparer. Par exemple, dans [Dessein et al., 2010; Niedermayer, 2008], il est question d'apprendre les spectres caractéristiques d'un instrument à partir d'enregistrements de notes isolées de ce même instrument dans des conditions proches de celles du mélange à séparer. Si les conditions d'enregistrement entre apprentissage et séparation sont différentes, il est proposé dans [Jaureguiberry et al., 2011] d'adapter le dictionnaire à l'aide de l'estimation d'un filtre modélisant la différence des conditions acoustiques. Plus tard dans ce manuscrit, nous mettrons en place un tel apprentissage afin d'apprendre un dictionnaire de spectres caractéristiques d'un locuteur.

Si aucune information auxiliaire n'est disponible, il est encore possible d'introduire des connaissances *a priori* sur les caractéristiques des sources à estimer. Pour cela, il est possible d'employer la technique classique dite de *régularisation*. Cela consiste à ajouter des termes de pénalité à la fonction de coût définie en (2.13) de telle sorte que la fonction à minimiser devient

$$\mathcal{C}(\mathbf{W},\mathbf{H}) = \mathcal{D}(|\mathbf{X}|^{\cdot p}|\mathbf{W}\mathbf{H}) + \lambda_{\mathbf{W}}\mathcal{C}_{\mathbf{W}}(\mathbf{W}) + \lambda_{\mathbf{H}}\mathcal{C}_{\mathbf{H}}(\mathbf{H}). \tag{2.25}$$

Les termes $\mathcal{C}_{\mathbf{W}}(\mathbf{W})$ et $\mathcal{C}_{\mathbf{H}}(\mathbf{H})$ ont pour objectif de contraindre les valeurs des paramètres NMF avec un poids déterminé par les scalaires $\lambda_{\mathbf{W}}$ et $\lambda_{\mathbf{H}}$ respectivement. De nombreuses contraintes ont été proposées dans la littérature. Nous citerons par exemple les contraintes de *parcimonie* étudiées dans [Eggert et Korner, 2004; Hoyer, 2004; Joder et al., 2013; Virtanen, 2007], les contraintes de *continuité temporelle* présentée dans [Bertin et al., 2009a; Virtanen, 2007] ou encore la contrainte de *décorrélation* introduite dans [Zhang et Fang, 2007]. Toutes ces contraintes ont été appliquées aux lignes de la matrice d'activation $\mathbf{H}$ par le biais du terme de pénalité $\mathcal{C}_{\mathbf{H}}(\mathbf{H})$.

Une autre façon de contraindre la factorisation sans information auxiliaire consiste à introduire des paramètres supplémentaires permettant de contraindre la structure du modèle NMF. Ces méthodes peuvent être regroupées sous le terme de méthodes par *paramétrisation*. Par exemple, [Ewert et al., 2013; Hennequin et al., 2010] proposent d'imposer aux spectres du dictionnaire $\mathbf{W}$ d'être des peignes harmoniques paramétrisés seulement par leur fréquence fondamentale et l'amplitude de leurs partiels. Dans [Bertin et al., 2010], les spectres du dictionnaire $\mathbf{W}$ sont exprimés comme la combinaison linéaire de spectres à bande étroite de sorte que les spectres résultants soient lisses. Similairement, dans [Ochiai et al., 2012], les activations $\mathbf{H}$ sont modélisées comme une somme de distributions gaussiennes centrées en fonction de la structure rythmique du mélange. Enfin, d'autres propositions consistent à refactoriser les matrices $\mathbf{W}$ et $\mathbf{H}$ en d'autres matrices elles aussi non-négatives. Comme nous étudierons de tels modèles tout au long de ce manuscrit, nous reviendrons plus tard sur ces méthodes.

Formulation probabiliste

La NMF pour la séparation de sources peut être également formalisée dans un cadre probabiliste. En particulier, un *modèle génératif gaussien* pour la NMF a été introduit dans [Févotte et al., 2009]. Nous présenterons ici ce modèle car il formera la base de notre travail sur la fusion bayésienne dans le chapitre 5. Pour les autres formulations probabilistes, on citera par exemple le *modèle génératif de Poisson* [Virtanen et al., 2008a] ou l'*analyse en composantes latentes* (*PLCA* pour *Probabilistic Latent Component Analysis* en anglais) [Shashanka et al., 2008] dont les formulations sont équivalentes à l'approche déterministe par minimisation de la divergence de KL pour la factorisation des spectrogrammes d'amplitude.

Le modèle génératif gaussien [Févotte et al., 2009] suppose que chaque coefficient x_{fn} de la TFCT du mélange est la somme de K variables latentes complexes $c_{k,fn}$ indépendantes deux à deux

$$\forall f,n, \quad x_{fn} = \sum_{k=1}^{K} c_{k,fn}. \tag{2.26}$$

Chaque variable $c_{k,fn}$ est supposée distribuée selon une loi normale complexe circulaire [Neeser et Massey, 1993] centrée dont la variance est le résultat de la NMF pour le point temps-fréquence considéré, soit

$$c_{k,fn} \sim \mathcal{N}(0, w_{fk}h_{kn}). \tag{2.27}$$

La log-vraisemblance de $\mathbf{X}$ sachant les paramètres de NMF $\mathbf{W}$ et $\mathbf{H}$ s'écrit donc

$$\log p(\mathbf{X}|\mathbf{W},\mathbf{H}) = \sum_{f=1}^{F}\sum_{n=1}^{N} \log \mathcal{N}\left(x_{fn}\middle|0, \sum_{k=1}^{K} w_{fk}h_{kn}\right). \tag{2.28}$$

Il a alors été démontré dans [Févotte et al., 2009] que l'estimation du maximum de vraisemblance de $\mathbf{W}$ et $\mathbf{H}$ était équivalente à la solution obtenue par résolution du problème déterministe (2.12) pour la divergence d'IS ($\beta \to 0$) entre le modèle et le spectrogramme de puissance ($p = 2$), puisque la log-vraisemblance peut s'écrire

$$\log p(\mathbf{X}|\mathbf{W},\mathbf{H}) = -\sum_{f=1}^{F}\sum_{n=1}^{N} d_{\text{IS}}\left(|x_{fn}|^2\middle|\sum_{k=1}^{K} w_{fk}h_{kn}\right) + \text{cte}. \tag{2.29}$$

Comme pour l'approche déterministe, afin d'identifier les sources du mélange, il convient d'exploiter des informations auxiliaires ou de contraindre la structure de la NMF. Grâce à ce formalisme, ces contraintes peuvent être désormais introduites sous forme d'*a priori* probabiliste. Dans ce cas, chaque source peut être décrite par une NMF propre telle que $s_{j,fn} \sim \mathcal{N}(0, v_{j,fn})$ où la variance $v_{j,fn} = \sum_{k=1}^{K_j} w_{j,fk}h_{j,kn}$ est le résultat d'une NMF d'ordre K_j. Les TFCTs des sources estimées peuvent être finalement estimées grâce l'équation de filtrage (2.24) déjà exprimée dans le cas déterministe. Dans ce cadre probabiliste, il a de plus été démontré dans [Févotte et al., 2009] que la source $\tilde{s}_j(t)$ ainsi obtenue est un estimateur de $s_j(t)$ sachant $v_{j,fn}$ au sens du minimum de l'erreur quadratique moyenne. La reconstruction des sources selon (2.24) est donc le résultat d'un filtrage de Wiener.

Prise en compte de l'information spatiale

Jusqu'à présent, nous n'avons envisagé la NMF que dans le cadre de mélanges monocanaux ($I = 1$). Les premiers travaux sur les mélanges à plusieurs canaux ont proposé soit de concaténer les spectrogrames de chacun des canaux afin de ne former qu'une seule matrice qui sera elle décomposée à l'aide d'une NMF classique [Parry et Essa, 2006], soit par application du principe de factorisation en tenseurs (matrices à trois dimensions) non-négatifs [FitzGerald et al., 2005]. Toutefois, ces deux méthodes supposent que le mélange des sources est linéaire instantané, ce qui réduit leur champ d'application.

Pour y remédier, la *NMF multicanale* a été introduite dans [Ozerov et Févotte, 2010]. La NMF multicanale s'inspire directement du modèle génératif gaussien décrit dans la partie précédente et introduit dans [Févotte et al., 2009]. Le modèle de mélange adopté est le modèle de mélange convolutif (voir équation (2.5)) et s'écrit donc dans le domaine temps-fréquence

$$\forall f,n, \quad \mathbf{x}_{fn} = \sum_{j=1}^{J} \mathbf{s}_{j,fn} + \mathbf{b}_{fn} = \mathbf{A}_f\mathbf{y}_{fn} + \mathbf{b}_{fn} \tag{2.30}$$

où le vecteur $\mathbf{b}_{fn} = [b_{1,fn}, ..., b_{i,fn}, ..., b_{I,fn}]^\intercal$ de longueur I modélise le bruit de capteur. Il est supposé gaussien et stationnaire tel que

$$\forall i, \quad b_{i,fn} \sim \mathcal{N}(0, \sigma^2_{i,f}). \tag{2.31}$$

Les sources $y_{j,fn}$ composant le vecteur $\mathbf{y}_{fn} = [y_{1,fn}, ..., y_{j,fn}, ..., y_{J,fn}]^\intercal$ sont supposées monocanales ($R_j = 1$) et distribuées selon une loi normale complexe circulaire dont la variance est exprimée comme le résultat d'une NMF $\mathbf{W}_j\mathbf{H}_j$ au point temps-fréquence considéré, tel que

$$\forall j, \quad y_{j,fn} \sim \mathcal{N}\left(0, v_{j,fn}\right), \tag{2.32}$$

avec $v_{j,fn} = \sum_{k=1}^{K_j} w_{j,fk} h_{j,kn}$. Notons au passage que le cas $R_j > 1$ a été également étudié, mais en vue de nos applications pratiques, nous nous limitons ici au cas $R_j = 1$.

Par rapport au modèle probabiliste monocanal présenté dans la partie précédente, la TFCT du mélange suit cette fois une loi normale complexe circulaire centrée, telle que

$$\mathbf{x}_{fn} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{x}_{fn}}), \tag{2.33}$$

dont la covariance est donnée par

$$\mathbf{\Sigma}_{\mathbf{x}_{fn}} = \mathbf{A}_f \mathbf{\Sigma}_{\mathbf{y}_{fn}} \mathbf{A}_f^H + \mathbf{\Sigma}_{\mathbf{b}_{fn}} \tag{2.34}$$

où $\mathbf{A}_f^H$ désigne la matrice transposée-conjuguée de la matrice de mélange $\mathbf{A}_f$ et $\mathbf{\Sigma}_{\mathbf{b}_{fn}} = \mathrm{diag}([\sigma^2_{i,f}]_i)$ est la matrice diagonale dont les éléments diagonaux sont les variances $\sigma^2_{i,f}$ de la variable de bruit de capteur. De la même manière, $\mathbf{\Sigma}_{\mathbf{y}_{fn}}$ est une matrice diagonale dont les éléments diagonaux sont les variances $v_{j,fn}$ des J sources $y_{j,fn}$.

En notant $\mathbf{A}$ l'ensemble des matrices de mélange $\mathbf{A}_f$, $\mathbf{W}$ et $\mathbf{H}$ l'ensemble des paramètres NMF $\mathbf{W}_j$ et $\mathbf{H}_j$ et $\mathbf{\Sigma}_\mathbf{b}$ l'ensemble des variances du bruit, les paramètres $\mathbf{Z} = \{\mathbf{A}, \mathbf{W}, \mathbf{H}, \mathbf{\Sigma}_\mathbf{b}\}$ peuvent être estimés au sens du maximum de vraisemblance à l'aide d'un algorithme d'Espérance-Maximisation. Des règles multiplicatives semblables à celles décrites aux équations (2.19) et (2.20) peuvent être également dérivées dans le cas linéaire instantané.

Par extension du modèle génératif gaussien monocanal, ce modèle permet d'exploiter l'information spatiale encodée par la matrice de mélange $\mathbf{A}$ afin d'implicitement grouper les composantes de la NMF qui forment les J sources. Pour revenir à notre objectif de séparation de sources défini à l'équation (2.3), les coefficients TFCT des images des sources estimées peuvent être reconstruites par filtrage de Wiener multicanal

$$\tilde{\mathbf{s}}_{j,fn} = \frac{v_{j,fn}\mathbf{R}_{j,f}}{\sum_{j=1}^{J} v_{j,fn}\mathbf{R}_{j,f}} \mathbf{x}_{fn}, \tag{2.35}$$

où la matrice $\mathbf{R}_{j,f} = \mathbf{A}_{j,f}\mathbf{A}_{j,f}^H$ de taille $I \times I$ est nommée matrice de *covariance spatiale*.

Autres structures de NMF

La proposition originelle de la NMF [Lee et Seung, 1999] a donné lieu à de nombreuses propositions visant à adapter le principe aux spécificités des signaux audio. En particulier, une structure de factorisation étendue a été proposé dans [Ozerov et al., 2012] à partir de la NMF multicanale [Ozerov et Févotte, 2010]. Plutôt que de factoriser la variance $\mathbf{V}_j = \{v_{j,fn}\}_{n=1..N}^{f=1..F}$ d'une source par NMF standard, il est plutôt proposé de factoriser la variance selon

$$\mathbf{V}_j = \mathbf{V}_j^{\mathsf{ex}} \circ \mathbf{V}_j^{\mathsf{ft}} = \left(\mathbf{W}_j^{\mathsf{ex}}\mathbf{U}_j^{\mathsf{ex}}\mathbf{G}_j^{\mathsf{ex}}\mathbf{H}_j^{\mathsf{ex}}\right) \circ \left(\mathbf{W}_j^{\mathsf{ft}}\mathbf{U}_j^{\mathsf{ft}}\mathbf{G}_j^{\mathsf{ft}}\mathbf{H}_j^{\mathsf{ft}}\right), \tag{2.36}$$

où toutes les matrices sont non-négatives. Cette structure communément appelée *excitation-filtre* avait auparavant été proposée dans [FitzGerald et al., 2008] pour la factorisation en tenseur non-négatif et dans une version simplifiée dans [Durrieu et al., 2009] pour la modélisation de sources de voix chantée. Nous exploiterons ce modèle et en donnerons plus de détails dans les parties 2.2.1 et 2.2.2.

2.1.4 Masquage temps-fréquence

Le masquage temps-fréquence est un principe très utilisé en séparation de sources. En effet, de nombreuses méthodes de séparation, qu'elles soient spectrales ou spatiales, ne modélisent pas la phase des sources. C'est le cas notamment de la NMF déterministe qui ne modélise que les spectrogrammes des sources. Pourtant, afin d'atteindre l'objectif de la séparation et reconstruire les sources estimées dans le domaine temporel, il est indispensable de donner une information de phase à chacune des sources. Pour cela, la plupart des méthodes que nous avons jusqu'à présent citées utilisent le principe de masquage temps-fréquence qui a pour finalité d'attribuer à chaque source estimée la même phase que le mélange.

Le principe du masquage temps-fréquence est simple. Le coefficient $\tilde{s}_{ji,fn}$ de la TFCT relative au $i^{\text{ième}}$ canal de la $j^{\text{ième}}$ source estimée est exprimé comme une portion $\tilde{\xi}_{ji,fn} \in [0,1]$ de la TFCT du mélange $x_{i,fn}$ sur le même canal selon

$$\tilde{s}_{ji,fn} = \tilde{\xi}_{ji,fn}\, x_{i,fn}. \tag{2.37}$$

En notant $\tilde{\mathbf{S}}_j = \{\tilde{\mathbf{s}}_{j,fn}\}_{n=1..N}^{f=1..F}$ la TFCT estimée pour la $j^{\text{ième}}$ source, $\mathbf{X} = \{\mathbf{x}_{fn}\}_{n=1..N}^{f=1..F}$ la TFCT du mélange et $\tilde{\mathbf{\Xi}}_j$ le tenseur formé des vecteurs $\tilde{\boldsymbol{\xi}}_{j,fn} = [\xi_{j1,fn}, ..., \xi_{ji,fn}, ..., \xi_{jI,fn}]^\intercal$ pour l'ensemble des fréquences f et trames n, le principe de masquage temps-fréquence peut s'écrire

$$\tilde{\mathbf{S}}_j = \tilde{\mathbf{\Xi}}_j \circ \mathbf{X}. \tag{2.38}$$

Le tenseur $\tilde{\mathbf{\Xi}}_j$ est couramment appelé *masque temps-fréquence*. Lorsque ses coefficients sont réels et compris entre 0 et 1, on parle de *masquage doux*. Lorsque ses coefficients sont contraints à être égaux à 1 ou 0, on parle alors de *masquage binaire*. Cette notion de masquage binaire permet de comprendre l'utilisation du terme *masquage*. En effet, le masque $\tilde{\mathbf{\Xi}}_j$, de même dimension que la TFCT du mélange $\mathbf{X}$, permet d'exprimer la TFCT de la $j^{\text{ième}}$ source en *masquant* les points temps-fréquence (f,n) tels que $\xi_{ji,fn} = 0$ et ne conservant que les points temps-fréquence (f,n) où $\xi_{ji,fn} = 1$. Dans le cas d'un masquage doux, les points temps-fréquence ne sont donc que partiellement masqués.

Une fois la TFCT d'une source estimée par masquage, la source estimée $\tilde{\mathbf{s}}_j(t)$ dans le domaine temporel peut être obtenue par simple transformée inverse de la TFCT estimée $\tilde{\mathbf{S}}_j$. Étant donné le modèle de mélange linéaire (2.2) considéré dans notre étude, nous noterons que nécessairement

$$\forall i, \quad \forall f,n, \quad \sum_{j=1}^{J} \xi_{ji,fn} = 1. \tag{2.39}$$

Enfin, nous remarquerons en comparant la définition (2.38) aux équations (2.24) et (2.35) que la NMF monocanale introduite dans la partie 2.1.3 utilise ce principe de masquage temps-fréquence pour estimer les TFCTs des sources et les signaux temporels associés. Dans ce cas particulier, le masquage est équivalent à une opération dite de *filtrage de Wiener* [Wiener, 1949].

2.1.5 Évaluation de la qualité de séparation

Nous avons défini l'objectif de la séparation de sources dans la partie 2.1.1 comme étant l'estimation des J signaux $\mathbf{s}_j(t)$ multicanaux qui composent un mélange $\mathbf{x}(t)$ lui aussi multicanal

défini par le modèle de mélange linéaire (2.2). Afin de développer des algorithmes de séparation et, dans cette thèse, afin d'évaluer les méthodes de fusion que nous allons introduire dans les prochains chapitres, il est nécessaire de pouvoir quantifier à l'aide de mesures la qualité de la séparation.

Les mesures utilisées en séparation de sources peuvent être réparties en deux catégories principales : les *mesures objectives* et les mesures *subjectives*. Qu'elles soient objectives ou subjectives, les mesures proposées dans la littérature cherchent souvent à refléter la qualité de la séparation au moyen d'un ou plusieurs scores numériques. Dans l'idéal, pour notre application de séparation de sources, les mesures subjectives semblent les plus appropriées. En effet, celles-ci reposent sur la quantification de la qualité de séparation par un ou plusieurs auditeurs. Elles sont donc parfaitement corrélées à notre objectif. Toutefois, ces mesures sont souvent difficiles et coûteuses à mettre en place (pensons, par exemple, à l'organisation d'une campagne d'évaluation).

Les mesures objectives sont elles beaucoup plus simples à mettre à place car elles ne reposent que sur le calcul systématique de quantités dépendant principalement des signaux de référence et des signaux estimés. Parmi elles, certaines mesures dites *perceptives* tentent de prendre en compte les caractéristiques du système auditif humain, soit par modélisation de ce dernier comme dans le logiciel *PEMO-Q* [Huber et Kollmeier, 2006] ou dans le standard *PEAQ* [Thiede et al., 2000], soit par modélisation du lien entre mesures objectives et mesures subjectives [Fox et Pardo, 2007]. Les mesures *PEASS* introduites dans [Emiya et al., 2011] et améliorées dans [Vincent, 2012] combinent par ailleurs ces deux principes. Enfin, d'autres mesures moins générales peuvent être également utilisées dans des applications particulières de la séparation de sources. C'est le cas par exemple de la mesure *PESQ* [Rix et al., 2001] qui est dédiée à l'évaluation perceptive de la qualité de la parole.

Pour notre travail, nous avons choisi les mesures initialement proposées dans [Vincent et al., 2006] pour l'évaluation de la séparation de sources dans le cas monocanal et étendues aux mélanges multicanaux dans [Vincent et al., 2007]. Ces mesures objectives sont en effet les plus employées en séparation de sources et ont l'avantage d'être rapides à calculer, et ce grâce à la mise à disposition d'une implémentation en *MATLAB*. Nous en présentons ci-après le principe.

Suivant le modèle de mélange linéaire (2.2), les mesures objectives que nous allons considérer sont construites à partir de la connaissance des vraies images spatiales des sources $\mathbf{s}_j(t)$ qui composent le mélange $\mathbf{x}(t)$. Pour une source donnée, il est proposé d'expliquer la distorsion globale entre l'estimée $\tilde{\mathbf{s}}_j(t)$ de l'image spatiale de cette source et son image spatiale vraie $\mathbf{s}_j(t)$ comme étant le résultat de trois termes de distorsion différents selon

$$\tilde{\mathbf{s}}_j(t) - \mathbf{s}_j(t) = \mathbf{e}_j^{\text{spat}}(t) + \mathbf{e}_j^{\text{interf}}(t) + \mathbf{e}_j^{\text{artef}}(t). \tag{2.40}$$

Parmi ces termes, le terme $\mathbf{e}_j^{\text{interf}}(t)$ vise à expliquer les *interférences*, c'est-à-dire la distorsion due à la possible présence d'une partie des autres sources du mélange ($\mathbf{s}_{j' \neq j}(t)$) dans l'image spatiale $\tilde{\mathbf{s}}_j(t)$ estimée. Le terme $\mathbf{e}_j^{\text{artef}}(t)$ tend lui à expliquer les *artefacts*, c'est-à-dire la distorsion créée par la méthode de séparation utilisée pour estimer $\tilde{\mathbf{s}}_j(t)$. Contrairement aux interférences, les artefacts ne proviennent pas des autres sources et ne font aucunement partie du mélange originel. Dans la littérature de séparation de sources, il est souvent fait référence aux artefacts par le terme de *bruit musical*. Enfin, le terme $\mathbf{e}_j^{\text{spat}}(t)$ représente lui les déformations relatives à l'image spatiale vraie et dues à des effets de filtrage ou de distorsions spatiales. La décomposition (2.40) est obtenue par projection orthogonale sur des sous-espaces selon la méthode détaillée dans [Vincent et al., 2006].

Une fois les différents termes de distorsions estimés, trois grandeurs exprimées en décibels (dB) sont proposées afin de mesurer respectivement la quantité relative de distorsion spatiale (nommée *ISR* pour *Image to Spatial distortion Ratio*), d'interférences (nommée *SIR* pour *Source*

to Interference Ratio) et d'artefacts (nommée *SAR* pour *Sources to Artifacts Ratio*) :

$$\mathrm{ISR}\left[\tilde{s}_j\right] = 10\log_{10}\frac{\sum_t \|\mathbf{s}_j(t)\|^2}{\sum_t \|\mathbf{e}_j^{\mathsf{spat}}(t)\|^2}, \tag{2.41}$$

$$\mathrm{SIR}\left[\tilde{s}_j\right] = 10\log_{10}\frac{\sum_t \|\mathbf{s}_j(t)+\mathbf{e}_j^{\mathsf{spat}}(t)\|^2}{\sum_t \|\mathbf{e}_j^{\mathsf{interf}}(t)\|^2}, \tag{2.42}$$

$$\mathrm{SAR}\left[\tilde{s}_j\right] = 10\log_{10}\frac{\sum_t \|\mathbf{s}_j(t)+\mathbf{e}_j^{\mathsf{spat}}(t)+\mathbf{e}_j^{\mathsf{interf}}(t)\|^2}{\sum_t \|\mathbf{e}_j^{\mathsf{artef}}(t)\|^2}. \tag{2.43}$$

La notation $\|\mathbf{s}_j(t)\|^2$ désigne le carré de la norme euclidienne du vecteur $\mathbf{s}_j(t)$ formé des I scalaires $s_{ji}(t)$ relatifs aux I canaux du mélange. La qualité globale est elle mesurée par le SDR (pour *Signal to Distortion Ratio*)

$$\mathrm{SDR}\left[\tilde{s}_j\right] = 10\log_{10}\frac{\sum_t \|\mathbf{s}_j(t)\|^2}{\sum_t \|\mathbf{s}_j(t)-\tilde{\mathbf{s}}_j(t)\|^2}. \tag{2.44}$$

Comme nous l'avons déjà évoqué précédemment, nous évaluerons tout au long de ce manuscrit nos méthodes de fusion grâce à ces quatre mesures. De plus, nous verrons plus tard que la simplicité de calcul du SDR nous permettra de mettre en œuvre des méthodes de fusion basées sur l'optimisation de cette mesure.

2.2 Applications spécifiques et modèles dédiés

Nous proposons dans ce manuscrit d'évaluer nos méthodes de fusion sur deux applications particulières de séparation de sources, à savoir le rehaussement de la parole et l'extraction de voix chantée. Dans cette partie, nous proposons ici un bref état de l'art spécifique à chacune de ces applications ainsi qu'un aperçu plus détaillé des modèles que nous avons retenu pour la mise en pratique de nos méthodes de fusion. La partie 2.2.1 sera consacrée au rehaussement de la parole et la partie 2.2.2 à l'extraction de voix chantée.

2.2.1 Application 1 : rehaussement de la parole

Le *rehaussement de la parole*, consiste en l'estimation d'un signal de parole, que nous noterons $\mathbf{s}_1(t)$, noyé dans un bruit noté $\mathbf{s}_2(t)$ à partir de la seule observation du mélange des deux défini selon le modèle de mélange linéaire (2.2)

$$\forall t, \quad \mathbf{x}(t) = \mathbf{s}_1(t) + \mathbf{s}_2(t). \tag{2.45}$$

Le rehaussement de la parole fut l'un des premiers problèmes de séparation de sources à être étudié. La littérature est donc abondante et les méthodes peuvent être regroupées en deux classes [Benesty et al., 2005] : les méthodes multicanales et les méthodes monocanales. Comme nous l'avons déjà évoqué dans un cadre plus général, les méthodes multicanales cherchent à exploiter la diversité spatiale des sources afin de les séparer par formation de voies (*beamforming* en anglais) [Gannot et al., 2001] ou filtrage spatio-temporel [Doclo et Moonen, 2002] par exemple. Les méthodes monocanales, à l'inverse, ne peuvent tirer profit d'informations spatiales pour distinguer les sources. On peut alors les regrouper en trois catégories principales [Loizou, 2013] : les approches par soustraction spectrale [Boll, 1979], les approches par sous-espaces [Ephraim et Van Trees, 1995] et les approches par modélisation statistique [Ephraim, 1992]. Parmi ces dernières, de nombreux modèles cherchent à caractériser le contenu spectral des sources de parole et de bruit afin de les distinguer. Pour ce faire, il peut être fait appel notamment

aux modèles de mélanges de gaussiennes (GMMs) [Hao et al., 2009], aux méthodes basées sur l'exemple (*examplar-based methods* en anglais) [Gemmeke et al., 2013] ou aux méthodes basées sur des codebooks (*codebook-driven techniques* en anglais) [Mowlaee et al., 2012; Srinivasan et al., 2006].

Dans notre travail, nous nous limiterons à l'étude du rehaussement de la parole monocanal. Pour ce faire, nous utiliserons l'un des modèles de source les plus populaires [Geiger et al., 2013; Mohammadiha et al., 2012; Moritz et al., 2013; Raj et al., 2011] : la NMF, que nous avons déjà introduite dans la partie 2.1.3 et qui, de plus, a atteint de très bonnes performances sur le corpus CHiME [Geiger et al., 2013; Vincent et al., 2013a] que nous introduirons dans la partie 3.3.2. Le principe consiste à modéliser la source de bruit $s_2(t)$ et la source de parole $s_1(t)$ par deux NMFs distinctes. Nous adopterons le formalisme du modèle génératif gaussien introduit dans la partie 2.1.3 et plus largement étudié dans [Ozerov et al., 2012].

Modèle de bruit

Nous proposons de modéliser la source de bruit par une NMF simple. La source $s_{2,fn}$ est donc supposée distribuée selon une loi normale univariée

$$s_{2,fn} \sim \mathcal{N}\left(0, v_{2,fn}\right) \tag{2.46}$$

dont le terme de variance $v_{2,fn}$ est le résultat d'une NMF d'ordre K_2 tel que

$$v_{2,fn} = \sum_{k=1}^{K_2} w_{2,fk} h_{2,kn}. \tag{2.47}$$

Nous proposerons dans la partie 3.3.3 d'apprendre le dictionnaire du bruit $\mathbf{W}_2$ sur des parties du mélange $x(t)$ ne comportant que du bruit et pas de parole.

Modèle de parole

Nous proposons également de modéliser la source de parole par NMF. Dans le plan temps-fréquence, la source $s_{1,fn}$ est donc supposée distribuée selon une loi normale univariée

$$s_{1,fn} \sim \mathcal{N}\left(0, v_{1,fn}\right). \tag{2.48}$$

Cette fois, nous proposons d'étudier deux types de structure de NMF pour la variance $v_{1,fn}$: une structure de type NMF simple, similaire à celle proposée pour le modèle de bruit, et une structure de type excitation-filtre, initialement employée dans [Durrieu et al., 2009].

Structure de type NMF simple Dans un premier temps, nous proposons donc que le terme de variance $v_{1,fn}$ soit modélisé par une NMF simple

$$v_{1,fn} = \sum_{k=1}^{K_1} w_{1,fk} h_{1,kn}. \tag{2.49}$$

Dans ce cas, les spectres caractéristiques de chacun des locuteurs seront appris à partir de signaux de parole non-bruités afin d'initialiser les coefficients du dictionnaire $w_{1,fk}$ ainsi que les coefficients d'activation $h_{1,kn}$. Nous détaillerons cette procédure d'apprentissage dans la partie 3.3.4.

Structure de type excitation-filtre Nous étudierons également l'opportunité d'utiliser une autre forme de factorisation pour modéliser les caractéristiques spectrales de la source de parole, à savoir la forme Excitation-Filtre (EF), parfois nommée source-filtre. Pour rappel, la forme excitation-filtre [FitzGerald et al., 2008; Ozerov et al., 2012] permet de décomposer la variance $\mathbf{V}_1 = \{v_{1,fn}\}_{f=1..F}^{n=1..N}$ d'une source en un produit terme à terme de deux autres matrices non-négatives $\mathbf{V}_1^{\mathsf{ex}} = \{v_{1,fn}^{\mathsf{ex}}\}_{f=1..F}^{n=1..N}$ et $\mathbf{V}_1^{\mathsf{ft}} = \{v_{1,fn}^{\mathsf{ft}}\}_{f=1..F}^{n=1..N}$ dénotant respectivement la partie excitation et la partie filtre de la factorisation. Comme proposé dans [Durrieu et al., 2009, 2010], cette forme de factorisation se prête bien à la modélisation de la voix. Dans ce cas, en effet, le terme $\mathbf{V}_1^{\mathsf{ex}}$ est voué à modéliser le signal glottique généré par les cordes vocales et responsable de la hauteur de la voix (*pitch* en anglais). Le terme $\mathbf{V}_1^{\mathsf{ft}}$ se charge lui de modéliser les effets du conduit vocal sur ce signal glottique, c'est-à-dire les formants caratéristiques de la voix. Ainsi, la variance totale de la source de parole s'exprime comme

$$v_{1,fn} = v_{1,fn}^{\mathsf{ex}} v_{1,fn}^{\mathsf{ft}} = \left(\sum_{k=1}^{K_1^{\mathsf{ex}}} w_{1,fk}^{\mathsf{ex}} h_{1,kn}^{\mathsf{ex}}\right)\left(\sum_{k=1}^{K_1^{\mathsf{ft}}} w_{1,fk}^{\mathsf{ft}} h_{1,kn}^{\mathsf{ft}}\right). \tag{2.50}$$

Le dictionnaire $\mathbf{W}_1^{\mathsf{ex}}$ est composé de peignes harmoniques pour plusieurs fréquences fondamentales f_0. Les amplitudes de ces spectres harmoniques sont fixées par le modèle de source glottique *KLGLOTT88* proposé dans [Klatt et Klatt, 1990], comme illustré sur la figure 2.3. Les fréquences fondamentales sont choisies pour couvrir la tessiture moyenne d'un locuteur, soit $f_0 \in [50\text{ Hz}, 400\text{ Hz}]$. Cet intervalle est discrétisé à raison d'un spectre harmonique tous les 1 Hz. Le dictionnaire est complété par une colonne représentant un bruit blanc, c'est-à-dire une colonne de 1, afin de modéliser les sons non-voisés. Similairement au cas de la NMF simple, nous proposerons dans la partie 3.3.4 une méthode d'apprentissage des valeurs d'initialisation de la matrice d'activation associée $\mathbf{H}_1^{\mathsf{ex}}$.

Le dictionnaire de la partie filtre $\mathbf{W}_1^{\mathsf{ft}}$ est lui-même factorisé en un produit de deux matrices non-négatives de sorte que $\mathbf{W}_1^{\mathsf{ft}} = \mathbf{B}_1^{\mathsf{ft}}\mathbf{U}_1^{\mathsf{ft}}$. Cette décomposition supplémentaire vise à imposer une contrainte de lissage sur les filtres du modèle de voix. Pour cela, le dictionnaire $\mathbf{B}_1^{\mathsf{ft}}$ est composé de spectres à bande étroite. Ainsi, les activations associées $\mathbf{U}_1^{\mathsf{ft}}$ permettent de décrire les filtres de la voix $\mathbf{W}_1^{\mathsf{ft}}$ comme une combinaison de spectres à bande étroite, les contraignant ainsi à former des filtres lisses. De même que pour la matrice d'activation $\mathbf{H}_1^{\mathsf{ex}}$ de la partie excitation, nous proposerons dans la partie 3.3.4 d'apprendre les valeurs d'initialisation des matrices d'activation $\mathbf{U}_1^{\mathsf{ft}}$ et $\mathbf{H}_1^{\mathsf{ft}}$.

2.2.2 Application 2 : extraction de voix chantée

Dans la suite de ce travail, nous évaluerons également nos méthodes de fusion sur une tâche d'*extraction de voix chantée*. Le problème d'extraction de voix chantée consiste à séparer un signal de voix que nous noterons $\mathbf{s}_1(t)$ d'un signal d'accompagnement musical $\mathbf{s}_2(t)$ à partir de l'observation de leur mélange $\mathbf{x}(t)$. Comme pour le problème de rehaussement de parole, certaines méthodes proposées dans la littérature s'appuient sur l'estimation de la position spatiale de la source de voix pour la séparer de son accompagnement. C'est le cas notamment de la technique DUET (*Degenerate Unmixing Estimation Technique*) [Jourjine et al., 2000; Rickard et Yilmaz, 2002].

Même si dans les enregistrements stéréophoniques de musique actuelle la voix est souvent centrée, les approches uniquement basées sur des critères spatiaux peinent à estimer le signal de voix chantée car de nombreux effets, dont des effets de réverbération, sont souvent ajoutés en cours de production. D'autres méthodes s'attachent donc à modéliser directement le signal d'accompagnement musical [Han et Chen, 2011; Liutkus et al., 2012; Rafii et Pardo, 2012]

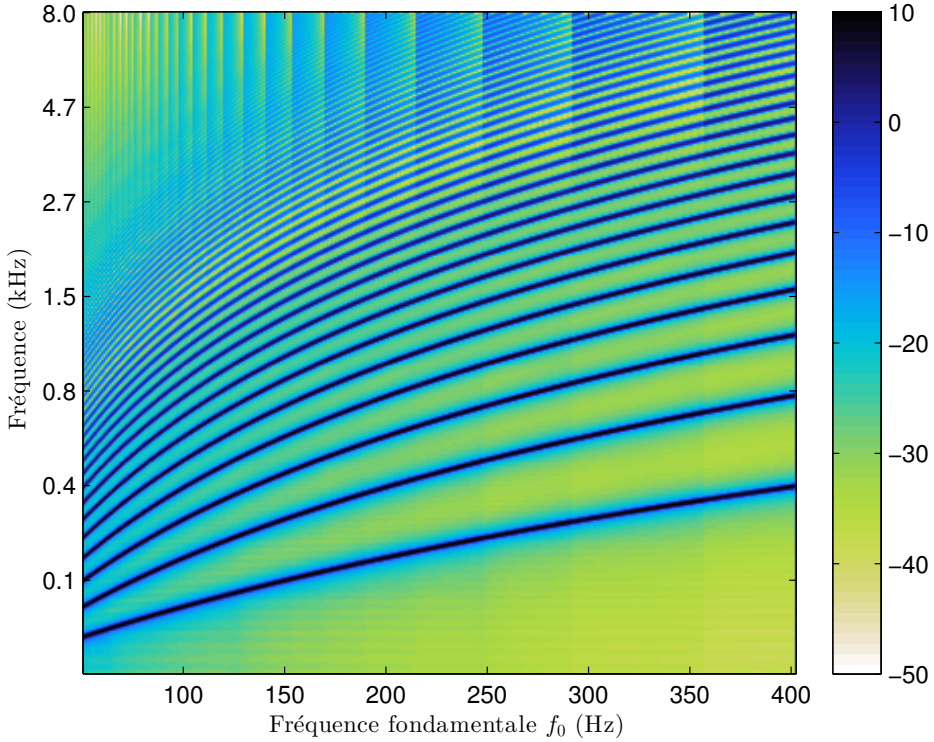


FIGURE 2.3 – Dictionnaire $\mathbf{W}_1^{\text{ex}}$ composé de spectres harmoniques pour différentes fréquences fondamentales f_0.

ou le signal de voix [HSU et JANG, 2010; LI et WANG, 2007]. D'autres méthodes encore combinent un modèle de voix chantée et un modèle de musique [DURRIEU et al., 2011; HUANG et al., 2012; LIUTKUS et al., 2014; VIRTANEN et al., 2008b].

Pour notre étude, nous avons sélectionné parmi elles quatre méthodes distinctes que nous proposons d'appliquer au problème d'extraction de voix chantée afin d'évaluer nos méthodes de fusion. Nous présentons ci-après leur principe et nous étudierons dans le chapitre 3 leurs performances individuelles.

Modèle de mélange instantané (IMM)

Le *modèle de mélange instantané* (*IMM* pour *Instantaneous Mixture Model*) a été proposé initialement dans [DURRIEU et al., 2011]. Comme son nom l'indique, le spectrogramme de puissance du mélange $|\mathbf{X}|^{.2} = \left\{|\mathbf{x}_{fn}|^{.2}\right\}_{f=1..F}^{n=1..N}$ est supposé être la somme des spectrogrammes de puissance de la voix chantée $|\mathbf{S}_1|^{.2}$ et de l'accompagnement musical $|\mathbf{S}_2|^{.2}$. Ce modèle est en fait similaire à celui qui a été présenté dans la partie 2.2.1 pour le problème de rehaussement de parole. La principale différence est qu'il ne fait pas appel à la formulation probabiliste de la NMF mais se base sur sa formulation déterministe originelle [LEE et SEUNG, 2001].

En effet, ici, l'accompagnement musical est modélisé tel que son spectrogramme de puissance $|\mathbf{S}_2|^{.2}$ est le résultat d'une NMF simple :

$$|\mathbf{S}_2|^{.2} = \mathbf{W}_2\mathbf{H}_2 \tag{2.51}$$

où $\mathbf{W}_2$ désigne le dictionnaire de formes spectrales décrivant la source de musique et $\mathbf{H}_2$ la matrice d'activation associée.

Le spectrogramme de puissance de la voix chantée $|\mathbf{S}_1|^{\cdot 2}$ est lui exprimé par une factorisation de type excitation-filtre telle que :

$$|\mathbf{S}_1|^{\cdot 2} = (\mathbf{W}_1^{\text{ex}}\mathbf{H}_1^{\text{ex}}) \circ \left(\mathbf{B}_1^{\text{ft}}\mathbf{U}_1^{\text{ft}}\mathbf{H}_1^{\text{ft}}\right) \tag{2.52}$$

où le terme $\mathbf{W}_1^{\text{ex}}$ est un dictionnaire de peignes harmoniques initialisés grâce au modèle *KLGLOTT88* [Klatt et Klatt, 1990], similaire à celui présenté à la figure 2.3, $\mathbf{H}_1^{\text{ex}}$ représente la matrice d'activation associée, le produit $\mathbf{W}_1^{\text{ft}} = \mathbf{B}_1^{\text{ft}}\mathbf{U}_1^{\text{ft}}$ représente un dictionnaire de formes spectrales lisses modélisant les filtres du conduit vocal et la matrice $\mathbf{H}_1^{\text{ft}}$ représente la matrice d'activation associée.

A la différence du modèle utilisé pour le rehaussement de la parole dans la partie 2.2.1, le processus de séparation n'inclut pas une phase d'apprentissage de modèle de voix spécifique. L'intégralité des matrices, exceptées $\mathbf{B}_1^{\text{ft}}$ et $\mathbf{W}_1^{\text{ex}}$, seront donc mises à jour directement à partir du mélange. De surcroît, l'estimation des paramètres est cette fois réalisée en trois temps :

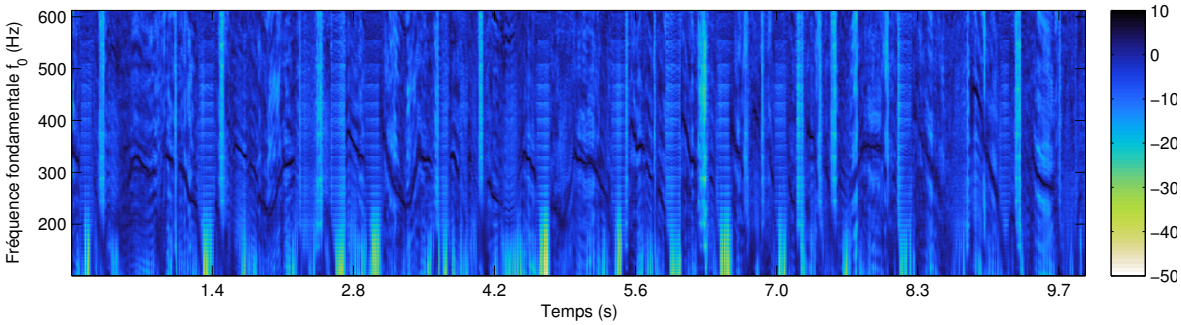
1. Lors de la première passe, les matrices du modèle, à l'exception des dictionnaires $\mathbf{B}_1^{\text{ft}}$ et $\mathbf{W}_1^{\text{ex}}$ qui sont fixés, sont initialisées de manière aléatoire puis estimées par mises à jour multiplicatives selon les règles introduites dans la partie 2.1.3. Suite à cette première estimation, la matrice d'activation des peignes harmoniques $\mathbf{H}_1^{\text{ex}}$ est affinée de sorte que seule une activation par trame temporelle est retenue pour les passes suivantes. Pour ce faire, il est proposé d'utiliser un algorithme de Viterbi, comme illustré à la figure 2.4. La matrice $\widetilde{\mathbf{H}}_1^{\text{ex}}$ ainsi obtenue peut être interprétée comme représentant la ligne mélodique chantée par la voix. L'emploi de l'algorithme de Viterbi permet à ce titre de favoriser la continuité de cette ligne mélodique.
2. Pour la deuxième passe, la matrice $\mathbf{H}_1^{\text{ex}}$ est alors initialisée par sa version $\widetilde{\mathbf{H}}_1^{\text{ex}}$ affinée par Viterbi. Les autres matrices du modèle, sauf $\mathbf{B}_1^{\text{ft}}$ et $\mathbf{W}_1^{\text{ex}}$, sont réinitialisées aléatoirement et réestimées grâce aux règles de mises à jour multiplicatives.
3. Enfin, lors de la troisième passe, une colonne représentant un bruit blanc est ajoutée au dictionnaire de l'excitation $\mathbf{W}_1^{\text{ex}}$ afin d'estimer les contributions non-voisées du signal de voix chantée. Pour ce faire, toutes les matrices du modèle de voix sont fixées aux valeurs estimées à la passe précédente et seul le modèle d'accompagnement musical et la dernière ligne de $\mathbf{H}_1^{\text{ex}}$ correspondant au bruit blanc sont estimés par mises à jour multiplicatives.

Contrairement au rehaussement de la parole, le mélange considéré ici est stéréophonique. Il nous faut donc également estimer la position spatiale des sources de voix chantée et de musique. Pour cela, la NMF multicanale [Ozerov et Févotte, 2010] est employée, en supposant que le mélange est linéaire instantané ($\mathbf{A}_{fn} = \mathbf{A}$ dans l'équation (2.5)). Au final, les TFCTs estimées de la voix et de l'accompagnement musical sont obtenues par filtrage de Wiener et les signaux temporels correspondants sont obtenus par transformée inverse, selon le principe exposé dans la partie 2.1.4.

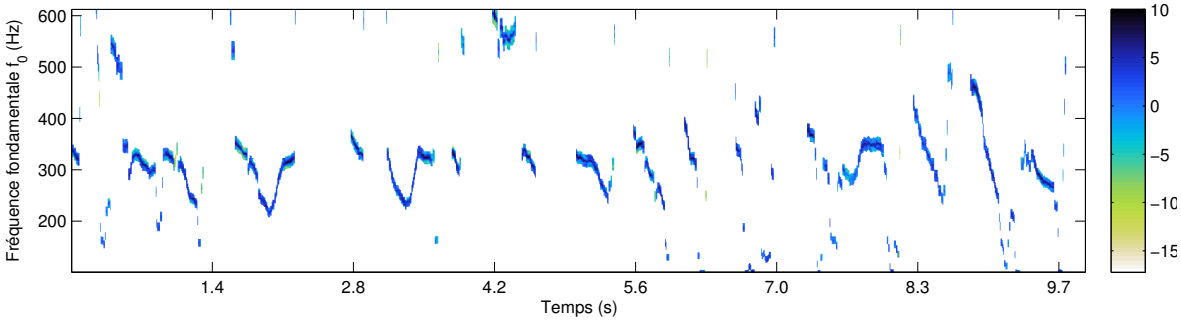
Analyse en composantes principales robuste (RPCA)

L'*analyse en composantes principales robuste* (*RPCA* pour *Robust Principal Component Analysis*) a été initialement proposée dans [Candès et al., 2011]. Son objectif est de décomposer une matrice réelle $\mathbf{M}$ comme la somme de deux matrices réelles de même taille, l'une, $\mathbf{L}$, étant de rang faible et l'autre, $\mathbf{P}$, étant parcimonieuse. Le problème peut être formulé comme un problème de minimisation tel que

$$\begin{array}{ll} \text{minimiser} & \|\mathbf{L}\|_* + \lambda\|\mathbf{P}\|_1 \\ \text{avec} & \mathbf{L} + \mathbf{P} = \mathbf{M}. \end{array} \tag{2.53}$$



(a) Avant Viterbi



(b) Après Viterbi

FIGURE 2.4 – Matrice d'activation $\mathbf{H}_1^{\text{ex}}$ de la partie excitation avant et après Viterbi. L'axe des ordonnées représente les fréquences fondamentales des peignes harmoniques du dictionnaire $\mathbf{W}_1^{\text{ex}}$ correspondant.

où $\|\mathbf{L}\|_*$ représente la norme nucléaire de la matrice $\mathbf{L}$, autrement dit la somme de ses valeurs singulières, et $\|\mathbf{P}\|_1$ représente la norme L_1 de la matrice $\mathbf{P}$, soit la somme des valeurs absolues de ses coefficients. Le coefficient λ est un hyperparamètre à choisir permettant de faire un compromis entre le rang de $\mathbf{L}$ et la parcimonie de $\mathbf{P}$ dans la décomposition de l'observation $\mathbf{M}$.

Dans [HUANG et al., 2012], l'auteur propose d'appliquer la RPCA au problème d'extraction de voix en se basant sur deux hypothèses destinées à discriminer la voix chantée de l'accompagnement musical. La première hypothèse est que la structure de l'accompagnement musical est par essence répétitive. L'auteur suggère donc que le spectrogramme de cet accompagnement musical est un modèle de rang faible. À l'inverse, et c'est là la deuxième hypothèse formulée, la voix chantée présente plus de variabilité mais est relativement parcimonieuse dans le plan temps-fréquence. Son spectrogramme est donc de rang plus élevé que celui de l'accompagnement musical. L'auteur propose donc de décomposer le spectrogramme d'amplitude de l'observation $\mathbf{M} \equiv |\mathbf{X}|$ en la somme d'une matrice de rang faible $\mathbf{L} \equiv |\mathbf{S}_2|$ décrivant le spectrogramme d'amplitude de l'accompagnement musical et d'une matrice $\mathbf{P} \equiv |\mathbf{S}_1|$ parcimonieuse décrivant le spectrogramme d'amplitude de la voix chantée. Des masques temps-fréquence binaires sont alors construits et appliqués à la TFCT du mélange $\mathbf{X}$ afin d'obtenir les TFCTs des signaux de voix $\mathbf{S}_1 = \{s_{1,fn}\}$ et de musique $\mathbf{S}_2 = \{s_{2,fn}\}$ qui sont ensuite inversés afin d'obtenir les signaux temporels estimés $\mathbf{s}_1(t)$ et $\mathbf{s}_2(t)$. Notons enfin que cette méthode ayant été développée pour des mélanges monocanaux, nous appliquerons dans le cas stéréo le même procédé indépendamment aux canaux droit et gauche du mélange original.

Extraction par similarité de motifs répétés (REPETsim)

La technique dite d'*extraction par similarité de motifs répétés* (*REPETsim* pour *Repeating Pattern Extraction Technique by Similarity*) proposée dans [RAFII et PARDO, 2012] est une

évolution de la technique d'extraction de motifs répétés (REPET) proposée initialement dans [Rafii et Pardo, 2011]. Comme son nom l'indique, cette technique propose d'identifier le contenu répétitif d'un mélange à l'aide d'une mesure de similarité entre les diverses trames temporelles de ce mélange et d'identifier ce contenu répétitif à l'accompagnement musical. Comme la méthode précédente, cette technique a été développée pour des mélanges monocanaux. Par conséquent, nous avons appliqué la procédure suivante successivement et indépendamment aux canaux droit et gauche du mélange original.

La première étape consiste donc à calculer la matrice de similarité $\mathbf{D} = \{d_{n_1 n_2}\}$ du mélange. En notant $|\mathbf{X}| = \{|x_{fn}|\}$ le spectrogramme d'amplitude d'un des canaux du mélange (droit ou gauche), la similarité entre les trames n_1 et n_2 est définie comme

$$d_{n_1 n_2} = \frac{\sum_f |x_{fn_1}||x_{fn_2}|}{\sqrt{\sum_f |x_{fn_1}|^2}\sqrt{\sum_f |x_{fn_2}|^2}}, \tag{2.54}$$

où f représente l'indice des fréquences.

Ensuite, pour chaque trame n du mélange, l'ensemble $\mathcal{N}_n$ des trames dont la similarité $d_{nn'}$ avec la trame n est supérieure à un seuil d_{seuil} est identifié. Seules les trames n' suffisamment proches de la trame n peuvent être retenues comme similaires et le nombre de trames pouvant être retenues comme similaires est également limité. Un modèle dit de *spectrogramme répétitif* $|\mathbf{S}_2| = \{|s_{2,fn}|\}$ est alors construit de telle sorte que

$$\forall f, n,\ |s_{2,fn}| = \underset{n' \in \mathcal{N}_n}{\text{médiane}} \left\{|x_{fn'}|\right\}. \tag{2.55}$$

Enfin, un masque temps-fréquence $\mathbf{\Xi}_2 = \{\xi_{2,fn}\}$ est déduit du modèle de spectrogramme répétitif $|\mathbf{S}_2|$ tel que

$$\forall f, n,\ \xi_{2,fn} = \min\left(\frac{|s_{2,fn}|}{|x_{fn}|}, 1\right). \tag{2.56}$$

Ce masque temps-fréquence est alors appliqué à la TFCT du mélange $\mathbf{X}$ afin d'estimer la TFCT du signal répétitif $\mathbf{S}_2$. Le signal temporel correspondant $s_2(t)$ est obtenu par transformée inverse et est identifié au signal d'accompagnement musical. Le signal de voix chantée $s_1(t)$ est quant à lui obtenu en soustrayant le signal d'accompagnement musical estimé du mélange initial $x(t)$.

Modèle additif à noyaux (KAM)

Le quatrième et dernier modèle que nous proposons d'utiliser est nommé *modèle additif à noyaux* (*KAM* pour *Kernel Additive Model*). Le papier original [Liutkus et al., 2014] propose de résoudre le problème de séparation de sources en posant des modèles paramétriques particuliers sur les sources qui composent un mélange. Ces modèles paramétriques reposent sur le concept de noyaux dits *de proximité*. Sans rentrer dans le détail de la théorie, l'application de ce modèle à notre problème d'extraction de voix chantée suffira à illustrer le rôle de ces noyaux.

De manière identique au modèle présenté dans la partie 2.1.3, l'approche par noyaux propose de modéliser chacune des sources $\mathbf{s}_j$ du problème par une loi normale centrée multivariée de sorte que, dans le plan temps-fréquence,

$$\forall f, n,\ \mathbf{s}_{j,fn} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{j,f} v_{j,fn}) \tag{2.57}$$

où $\mathbf{R}_{j,f}$ représente la matrice de covariance spatiale et $v_{j,fn}$ la densité spectrale de puissance de la source j au point temps-fréquence (f, n). Cette densité spectrale est modélisée à l'aide d'un noyau de proximité spécifique appelé noyau des k plus proches voisins [Stone, 1977]. C'est ce noyau, et en particulier sa forme, qui permet de caractériser une source et de la distinguer des autres.

En effet, un tel noyau permet de spécifier un ensemble de k points temps-fréquence $\mathcal{I}_j(f,n)$ pour lesquels la densité spectrale doit avoir une valeur proche de la densité spectrale $v_{j,fn}$ du point temps-fréquence considéré, d'où le nom de noyau de proximité. En termes mathématiques, nous avons :

$$\forall (f',n') \in \mathcal{I}_j(f,n), \quad v_{j,f'n'} \approx v_{j,fn}. \tag{2.58}$$

Pour notre problème, deux noyaux vont être définis, l'un pour la voix chantée, l'autre pour l'accompagnement musical. Pour l'accompagnement musical, le noyau choisi est composé de deux sous-noyaux distincts : l'un est un noyau périodique, comme illustré sur la figure 2.5(b), dont la période sera fixée par estimation du tempo du mélange, l'autre est un noyau horizontal, comme illustré sur la figure 2.5(a), de durée égale à 2 secondes permettant de modéliser les contributions harmoniques. La voix chantée est elle modélisée à l'aide de noyaux en forme de croix, comme illustré sur la figure 2.5(c). Une telle forme de noyau permet de modéliser des variations spectrales lentes bien adaptées à la modélisation de la voix. La dimension du noyau est arbitrairement fixée à 15 Hz en hauteur et 20 ms en largeur.

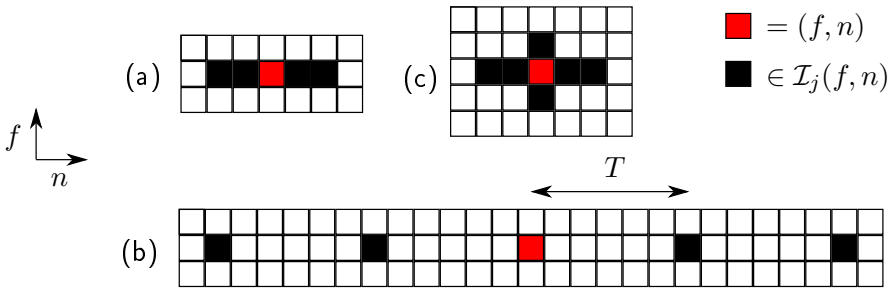


FIGURE 2.5 – Forme des noyaux de proximité choisis pour modéliser l'accompagnement musical (noyau horizontal (a) et noyau périodique (b)) et la voix chantée (noyau en croix (c)). Cette figure est une reproduction partielle d'une figure issue de la publication originale [LIUTKUS et al., 2014].

L'estimation des sources est menée grâce à un algorithme itératif nommé *kernel backfitting* dont nous ne donnerons pas le détail ici. Notons toutefois que, de façon similaire à l'approche REPETsim introduite préalablement, la densité spectrale $v_{j,fn}$ est finalement estimée comme la médiane des densités spectrales des k plus proches voisins définis par le noyau correspondant. Les sources sont alors reconstruites par filtrage de Wiener selon (2.24).

2.3 Fusion et sélection de modèles

Dans la partie précédente, nous avons dressé un état de l'art des méthodes de séparation de sources. Bien que non exhaustive, notre revue montre déjà la diversité des méthodes et applications de séparation de sources. Comme nous l'avons déjà abordé en introduction, lorsque nous sommes confrontés en pratique à un problème de séparation de sources, il est nécessaire en premier lieu de choisir une méthode de séparation et de l'adapter au problème posé. Ce choix est généralement guidé par l'expérience, ou par des considérations parfois décorrélées de l'objectif de séparation (choix par simplicité de la méthode, choix aléatoire, *etc.*). Par exemple, pour le cas du rehaussement de la parole introduit dans la partie 2.2.1, notre choix a été guidé par les bons résultats de la méthode retenue sur le corpus *CHiME*.

Une fois la méthode choisie, il convient ensuite de fixer les hyperparamètres de la méthode de séparation retenue. Là encore, l'expérience guide le plus souvent ce choix. Lorsqu'aucune expérience n'a été préalablement acquise, les méthodes de séparation possèdent souvent un paramètrage *par*

défaut. Toutefois, il est bien connu que le paramètrage d'un modèle a une influence non-négligeable sur la qualité de séparation. Par exemple, les performances du modèle NMF que nous avons retenu pour traiter le problème de rehaussement de la parole introduit dans la partie 2.2.1 dépendent particulièrement de l'ordre K_j choisi pour modéliser chaque source à séparer [Bertin et al., 2007]. Nous aurons d'ailleurs l'occasion dans la partie 3.3 d'observer ce phénomène.

Ce problème de choix d'un modèle et de son paramétrage a fait l'objet de nombreuses recherches. Certaines méthodes dites de *sélection de modèles* ont été développées afin de guider ce choix quel que soit le problème d'estimation posé. Nous présenterons certaines de ces méthodes dans la partie 2.3.1. Toutefois, il a également été proposé dans la littérature de ne pas forcément retenir un unique modèle et il a même été démontré que dans certains cas, la combinaison de plusieurs modèles permettait de dépasser les performances individuelles d'un unique modèle. Ce principe, que nous nommerons *fusion*, sera évoqué dans la partie 2.3.2. Enfin, les méthodes de sélection et de fusion ont souvent été développées pour la résolution d'un problème particulier. Aussi, nous consacrerons la dernière partie 2.3.3 de ce chapitre à une rapide revue des approches de sélection et de fusion pour la séparation de sources proposées dans la littérature.

2.3.1 Sélection de modèles : cadre théorique

La littérature relative aux méthodes de sélection de modèles est riche. Cependant, de nombreuses propositions sont dédiées à une application ou à un type d'algorithme d'estimation et leur généralisation à d'autres problèmes n'est pas souvent évidente. Nous proposons ici de passer en revue les principaux critères de sélection de modèles sans *a priori* sur le type de modèle utilisé. Les méthodes de sélection dédiées à la séparation de sources seront elles présentées dans la partie 2.3.3.

La sélection de modèles peut être formulée comme le problème de choisir parmi un ensemble de M modèles $\{\mathcal{M}_1, ..., \mathcal{M}_m, ..., \mathcal{M}_M\}$ le modèle qui explique le mieux des données $\mathbf{X}$ de dimension D, chaque modèle $\mathcal{M}_m$ étant défini par un ensemble de paramètres noté $\mathbf{Z}_m$ et de dimension d_m.

Méthodes par ré-échantillonnage

Dans le domaine de l'apprentissage automatique, les paramètres d'un modèle $\mathcal{M}_m$ sont préalablement estimés sur des données dites d'apprentissage $\mathbf{X}$ et le modèle a alors pour vocation d'être utilisé sur des données $\mathbf{X}'$ non-utilisées lors de l'apprentissage. Dans ce cas, le principe de sélection de modèles est analogue au principe de *généralisation*. En effet, on choisira le plus souvent en apprentissage le modèle qui *généralise* le mieux, c'est-à-dire celui qui aura les meilleurs performances sur des données $\mathbf{X}'$ différentes des données d'apprentissage, donc non-encore vues par le modèle. Ces données sont souvent qualifiées de données de *validation*. Lorsqu'un tel ensemble de données n'est pas disponible, il est possible de faire appel à des méthodes dites de *ré-échantillonnage*. Comme leur nom l'indique, ces méthodes visent à réexploiter les données d'apprentissage $\mathbf{X}$ afin de générer des données de validation. Parmi ces méthodes, on citera les techniques de *validation croisée* [Refaeilzadeh et al., 2009], très simples à mettre en œuvre, qui consistent simplement à scinder l'ensemble des données en deux ensembles distincts : l'un formant l'ensemble d'apprentissage et l'autre formant l'ensemble de validation destiné à mesurer la capacité de généralisation du modèle appris. Les méthodes dites de *bootstrap* [Efron, 1983; Efron et Tibshirani, 1994] sont elles des techniques de ré-échantillonnage statistiques. Le *bootstrap* consiste à simuler de nouvelles données $\mathbf{X}_b$ à partir des données $\mathbf{X}$ disponibles, par tirage aléatoire avec remise. L'idée du *bootstrap* est de renouveler cette simulation de données un grand nombre de fois et d'estimer la performance moyenne du modèle sur ces données. Quelle que soit la méthode de ré-échantillonnage employée, les modèles peuvent être comparés sur la base de leur performance moyenne sur les ensembles de données ré-échantillonnées.

Les méthodes par ré-échantillonnage ont toutefois l'inconvénient qu'elles nécessitent beaucoup de calculs coûteux en temps, puisque chaque modèle doit être évalué sur un grand nombre de données dans le cas du *bootstrap* et être entraîné sur un grand nombre de données dans le cas de la *validation croisée*. Pour éviter des calculs prohibitifs, une autre catégorie de méthodes de sélection de modèles fait appel à des critères purement statistiques.

Méthodes statistiques

Les méthodes statistiques de sélection proposent généralement de formuler le choix d'un modèle en fonction de critères souvent nommés *critères d'information* [Stoica et Selen, 2004]. Ces critères prennent généralement la forme suivante, pour le modèle $\mathcal{M}_m$,

$$\text{Crit}_m = \mathcal{E}_m + \mathcal{C}_m \tag{2.59}$$

où $\mathcal{E}_m$ rend compte de l'adéquation du modèle $\mathcal{M}_m$ aux données $\mathbf{X}$ (ou l'erreur de modélisation, en d'autres termes) et $\mathcal{C}_m$ mesure la complexité de ce même modèle. La sélection consiste alors à choisir le modèle dont le critère Crit_m est minimal. Afin de calculer $\mathcal{E}_m$ et $\mathcal{C}_m$, deux principaux paradigmes ont été employés dans la littérature. Le premier modélise le problème de sélection de modèles comme un problème statistique dans un cadre bayésien alors que le deuxième s'inspire de la théorie de l'information. Quel que soit le paradigme retenu, les critères de sélection de modèle obtenus sont donc le fruit d'un compromis entre adéquation du modèle aux données et complexité du modèle. Nous donnons ci-après un bref aperçu des principaux critères dérivés de ces deux paradigmes.

Approche bayésienne Le paradigme bayésien offre un cadre théorique très populaire à la sélection de modèles. Le principe de sélection par critère bayésien s'appuie sur le calcul de la probabilité *a posteriori* $p(\mathcal{M}_m|\mathbf{X})$ de chacun des modèles $\mathcal{M}_m$ sachant les données $\mathbf{X}$. Cette probabilité s'écrit, selon la règle de Bayes,

$$p(\mathcal{M}_m|\mathbf{X}) = \frac{p(\mathbf{X}|\mathcal{M}_m)\, p(\mathcal{M}_m)}{\sum_{m'=1}^{M} p(\mathbf{X}|\mathcal{M}_{m'})\, p(\mathcal{M}_{m'})}, \tag{2.60}$$

où $p(\mathcal{M}_m)$ et $p(\mathbf{X}|\mathcal{M}_m)$ désignent respectivement la probabilité *a priori* et la vraisemblance du modèle $\mathcal{M}_m$. Le critère de sélection bayésien, connu sous le nom de *règle du maximum a posteriori* consiste alors à sélectionner le modèle $\mathcal{M}_{m^*}$ dont la probabilité *a posteriori* $p(\mathcal{M}_m|\mathbf{X})$ est maximale, soit

$$m^* = \operatorname{argmax}\, p(\mathcal{M}_m|\mathbf{X}). \tag{2.61}$$

La littérature fait parfois référence à cette règle sous le terme de *comparaison de modèles bayésienne* [Bishop, 2006]. Dans le cas où la distribution des probabilités *a priori* des modèles $p(\mathcal{M}_m)$ est uniforme (lorsque $\forall m, m', p(\mathcal{M}_m) = p(\mathcal{M}_{m'})$), le critère bayésien revient à sélectionner le modèle $\mathcal{M}_m$ dont la *vraisemblance marginale* $p(\mathbf{X}|\mathcal{M}_m)$ est maximale.

La vraisemblance marginale est définie comme l'intégration de la probabilité jointe des données $\mathbf{X}$ et des paramètres du modèle $\mathbf{Z}_m$ par rapport à toutes les valeurs de ces paramètres selon

$$p(\mathbf{X}|\mathcal{M}_m) = \int p(\mathbf{X}, \mathbf{Z}_m|\mathcal{M}_m) d\mathbf{Z}_m = \int p(\mathbf{X}|\mathbf{Z}_m, \mathcal{M}_m) p(\mathbf{Z}_m|\mathcal{M}_m) d\mathbf{Z}_m \tag{2.62}$$

où $p(\mathbf{Z}_m|\mathcal{M}_m)$ représente la probabilité *a priori* des paramètres $\mathbf{Z}_m$ du modèle $\mathcal{M}_m$. Dans la littérature [Kass et Raftery, 1995], il est proposé de comparer les vraisemblances marginales de deux modèles $\mathcal{M}_m$ et $\mathcal{M}_{m'}$ en calculant leur rapport nommé *facteur de Bayes* et défini par

$$\mathcal{B}_{mm'} = \frac{p(\mathbf{X}|\mathcal{M}_m)}{p(\mathbf{X}|\mathcal{M}_{m'})}. \tag{2.63}$$

Si $\mathcal{B}_{mm'} > 1$, le modèle $\mathcal{M}_m$ représente mieux les données que le modèle $\mathcal{M}_{m'}$. Plus la valeur de $\mathcal{B}_{mm'}$ est élevée, plus la confiance en cette décision est grande.

Toutefois, en pratique, ni le critère bayésien ni le facteur de Bayes ne sont exploitables en l'état car le calcul de la vraisemblance marginale est souvent impossible. La vraisemblance marginale peut tout de même être approximée au moyen de méthodes d'inférence approchées, comme les méthodes de Monte-Carlo par chaînes de Markov [Andrieu et al., 1999] ou l'inférence variationnelle [Bishop, 2006] que nous aborderons dans le chapitre 5.

Une valeur approchée de la log-vraisemblance marginale peut être également obtenue en en effectuant un développement en série de Taylor du second ordre autour du point $\mathbf{Z}_m^\star$ défini comme étant l'estimateur du maximum de vraisemblance, soit

$$\mathbf{Z}_m^\star = \underset{\mathbf{Z}_m}{\operatorname{argmax}}\, p(\mathbf{X}, \mathbf{Z}_m|\mathcal{M}_m). \tag{2.64}$$

La démonstration est disponible dans [Stoica et Selen, 2004]. Une démonstration équivalente au moyen d'une approximation de Laplace est disponible dans [Lebarbier et Mary-Huard, 2006]. Dans ces deux cas toutefois, la log-vraisemblance marginale se trouve approchée par

$$\log p(\mathbf{X}, \mathbf{Z}_m|\mathcal{M}_m) \approx \log p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m) - \frac{d_m}{2}\log D + \log p(\mathcal{M}_m), \tag{2.65}$$

où, pour rappel, d_m représente le nombre de paramètres du modèle $\mathcal{M}_m$ (soit la longueur du vecteur de paramètres $\mathbf{Z}_m$) et D représente la dimension des données $\mathbf{X}$. La règle du maximum *a posteriori* (2.61) peut donc être approchée par la règle suivante :

$$m^* = \operatorname{argmax}\, \log p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m) - \frac{d_m}{2}\log D + \log p(\mathcal{M}_m). \tag{2.66}$$

Enfin, en supposant que la distribution des probabilités *a priori* des modèles $p(\mathcal{M}_m)$ est uniforme et en reformulant les termes de (2.66), l'approximation (2.65) permet de formuler le critère, connu sous le nom de *Bayesian Information Criterion* (*BIC*) et introduit par [Schwarz, 1978], qui s'écrit, pour le modèle $\mathcal{M}_m$,

$$\mathrm{BIC}_m = -2\log p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m) + d_m \log(D). \tag{2.67}$$

Le modèle sélectionné selon ce critère est alors le modèle qui a le critère BIC le plus petit, soit

$$m^* = \operatorname{argmin}\, \mathrm{BIC}_m. \tag{2.68}$$

Nous remarquerons que le critère BIC (2.67) est composée d'un terme d'adéquation aux données mesurée par la vraisemblance maximale $p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m)$ et d'un terme de pénalité de la complexité du modèle mesurée par le produit $d_m \log(D)$. Le critère BIC relève donc d'un compromis entre adéquation aux données et complexité.

Approche inspirée par la théorie de l'information En théorie de l'information, la dissimilarité entre deux distributions de probabilité est mesurée par la divergence de Kullback-Leibler que nous avons déjà introduite dans la partie 2.1.3. Ainsi, le modèle $\mathcal{M}_m$ qui approche au mieux la vraie distribution des données $p(\mathbf{X})$ doit être celui qui minimise la distance de KL entre la distribution vraie $p(\mathbf{X})$ et la vraisemblance $p(\mathbf{X}|\mathcal{M}_m)$ du modèle $\mathcal{M}_m$. Le meilleur modèle $\mathcal{M}_m^*$ est donc défini comme

$$m^* = \operatorname{argmin}\, \mathcal{D}_{\mathsf{KL}}(p(\mathbf{X})|p(\mathbf{X}|\mathcal{M}_m)) \tag{2.69}$$

avec

$$\mathcal{D}_{\mathsf{KL}}(p(\mathbf{X})|p(\mathbf{X}|\mathcal{M}_m)) = \int p(\mathbf{X}) \log \frac{p(\mathbf{X})}{p(\mathbf{X}|\mathcal{M}_m)} d\mathbf{X}. \tag{2.70}$$

De façon équivalente, le meilleur modèle $\mathcal{M}_m^*$ est celui qui maximise la quantité $I(p(\mathbf{X}), p(\mathbf{X}|\mathcal{M}_m))$ nommée *information de Kullback-Leibler* et définie comme

$$I(p(\mathbf{X}), p(\mathbf{X}|\mathcal{M}_m)) = \int p(\mathbf{X}) \log p(\mathbf{X}|\mathcal{M}_m) d\mathbf{X}. \tag{2.71}$$

Bien entendu, ni la divergence de KL ni l'information de KL ne peuvent être calculées puisque la distribution vraie des données $p(\mathbf{X})$ n'est pas connue. Au moyen d'un développement en série de Taylor du second ordre à proximité de l'estimateur du maximum de vraisemblance $\mathbf{Z}_m^\star$, similairement à l'approche bayésienne, il est montré notamment dans [Stoica et Selen, 2004] que l'information de KL (2.71) peut être approchée par

$$I(p(\mathbf{X}), p(\mathbf{X}|\mathcal{M}_m)) \approx \log p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m) - d_m. \tag{2.72}$$

La maximisation de ce critère relativement au modèle $\mathcal{M}_m$ permet donc de sélectionner le meilleur modèle au sens du maximum de l'information de KL et est équivalent à la minimisation du critère *AIC* (pour *Akaike Information Criterion*) [Akaike, 1992]

$$\text{AIC}_m = -2 \log p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m) + 2d_m. \tag{2.73}$$

Nous remarquerons que le critère AIC a une forme très similaire au critère BIC (2.67), en ce sens qu'il relève d'un compromis entre adéquation aux données exprimée par la vraisemblance maximale $p(\mathbf{X}|\mathbf{Z}_m^\star, \mathcal{M}_m)$ et la complexité du modèle qui cette fois est exprimée par un multiple du nombre de paramètres du modèle d_m, et ne dépendant donc pas, comme dans le cas bayésien, de la dimension des données D. D'autres critères reprennent cette idée de compromis entre adéquation et complexité d'un modèle, en changeant principalement l'expression du terme relatif à la complexité. On nommera par exemple le critère GIC (pour *Generalized Information Criterion*) [Stoica et Selen, 2004], le critère AIC corrigé [Hurvich et Tsai, 1993] ou le critère MDL (pour *Minimum Description Length*) [Rissanen, 1978].

2.3.2 Fusion : cadre théorique

Contrairement au principe de sélection, la fusion consiste à combiner plusieurs modèles afin de résoudre un problème donné, plutôt que de n'en choisir qu'un seul. Il est ainsi espéré que la solution obtenue par combinaison de divers modèles soit meilleure que celles données par chacun de ces modèles individuellement. La littérature sur ce sujet est abondante. De nombreux autres termes peuvent également faire référence au principe de fusion de modèles. Ainsi, certaines publications préfèrent les termes de *méthodes d'ensemble*, de *techniques d'agrégation*, de *mélange d'experts* ou encore le terme plus simple de *combinaison de modèles*.

La variété des termes employés pour désigner ce même principe de fusion fait écho à la diversité des champs de recherche et d'application qui ont déjà largement exploité ce principe, et ce depuis des décennies. Gardant en tête que notre objectif est d'appliquer ce principe à notre problème de séparation de sources, nous donnerons ici un aperçu des grands principes de la fusion, en restreignant notre étude aux problèmes connexes que sont la classification et la régression.

Dans la suite, nous supposerons donc que les M modèles $\mathcal{M}_m$ que nous cherchons à combiner sont soit des classifieurs, soit des régresseurs. Dans ces deux cas, les modèles $\mathcal{M}_m$ ont pour objectif de réaliser une prédiction à partir d'une observation $\mathbf{x}$. Pour un classifieur, l'objectif est de prédire la classe $\mathcal{C}_j$ à laquelle l'observation $\mathbf{x}$ appartient parmi un ensemble de J classes $\{\mathcal{C}_1, ..., \mathcal{C}_j, ..., \mathcal{C}_J\}$. Généralement, le classifieur $\mathcal{M}_m$ donne donc en sortie un vecteur $\mathbf{y}_m(\mathbf{x}) = (y_{m,1}, ..., y_{m,J})$ dont un seul coefficient $y_{m,j}$ est non-nul et égal à 1, indiquant que l'exemple $\mathbf{x}$ appartient à la classe $\mathcal{C}_j$. Pour un régresseur, l'objectif est simplement de prédire un vecteur de valeurs réelles $\mathbf{y}_m(\mathbf{x}) = (y_{m,1}, ..., y_{m,J})$.

Même si en pratique la fusion peut être envisagée à plusieurs niveaux, nous ne considérerons ici que le cas où la fusion a pour objectif de combiner les prédictions de M modèles. Dans la littérature, une telle fusion est qualifiée de *tardive* (par opposition à la fusion *précoce* qui, par exemple, combinerait plusieurs descripteurs en entrée des modèles). On parle également de *fusion des décisions*, par opposition à la *fusion de données*. Par conséquent, la fusion a pour objectif de formuler une nouvelle prédiction $\mathbf{y}(\mathbf{x})$ fonction des prédictions $\mathbf{y}_m(\mathbf{x})$ données par les M modèles.

Bien souvent, les problèmes de classification et de régression sont traités à l'aide de modèles appris de façon supervisée (c'est-à-dire que les paramètres du modèle sont estimés en fonction d'exemples d'apprentissage pour lesquels les prédictions $\mathbf{y}$ sont déjà connues). Par conséquent, la majorité des méthodes que nous allons ici évoquer sont des méthodes de fusion supervisée. Toutefois, nous voudrions souligner qu'il existe également des méthodes, plus récentes, de fusion non-supervisée. Généralement, ces méthodes ont été développées pour les problèmes de *partitionnement de données* (*clustering* en anglais), cas particuliers des problèmes de classification non-supervisée. De ce fait, ces méthodes sont souvent nommées *méthodes d'ensemble de partitionnement* (ou *clustering ensemble methods* en anglais). Une revue récente peut être trouvée dans [Vega-Pons et Ruiz-Shulcloper, 2011]. Par souci de concision et parce que nos travaux se sont majoritairement inspirés des approches de fusion supervisée pour la classification et la régression, nous n'évoquerons pas dans notre étude ces approches-là.

Parmi les méthodes de fusion tardive, on peut distinguer quatre principales classes d'approches [Bloch, 2003] :

1. les **méthodes par vote ou par moyennage** : il s'agit des méthodes les plus simples et les plus largement exploitées,
2. les **méthodes probabilistes et bayésiennes** : ici, les prédictions sont formulées à l'aide de distributions de probabilités et la fusion fait appel aux principes de l'inférence bayésienne,
3. les méthodes inspirées de la **théorie des croyances de Dempster-Shafer** [Shafer, 1976] : elles reposent sur la modélisation de l'imprécision et de l'incertitude des modèles à l'aide de fonctions de masse, de plausibilité et de croyance,
4. et les méthodes inspirées de la **théorie des ensembles flous et des possibilités** [Zimmermann, 2001] : ces approches offrent un cadre théorique pour représenter l'incertitude d'un modèle à partir de fonctions d'appartenance non-exclusive (par exemple, en classification, l'appartenance d'une observation à une classe n'est pas stricte mais définie selon un certain degré).

Parmi ces quatre classes, les méthodes par vote et les méthodes statistiques ont jusqu'alors été les plus étudiées et utilisées avec succès que ce soit en classification [Duda et al., 2012] ou plus largement en apprentissage automatique [Bishop, 2006]. Dans la suite, nous donnerons donc un bref aperçu des grands principes de ces deux classes qui, de plus, motiveront notre travail dans la suite de ce manuscrit. Pour les approches basées sur la théorie des croyances ou la théorie des ensembles flous et des possibilités, le lecteur intéressé pourra se référer à l'ouvrage [Bloch, 2003].

Méthodes par vote ou par moyennage

Les méthodes par vote, relatives principalement aux problèmes de classification, et les méthodes par moyennage, plutôt relatives aux problèmes de régression, forment une classe de méthodes simples et très largement étudiées. Toutes deux peuvent être *pondérées* de manière à inclure lors de l'étape de fusion une information sur la confiance portée en chacun des modèles fusionnés.

En régression, le principe général de la fusion par moyennage pondéré consiste à combiner les

prédictions issues des M modèles au moyen d'une simple somme pondérée telle que

$$\mathbf{y}(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m \mathbf{y}_m(\mathbf{x}). \tag{2.74}$$

Le terme α_m pondère la prédiction fournie par le modèle $\mathcal{M}_m$ et reflète, d'une manière générale, la confiance portée en ce modèle. Généralement, les poids α_m sont positifs ou nuls et doivent respecter la contrainte de normalisation suivante :

$$\sum_{m=1}^{M} \alpha_m = 1. \tag{2.75}$$

En classification, ce principe est connu sous le nom de *vote majoritaire*, éventuellement pondéré. En donnant le même poids à chacun des classifieurs, la décision finale sera simplement celle ayant été prise par le plus grand nombre de modèles. En ajoutant une pondération α_m pour chaque classifieur, on donne plus d'importance à la décision portée par les modèles $\mathcal{M}_m$ pour lesquels α_m est grand. La classe $\mathcal{C}_{j^*}$ finalement attribuée est telle que

$$j^* = \underset{j}{\operatorname{argmax}} \sum_{m=1}^{M} \alpha_m \, y_{m,j}. \tag{2.76}$$

Les applications du principe de moyennage ou de vote sont nombreuses. On citera, par exemple, les travaux de [Xu et al., 1992] pour la reconnaissance de caractères manuscrits et [Kuncheva et al., 2001]. C'est également ce principe qui est à l'origine de la technique de fusion *ROVER* (pour *Recognizer Output Voting Error Reduction*) dédié à la reconnaissance de la parole [Fiscus, 1997].

Plus récemment, ces règles de fusion ont également été exploitées dans d'autres systèmes de fusion où une attention particulière a été portée à la détermination des modèles à fusionner et/ou des poids de fusion α_m.

Bagging Le *bagging* [Breiman, 1996a], contraction de *bootstrap aggregating* en anglais, est un cas particulier de fusion par vote majoritaire où les modèles à fusionner sont obtenus en ne faisant varier que l'ensemble d'apprentissage, par *bootstrap* (voir 2.3.1). La prédiction finale peut être obtenue soit par moyennage des prédictions de chaque modèle avec $\alpha_m = 1/M$ si l'on a affaire à un problème de régression ou par vote majoritaire dans le cas d'un problème de classification. Les *forêts d'arbres décisionnels* [Breiman, 2001] sont l'exemple le plus connu de modèles construits à partir du principe de *bagging*. Nous illustrerons un peu plus ce principe dans la partie 2.3.3 pour la séparation de sources.

Boosting Contrairement au *bagging*, la méthode dite de *boosting* vise à combiner des modèles différents, appris sur les mêmes données [Schapire, 1990]. Cette méthode a été initialement proposée pour les problèmes de classification. L'idée principale réside dans la construction itérative du modèle fusionné. En effet, les classifieurs $\mathcal{M}_m$, souvent qualifiés de *faibles* car leurs performances doivent être seulement meilleures que le hasard, sont construits de façon séquentielle de sorte qu'à chaque itération, les exemples d'apprentissage mal classifiés par le classifieur faible construit à l'itération précédente auront davantage d'importance pour l'étape d'apprentissage du classifieur faible suivant. Le classifieur fusionné, qualifié lui de *fort*, combine les prédictions des classifieurs faibles ainsi construits selon la règle de vote majoritaire pondérée (2.76) où les poids α_m reflètent la qualité de classification de chaque classifieur.

L'algorithme *AdaBoost* [Freund et Schapire, 1997] est certainement l'algorithme de *boosting* le plus célèbre. À chaque itération, il convient dans un premier temps d'identifier le classifieur dont l'erreur de classification est minimale. Le poids α_m de ce classifieur et les poids des exemples d'apprentissage sont mis à jour en fonction de l'erreur de classification du modèle $\mathcal{M}_m$, de sorte que les exemples mal classifiés auront plus de poids à l'itération suivante. L'opération est répétée jusqu'à ce qu'aucun nouveau classifieur faible ne puisse diminuer l'erreur de classification.

Le principe de *boosting* a été également appliqué aux problèmes de régression [Duffy et Helmbold, 2002]. Aujourd'hui, le principe de *boosting* est encore employé. Par exemple, il est appliqué dans [Foucard et al., 2011] à la fusion de représentations temporelles multi-échelles pour la classification de musique ou dans [Cortes et al., 2014] pour la construction d'arbres décisionnels profonds.

Stacking L'approche par *stacking* (*empilement* en français), introduite sous le nom *stacked generalization* dans [Wolpert, 1992], n'est pas à proprement parler une méthode de fusion spécifique au même titre que le *boosting* ou le *bagging*. Elle représente plutôt une méthode générique visant à apprendre une fonction permettant de combiner les prédictions de M modèles préalablement sélectionnés et appris, sachant qu'aucun *a priori* n'est fait sur la nature des modèles envisagés ni sur les méthodes d'apprentissage employées.

Ainsi, la méthode peut être décrite en deux étapes distinctes et indépendantes. Dans un premier temps, chaque modèle $\mathcal{M}_m$ est appris sur les données d'apprentissage disponibles. Dans un deuxième temps, un modèle de combinaison, parfois appelé *meta-modèle* ou *meta-learner* en anglais, est appris sur ces mêmes données d'apprentissage. Contrairement aux modèles de base $\mathcal{M}_m$, ce dernier modèle dispose, en plus des données d'apprentissage, des prédictions $\mathbf{y}_m(\mathbf{x})$ données par chacun des M modèles à combiner.

Le principe de *stacking* a été employé avec succès pour de nombreux problèmes. La proposition initiale [Wolpert, 1992] est consacrée aux problèmes de classification. Les problèmes de régression sont eux étudiés dans [Breiman, 1996b].

Les approches par *stacking* ne font pas nécessairement partie de la catégorie des méthodes par vote ou moyennage. Toutefois, l'une des techniques les plus employées, et notamment introduite dans [Wolpert, 1992], consiste à simplement combiner les prédictions fournies par les M modèles au moyen d'une somme pondérée similaire à (2.74). Après avoir entraîné ces M modèles, il s'agit donc de déterminer les coefficients α_m permettant effectivement d'améliorer la prédiction fusionnée $\mathbf{y}(\mathbf{x})$. On notera, à titre d'exemple, que l'algorithme ayant remporté le *Prix Netflix* [1] exploitait ce principe de *stacking* [Koren].

Méthodes probabilistes et bayésiennes

Les méthodes de fusion probabilistes n'ont de différence avec les méthodes par vote ou moyennage présentées ci-dessus que la nécessité de définir les modèles dans un cadre probabiliste. Par exemple, un classifieur probabiliste $\mathcal{M}_m$ pourra attribuer à l'exemple $\mathbf{x}$ la classe $\mathcal{C}_{j^*}$ maximisant la probabilité *a posteriori* tel que

$$j^* = \underset{j}{\operatorname{argmax}}\ p(\mathcal{C}_j|\mathbf{x}, \mathcal{M}_m). \tag{2.77}$$

Dans ce cadre, il est alors possible d'exprimer les poids α_m de chacun des modèles sous forme de distribution de probabilité.

1. http://www.netflixprize.com/

Moyennage bayésien de modèles La façon la plus naturelle de formuler la fusion dans un cadre probabiliste consiste à employer des modèles $\mathcal{M}_m$ bayésiens. En effet, dans ce cas, il est possible de définir la probabilité *a posteriori* $p(\mathcal{M}_m|\mathbf{X})$ d'un modèle selon la règle de Bayes

$$p(\mathcal{M}_m|\mathbf{X}) = \frac{p(\mathbf{X}|\mathcal{M}_m)p(\mathcal{M}_m)}{\sum_{m'=1}^{M} p(\mathbf{X}|\mathcal{M}_{m'})p(\mathcal{M}_{m'})} \tag{2.78}$$

où $p(\mathcal{M}_m)$ représente la probabilité *a priori* du modèle $\mathcal{M}_m$ et $\mathbf{X}$ représente l'ensemble des données d'apprentissage.

Les coefficients α_m peuvent alors être identifiés à cette probabilité *a posteriori* $p(\mathcal{M}_m|\mathbf{X})$. Pour une application de classification, la prédiction pourra alors être formulée selon les probabilités $p(\mathcal{C}_j|\mathbf{x})$ moyennées sur les M modèles tels que

$$p(\mathcal{C}_j|\mathbf{x}) = \sum_{m=1}^{M} p(\mathcal{M}_m|\mathbf{X})\, p(\mathcal{C}_j|\mathbf{x}, \mathcal{M}_m). \tag{2.79}$$

De même, dans le cas de la régression, la prédiction $\mathbf{y}(\mathbf{x})$ pourra être moyennée selon

$$\mathbf{y}(\mathbf{x}) = \sum_{m=1}^{M} p(\mathcal{M}_m|\mathbf{X})\mathbf{y}_m(\mathbf{x}). \tag{2.80}$$

Ce principe bien connu porte le nom de *moyennage bayésien de modèles* [Hoeting et al., 1999]. Il a été tout aussi bien appliqué aux problèmes de classification qu'aux problèmes de régression [Raftery et al., 1997]. Nous aurons l'occasion de l'exploiter pour le moyennage de modèles NMF bayésiens dans le chapitre 5.

Critères d'information En pratique, le moyennage bayésien de modèles est difficile à mettre en œuvre car les probabilités *a posteriori* des modèles $p(\mathcal{M}_m|\mathbf{X})$ n'ont souvent pas de solution analytique. Dans ce cas, comme nous l'avons déjà évoqué dans la partie 2.3.1 relative au principe de sélection de modèles, il est possible d'employer des méthodes d'inférence approchées telles que les méthodes de Monte-Carlo par chaînes de Markov [Andrieu et al., 1999] ou l'inférence variationnelle [Bishop, 2006].

Une autre solution consiste à recourir à une approximation de la vraisemblance marginale des modèles $p(\mathbf{X}|\mathcal{M}_m)$ selon les principes introduits dans la partie 2.3.1 pour la sélection de modèles. En effet, il est proposé, notamment dans [Burnham et Anderson, 2004], d'exprimer les poids α_m en fonction des critères d'information type AIC (2.73) ou BIC (2.67). Ainsi, en notant Crit_m le critère d'information estimé pour le modèle $\mathcal{M}_m$, le poids de ce modèle peut s'écrire

$$\alpha_m = \frac{\mathrm{e}^{-\text{Crit}_m/2}}{\sum_{m'=1}^{M} \mathrm{e}^{-\text{Crit}_{m'}/2}}. \tag{2.81}$$

Combinaison bayésienne de modèles Considérer le moyennage bayésien de modèles comme une méthode de fusion de modèles a été discuté, parfois même critiqué de façon très virulente [Minka, 2002]. Il a été en effet reproché aux travaux [Domingos, 2000] de soutenir que le moyennage bayésien de modèles était bien moins performant que d'autres méthodes de fusion telles que le *bagging*. Sans alimenter à nouveau le débat, nous aimerions ici introduire une autre méthode de fusion bayésienne, nommée *combinaison bayésienne de modèles*, par opposition au moyennage bayésien de modèles.

Combinaison bayésienne et moyennage bayésien s'opposent sur l'hypothèse faite sur l'origine des données. Supposons ici que nous souhaitons modéliser de façon générative comment ont

été obtenues les données d'apprentissage $\mathbf{X}$. Pour ce faire, nous disposons d'un ensemble de M modèles $\{\mathcal{M}_1, ..., \mathcal{M}_m, ..., \mathcal{M}_M\}$. Dans le cas d'une fusion par moyennage bayésien, l'hypothèse génératrice sous-jacente suppose qu'un seul et unique modèle $\mathcal{M}_m$ est à l'origine des données $\mathbf{X}$. La probabilité *a posteriori* $p(\mathcal{M}_m|\mathbf{X})$ reflète la probabilité que le modèle $\mathcal{M}_m$ soit générateur des données $\mathbf{X}$ et le modèle $\mathcal{M}_{m^*}$ le plus probable est donc celui dont la probabilité *a posteriori* est maximale, tel que

$$\forall m \neq m^*, p(\mathcal{M}_{m^*}|\mathbf{X}) > p(\mathcal{M}_m|\mathbf{X}). \tag{2.82}$$

Si de nouvelles données d'apprentissage $\mathbf{X}'$ viennent compléter l'ensemble d'apprentissage initial, l'hypothèse selon laquelle le modèle $\mathcal{M}_{m^*}$ est le modèle générateur des données se trouve renforcée avec $p(\mathcal{M}_{m^*}|\mathbf{X}, \mathbf{X}') > p(\mathcal{M}_{m^*}|\mathbf{X})$. En résumé, les poids $\alpha_m = p(\mathcal{M}_m|\mathbf{X})$ reflètent l'impossibilité de distinguer quel modèle a généré les données, étant donné que les données disponibles sont limitées. Ils n'en restent pas moins une façon effective de moyenner les prédictions de plusieurs modèles selon l'équation (2.80).

À l'inverse, le principe de combinaison bayésienne suppose explicitement que les données d'apprentissage ont été générées par une combinaison linéaire des M modèles $\{\mathcal{M}_1, ..., \mathcal{M}_m, ..., \mathcal{M}_M\}$. Il ne s'agit donc plus d'estimer les probabilités des M modèles $\mathcal{M}_m$ indépendamment mais bien d'estimer la probabilité *a posteriori* de toutes les combinaisons linéaires possibles de ces mêmes M modèles. En notant $\mathcal{M}_{\boldsymbol{\alpha}}$ l'ensemble de ces modèles, la prédiction obtenue par moyennage s'écrit alors

$$\mathbf{y}(\mathbf{x}) = \sum_{\mathcal{M} \in \mathcal{M}_{\boldsymbol{\alpha}}} p(\mathcal{M}|\mathbf{X})\, \mathbf{y}_{\mathcal{M}}(\mathbf{x}) \tag{2.83}$$

où $\mathbf{y}_{\mathcal{M}}(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m \mathbf{y}_m(\mathbf{x})$ fait référence à la prédiction obtenue par le modèle combiné $\mathcal{M}$ et $p(\mathcal{M}|\mathbf{X})$ la probabilité *a posteriori* de ce même modèle combiné $\mathcal{M}$.

Formulé ainsi, la combinaison bayésienne de modèles revient à devoir estimer une infinité de probabilité *a posteriori* pour chacune des combinaisons linéaires $\mathcal{M} \in \mathcal{M}_{\boldsymbol{\alpha}}$, ce qui est évidemment illusoire en pratique. Toutefois, il a été montré dans [Monteith et al., 2011] qu'une simple discrétisation de l'ensemble des modèles $\mathcal{M}_{\boldsymbol{\alpha}}$ pour certaines valeurs de α_m pouvait suffire à améliorer les performances de classification comparativement au moyennage bayésien de modèles et aux approches par *boosting* et *bagging*. Malgré cela, le principe de combinaison bayésienne de modèles est encore peu exploité, au profit souvent du moyennage de modèles bayésien.

2.3.3 Fusion et sélection en séparation de sources sous-déterminée

Comme nous l'avons déjà évoqué, certaines méthodes de sélection et de fusion ont été développées pour des applications spécifiques, sans forcément faire appel aux grands principes que nous avons introduits dans les parties 2.3.1 et 2.3.2. Ainsi, nous proposons dans cette partie de passer en revue les applications relatives au problème de séparation de sources des méthodes de sélection et de fusion de modèles introduites plus tôt ainsi que d'autres propositions spécifiquement développées pour la séparation de sources.

Sélection de modèles

À notre connaissance, il n'existe pas encore de littérature générale portant sur la sélection de modèles dédiée aux problèmes de séparation de sources audio sous-déterminée. En pratique, lorsqu'un problème de séparation de sources se pose, le choix du modèle le plus adapté est donc souvent guidé par l'expérience ou par d'autres considérations pratiques (simplicité de la mise en œuvre par exemple). Il peut alors s'agir de choisir, entre autres, un type d'approche parmi celles

présentées dans la partie 2.1.2 par exemple, un type d'algorithme d'optimisation, les hyperparamètres d'un modèle (par exemple, l'ordre d'une NMF), *etc.* Ces choix sont loin d'être aisés et vont dépendre fortement du mélange $\mathbf{x}$ et des sources $\mathbf{s}_j(t)$ à séparer.

En fonction de la tâche considérée, il est tout de même possible de se référer à la littérature pour identifier les modèles les plus prometteurs. Par exemple, pour un problème de rehaussement de la parole, les campagnes d'évaluation sur le corpus *CHiME* [Vincent et al., 2013b] peuvent fournir des pistes intéressantes de choix. Plus généralement, les campagnes *SiSEC* (pour *Signal Separation Evaluation Campaing*) [Araki et al., 2012; Ono et al., 2013] définissent plusieurs tâches de séparation de sources et comparent différentes approches pour chacune.

Dans le cas où un ensemble de données représentatives de la tâche à réaliser peut être constitué, il est envisageable d'évaluer soi-même les performances moyennes de plusieurs modèles présélectionnés sur cet ensemble. Ces performances peuvent être par exemple évaluées à l'aide des métriques introduites dans la partie 2.1.5. Lorsque les modèles envisagés requièrent d'être appris sur un ensemble, il convient alors d'opérer la sélection par ré-échantillonage, comme évoqué dans la partie 2.3.1. Toutefois, dans le cas sous-déterminé, nous n'avons identifié aucune contribution majeure en ce sens.

De la même manière, les méthodes basés sur les critères d'information statistiques de type BIC ou AIC ne semblent pas avoir été largement exploitées. On pourra citer tout de même l'utilisation du critère BIC pour estimer le nombre de sources harmoniques dans une trame temporelle dans [Duan et al., 2008].

Finalement, nous avons identifié les contributions les plus marquantes pour le cas particulier des modèles NMF. En effet, comme nous l'avons déjà souligné dans la partie 2.1.3, le choix de l'ordre K d'une NMF, également nommé *nombre de composantes*, est déterminant pour la qualité de séparation [Bertin et al., 2007; Smaragdis, 2007]. Deux principales approches ont alors été proposées dans la littérature : une approche bayésienne exploitant les principes exposés dans la partie 2.3.1 et une approche basée sur le principe d'*élagage* (ou *pruning* en anglais).

Sélection bayésienne Comme cela a été évoqué dans [Tan et Févotte, 2013], les critères d'information type BIC ne peuvent être employés pour la sélection de l'ordre d'une NMF car ils supposent que le nombre d_m de paramètres à estimer ne dépend pas de la taille D des données. Or, pour la NMF, étant donnée la STFT d'un mélange de dimension $F \times N$, le nombre de paramètres de la NMF, $d_m = F \times K + K \times N$, se trouve être fonction de la taille des données représentée dans ce cas par le nombre de trames temporelles N.

Par conséquent, la sélection bayésienne ne peut être menée que par le recours à l'estimation de la probabilité *a posteriori* des modèles à comparer et donc l'estimation des *vraisemblances marginales* de chacun de ces modèles. Pour ce faire, comme nous l'avons déjà évoqué, il convient de recourir à des méthodes approchées. Les plus célèbres, à savoir les méthodes de Monte-Carlo par chaînes de Markov (MCMC) et les méthodes variationnelles bayésiennes (VB), ont toutes deux été exploitées dans cette optique. Quelle que soit la technique employée, la méthode proposée consiste à estimer la vraisemblance marginale de plusieurs NMFs d'ordres différents et de sélectionner la NMF dont la vraisemblance marginale est maximale.

Dans [Cemgil, 2009; Schmidt et al., 2009], les auteurs proposent de recourir à une estimation bayésienne par une méthode de MCMC. La vraisemblance marginale de chaque modèle est alors estimée grâce à la méthode dite de *Chib* [Chib, 1995]. Les méthodes VB ont également été développées pour la NMF dans [Cemgil, 2009; Winther et Petersen, 2007]. Nous reviendrons sur ces méthodes dans le chapitre 5 où nous exploiterons l'inférence VB pour notre cadre de fusion.

Sélection par élagage Les techniques d'élagage ne sont pas à proprement parler des techniques de sélection de modèle, contrairement aux méthodes générales introduites dans la partie

2.3.1, car elles ne permettent pas de comparer plusieurs modèles entre eux quels que soient ces modèles. Toutefois, leurs applications à la NMF en font un moyen effectif de sélection de l'ordre de la NMF. Le principe général des techniques d'élagage consiste à apprendre un modèle d'ordre élevé de façon itérative et à diminuer cet ordre au cours des itérations. Appliqué à la NMF, il consiste à faire évoluer l'ordre K d'une NMF d'une valeur élevée initiale à une valeur finale plus petite. Cette méthode a été introduite pour la NMF de diverses manières.

Dans [Tan et Févotte, 2009, 2013], l'élagage est implémenté au moyen de la technique dite de *détermination automatique de la pertinence* (*automatic relevance determination* en anglais, abrégé *ARD*) [MacKay, 1995] pour la factorisation de spectrogrammes de puissance $|\mathbf{X}|^{\cdot 2}$. Pour ce faire, le modèle introduit K paramètres λ_k nommés *poids de pertinence*, chaque poids étant relatif à l'une des K composantes du modèle de NMF. Les distributions *a priori* des colonnes $\mathbf{w}_k$ du dictionnaire et des lignes $\mathbf{h}_k$ de la matrice d'activation se trouvent alors conditionnées à ce paramètre λ_k dont la distribution *a priori* est de type Gamma (voir chapitre 5, équation (5.7)). Des règles de mises à jour multiplicatives des distributions des paramètres NMF $\mathbf{w}_k$ et $\mathbf{h}_k$ et des poids de pertinence λ_k sont obtenues par maximisation de leur probabilité *a posteriori* $p(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda} | |\mathbf{X}|^{\cdot 2})$. En pratique, les poids λ_k estimés vont former deux groupes selon leurs valeurs : un groupe de λ_k de valeurs faibles et un groupe de λ_k de valeurs plus élevées. Les composantes k dont les poids λ_k sont faibles sont alors qualifiées de *non pertinentes* et les normes des colonnes $\mathbf{w}_k$ et lignes $\mathbf{h}_k$ associées tendent toutes deux vers zéro. Par conséquent, ces composantes peuvent être retirées de la factorisation et l'ordre final de la NMF est alors donné par le nombre de composantes pertinentes dont les poids λ_k sont plus faibles. La publication [Tan et Févotte, 2013] étend la proposition originale [Tan et Févotte, 2009] aux β-divergences, alors que des travaux similaires ont été développés dans [Tipping, 2001].

Ce même principe a été également exploité dans un cadre bayésien par [Hoffman et al., 2010]. Cette fois, le poids λ_k pondère directement chaque composante de la NMF de sorte que le résultat de la NMF s'écrit

$$v_{fn} = \sum_{k=1}^{K} \lambda_k w_{fk} h_{kn}. \tag{2.84}$$

Contrairement à l'approche par ARD, un *a priori* de parcimonie est imposé sur la distribution des poids λ_k. De ce fait, les poids de certaines composantes vont tendre vers zéro et désactiver les colonnes $\mathbf{w}_k$ et lignes $\mathbf{h}_k$ correspondantes. Les paramètres NMF et les poids sont estimés par inférence VB.

Parmi les méthodes par élagage proposées, on notera que l'ARD a le mérite d'être moins coûteuse en temps de calcul, de par l'estimation des paramètres par maximum *a posteriori* plutôt que par inférence bayésienne complète.

Fusion de modèles

Si la littérature relative à la sélection de modèles n'est pas abondante, l'application du principe de fusion aux problèmes de séparation de sources est elle marginale. À notre connaissance, seules trois publications relativement récentes ont invoqué ce principe. Nous en donnons un rapide aperçu ci-après.

Fusion par *bagging* Dans [Chandna et Wang, 2014], il est proposé d'exploiter le principe de *bagging* introduit dans la partie 2.3.2 pour l'estimation de masques temps-fréquence d'une source de parole cible mélangée à deux autres sources de parole dans un mélange stéréo. La méthode proposée peut être décrite en trois temps. Dans un premier temps, le modèle spatial introduit dans [Alinaghi et al., 2011] permet d'estimer un masque temps-fréquence $\boldsymbol{\Xi} = \{\boldsymbol{\xi}_{fn}\}$ pour estimer la source de parole cible, selon le principe de masquage introduit dans la partie 2.1.4.

Cette première étape a surtout pour objectif d'estimer la distribution du mélange $\mathbf{x}(t)$ qui est supposé distribué selon une loi normale complexe

$$\mathbf{x}(t) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{2.85}$$

et dont les paramètres $\boldsymbol{\mu}$ et $\boldsymbol{\Sigma}$ sont estimés par maximum de vraisemblance au moyen d'un algorithme espérance-maximisation (EM). La deuxième étape consiste à générer B nouveaux mélanges $\{\mathbf{x}^{(b)}(t)\}_{b=1..B}$ de même taille que le mélange initial $\mathbf{x}(t)$. Ces mélanges sont obtenus par *bootstrap* (voir partie 2.3.1) selon la distribution (2.85) pour les paramètres $\tilde{\boldsymbol{\mu}}$ et $\tilde{\boldsymbol{\Sigma}}$. Enfin, dans un troisième temps, la méthode [Alinaghi et al., 2011] est réemployée sur chacun des mélanges de *bootstrap* $\mathbf{x}^{(b)}$ afin d'estimer B nouveaux masques $\boldsymbol{\Xi}^{(b)} = \{\boldsymbol{\xi}_{fn}^{(b)}\}$. Ces masques sont moyennés selon le principe de *bagging* afin de formuler le masque final

$$\forall f, n, \quad \boldsymbol{\xi}_{fn} = \frac{1}{B} \sum_{b=1}^{B} \boldsymbol{\xi}_{fn}^{(b)}. \tag{2.86}$$

La qualité de séparation est évaluée par SDR (voir partie 2.1.5) et l'auteur rapporte un gain moyen de 0.5 dB pour un total de 75 mélanges entre l'estimation initiale obtenue par maximum de vraisemblance et l'estimation obtenu par *bootstrap* avec $B = 500$. Le *bootstrap* appliqué à la séparation de sources permet donc d'améliorer significativement la qualité de séparation. Toutefois, la méthode proposée présente le défaut d'être très coûteuse en temps de calcul puisque le nombre de *boostraps* B employé est très grand et que chaque génération d'un mélange de *boostrap* emploie une méthode d'échantillonnage très complexe [Chandna et Walden, 2013].

Fusion séquentielle de modèles Dans [Arberet et al., 2012], il est proposé de fusionner plusieurs modèles de façon séquentielle. Dans le but d'alléger le temps d'estimation des modèles spectraux de sources tels que les modèles de mélanges gaussiens [Ozerov et al., 2007] ou la NMF, il est proposé de procéder en deux étapes. La première étape consiste en une *pré-estimation* de chacune des J sources au moyen d'un modèle quelconque, tel que le *modèle local gaussien* introduit dans [Vincent et al., 2009]. Une deuxième étape permet alors de *ré-estimer* chacune des sources indépendamment à l'aide d'un modèle spectral. L'erreur d'estimation à l'étape précédente est prise en compte grâce au calcul des moments d'ordres 2 et supérieurs de la source considérée. Il a été également proposé de répéter cette étape de *ré-estimation* une deuxième fois avec potentiellement un autre modèle spectral. Les expériences menées ont montré que la fusion séquentielle proposée permettait d'améliorer le SDR comparativement à d'autres méthodes type DUET [Jourjine et al., 2000] ou la NMF multicanale [Ozerov et Févotte, 2010].

Fusion de masques temps-fréquence Enfin, les principes de fusion par vote, par moyennage et par *stacking* ont été appliqués dans [Le Roux et al., 2013] à la fusion de masques temps-fréquence estimés à l'aide de différentes méthodes de rehaussement de la parole.

Les masques binaires et les masques doux ont tous deux été envisagés dans ce contexte. Pour le cas binaire, le masque résultant de la fusion des M masques binaires $\boldsymbol{\xi}_{m,fn}$ est obtenu par vote majoritaire non-pondéré et dans le cas de masques doux, le masque résultant est obtenu par moyennage non-pondéré, selon les principes évoqués dans la partie 2.3.2. Le cas de la médiane a été également étudié. Dans ce cas, le masque résultant s'écrit

$$\boldsymbol{\xi}_{fn} = \text{médiane}(\boldsymbol{\xi}_{1,fn}, ..., \boldsymbol{\xi}_{m,fn}, ..., \boldsymbol{\xi}_{M,fn}). \tag{2.87}$$

Pour l'approche par *stacking*, plusieurs classifieurs ont été étudiés, notamment les machines à vecteurs de support (SVM), les classifieurs bayésiens naïfs, les arbres de décision et les forêts

d'arbres décisionnels. Les classifieurs ont été appris de façon supervisée pour chaque bande de fréquence f. Plusieurs descripteurs ont été considérés en entrée, notamment afin de prendre en compte le contexte temporel et fréquentiel de chaque point temps-fréquence (f, n).

L'évaluation a été réalisée à partir du corpus *CHiME*, que nous introduirons dans la partie 3.3.2. Les séparateurs envisagés sont : les *séries de Taylor vectorielles* [Moreno et al., 1996], les *séries de Taylor vectorielles indirectes* [Le Roux et Hershey, 2012], le modèle *OMLSA-IMCRA* [Cohen, 2003] ainsi que les approches par minimum de l'erreur quadratique et log-minimum de l'erreur quadratique du système ROVER [Fiscus, 1997]. Les méthodes par vote et moyennage se sont révélées inefficaces, car elles n'améliorent pas les performances individuelles des séparateurs en terme de SDR moyen.

Pour l'approche par *stacking*, il a été montré que la prise en compte du contexte en entrée du SVM permettait d'améliorer significativement les performances, contrairement aux autres classifieurs dont les performances ne varient que très peu avec l'ajout de contexte. L'utilisation de masques doux en entrée a également permis d'améliorer les performances en terme de SDR mais en sortie, les masques binaires sont tout de même ceux qui permettent le gain le plus important par rapport au meilleur séparateur (gain de 2.36 dB de SDR).

Il est à noter que ces méthodes de fusion ont été publiées la même année que notre premier article lié aux travaux de cette thèse [Jaureguiberry et al., 2013]. Ces deux propositions concomitantes partagent quelques similarités, notamment car elles introduisent toutes deux la fusion de masques temps-fréquence. Toutefois, dans notre étude, la règle de fusion de masques proposée ne sert qu'à justifier l'introduction d'une autre règle de fusion exprimée dans le domaine temporel. Cette dernière formule l'estimée fusionnée $\mathbf{s}_j(t)$ d'une source comme la combinaison linéaire de M estimées $\mathbf{s}_{jm}(t)$ de cette même source, chacune étant pondérée par un coefficient de fusion α_m. Notre proposition ne requiert donc pas d'exprimer chaque estimée $\mathbf{s}_{jm}(t)$ comme le résultat d'une opération de masquage temps-fréquence, rendant ainsi notre règle de fusion plus générale. Nous verrons de plus que cette règle de fusion n'est qu'un cas particulier du cadre de fusion que nous introduirons dans le chapitre 3. Enfin, dans notre étude [Jaureguiberry et al., 2013] ainsi que dans ce manuscrit, nous nous intéressons aussi bien à la fusion de techniques différentes qu'à la fusion de techniques identiques dont seuls certains paramètres diffèrent, alors que les expériences dans [Le Roux et al., 2013] ne considèrent que le cas de la fusion de techniques différentes.

Chapitre 3

Cadre général pour la fusion en séparation de sources

Sommaire

Dans ce chapitre, nous introduisons le cadre général de fusion que nous avons développé et proposons une première analyse de son potentiel par l'étude de ses performances oracles sur les deux cas d'application que nous avons introduits dans la partie 2.2. Ce cadre général sera présenté dans la partie 3.1 et les cas particuliers que nous étudierons tout au long de ce document seront introduits dans la partie 3.2. Nous décrirons ensuite la mise en œuvre des modèles que nous avons introduits dans la partie 2.2 pour les deux cas d'application envisagés ainsi que les deux corpus sur lesquels nous évaluerons nos méthodes de fusion tout au long de ce document. Dans un premier temps, nous présenterons le corpus *CHiME* dédié au problème de rehaussement de parole dans la partie 3.3 puis nous introduirons dans la partie 3.4 le corpus *ccMixter* consacré à l'étude du problème d'extraction de voix chantée.

3.1 Cadre général

3.1.1 Formulation

Afin de pouvoir exploiter toute la diversité des méthodes de séparation que nous avons passées en revue dans la partie 2.1, nous avons voulu mettre au point un cadre de fusion général qui puisse être appliqué à n'importe laquelle de ces méthodes. Pour ce faire, nous avons adopté la formulation du problème de séparation introduite dans la partie 2.1. Ainsi, nous considérerons dans la suite que notre objectif consiste à estimer les J sources $\mathbf{s}_{j,fn}$ qui composent un mélange $\mathbf{x}$ défini par :

$$\forall f, n,\ \mathbf{x}_{fn} = \sum_{j=1}^{J} \mathbf{s}_{j,fn}. \tag{3.1}$$

Conformément aux principes présentés dans la partie 2.3, un cadre de fusion dédié à la séparation de sources doit nous permettre de formuler une nouvelle estimée $\tilde{\mathbf{s}}_{j,fn}$ de chacune des J sources à partir de M estimées $\tilde{\mathbf{s}}_{jm,fn}$ distinctes données par M méthodes de séparation différentes et indexées par m. Pour $m \in [1, M]$, $\tilde{\mathbf{s}}_{jm,fn}$ représente donc l'estimée de la source j par le séparateur m. Dans la suite, nous désignerons les méthodes envisagées pour la fusion par le terme de *séparateur*. Deux séparateurs pourront être fusionnés s'ils mettent en jeu des algorithmes d'estimation et des modèles différents (nous parlerons de *fusion hétérogène*) ou s'ils utilisent des algorithmes d'estimation et/ou modèles identiques mais dont seuls les hyperparamètres sont différents (nous parlerons alors de *fusion homogène*).

Pour opérer la fusion de M séparateurs, nous proposons donc simplement de formuler l'estimée fusionnée de la source j comme une somme pondérée des M estimées de cette même source selon :

$$\forall j, f, n, \quad \tilde{\mathbf{s}}_{j,fn} = \sum_{m=1}^{M} \alpha_{m,fn}\, \tilde{\mathbf{s}}_{jm,fn}. \tag{3.2}$$

Les coefficients $\alpha_{m,fn}$, ci-après dénommés *coefficients de fusion*, ont pour seules contraintes d'être positifs ou nuls et de sommer à 1, soit :

$$\forall f, n,\ \sum_{m=1}^{M} \alpha_{m,fn} = 1 \ \text{ et } \ \forall m,\ \alpha_{m,fn} \geq 0. \tag{3.3}$$

Bien que toutes les méthodes de séparation présentées dans la partie 2.1 n'opèrent pas dans le plan temps-fréquence, et bien qu'il soit possible d'envisager différentes résolutions temps-fréquence pour chacune des méthodes à fusionner, il sera toujours possible, afin d'opérer la fusion, d'exprimer les M estimées d'une même source à l'aide d'une représentation identique. Cette règle de fusion reste donc très générale.

3.1.2 Justification des contraintes

Nous avons défini en (3.3) deux contraintes sur la valeur des coefficients de fusion. Ces deux contraintes peuvent être justifiées par des considérations pratiques.

En premier lieu, la contrainte de sommation à 1 nous permet de respecter le modèle de mélange (3.1) adopté. En effet, si nous ne considérons que des séparateurs respectant ce modèle, alors l'équation de mélange est aussi vérifiée pour les sources estimées $\tilde{\mathbf{s}}_{jm,fn}$ de sorte que :

$$\forall m, f, n, \quad \sum_{j=1}^{J} \tilde{\mathbf{s}}_{jm,fn} = \mathbf{x}_{fn}. \tag{3.4}$$

En injectant cette expression dans notre règle de fusion (3.2), nous pouvons vérifier que les sources fusionnées $\tilde{\mathbf{s}}_{j,fn}$ respectent aussi l'équation de mélange (3.1) grâce à cette contrainte de sommation à 1 des coefficients de fusion :

$$\begin{aligned}
\forall f, n, \quad \sum_{j=1}^{J} \tilde{\mathbf{s}}_{j,fn} &= \sum_{j=1}^{J} \left(\sum_{m=1}^{M} \alpha_{m,fn} \tilde{\mathbf{s}}_{jm,fn} \right) && (3.5)\\
&= \sum_{m=1}^{M} \alpha_{m,fn} \left(\sum_{j=1}^{J} \tilde{\mathbf{s}}_{jm,fn} \right) && (3.6)\\
&= \left(\sum_{m=1}^{M} \alpha_{m,fn} \right) \mathbf{x}_{fn} && (3.7)\\
&= \mathbf{x}_{fn}. && (3.8)
\end{aligned}$$

D'autre part, la contrainte de positivité des coefficients de fusion nous permet de conserver une interprétation intuitive de la règle de fusion (3.2). En effet, assigner un coefficient de fusion négatif à l'un ou l'autre des M séparateurs reviendrait à inverser la phase des sources estimées par cette méthode, ce qui n'a pas de sens physique.

3.1.3 Parallèle avec la fusion en classification

Les deux contraintes introduites nous permettent aussi de donner un autre éclairage sur la règle de fusion (3.2) en la comparant aux approches classiques de fusion en classification. Dans nos travaux préliminaires [Jaureguiberry et al., 2013], nous avions introduit un cas particulier de la règle de fusion (3.2). Nous proposions de restreindre la fusion à des séparateurs basés sur l'estimation de masques temps-fréquence, conformément aux principes introduits dans la partie 2.1.4. La règle de fusion se limitait donc à estimer un nouveau masque $\tilde{\xi}_{j,fn}$ à partir d'une combinaison linéaire de M masques $\tilde{\xi}_{jm,fn}$ distincts pondérés par des coefficients de fusion selon :

$$\tilde{\xi}_{j,fn} = \sum_{m=1}^{M} \alpha_{m,fn} \tilde{\xi}_{jm,fn}. \tag{3.9}$$

Lorsque $\forall j, m, f, n, \tilde{\xi}_{jm,fn} \in \{0, 1\}$, on parle de masquage *binaire*. Dans ce cas, le problème de séparation de sources peut être compris comme un problème de classification. En effet, si un classifieur a pour objectif d'assigner un objet à une classe, un séparateur peut être vu comme un algorithme destiné à assigner chaque point temps-fréquence (f, n) du mélange $\mathbf{x}_{fn}$ à l'une des J sources qui le composent. Cette comparaison entre classification et séparation de sources a d'ailleurs été utilisée dans [Raphael, 2008] afin de mettre au point une méthode de séparation de signaux musicaux originale guidée par la partition de l'œuvre. Dans le cas binaire, on notera

que la règle de fusion dite *par vote majoritaire* (voir partie 2.3.2) est particulièrement populaire [Kittler et al., 1998].

Toutefois, les méthodes de séparation basées sur du masquage qualifié de *doux* sont connues pour donner des résultats de meilleure qualité par rapport aux méthodes basées sur du masquage binaire [Li et Wang, 2009; Reddy et Raj, 2007]. Pour rappel, on parle de masquage doux lorsque $\forall j, m, f, n, \tilde{\xi}_{jm,fn} \in [0, 1]$ tel que $\forall m, f, n, \sum_j \tilde{\xi}_{jm,fn} = 1$. Dans ce cas, un élément $\tilde{\xi}_{jm,fn}$ du masque peut être interprété comme étant la probabilité *a posteriori* estimée par le séparateur m que le point temps-fréquence (f, n) appartienne à la source j, de façon similaire aux problèmes de classification multiclasse [Tax et Duin, 2002]. Le coefficient de fusion $\alpha_{m,fn}$ peut alors être compris comme reflétant le degré de confiance porté au séparateur m pour le point temps-fréquence (f, n) [Xu et al., 1992]. On comprend alors mieux qu'il sera difficile d'obtenir une telle information pour tous les points temps-fréquence d'un mélange quelconque à séparer. Dans la suite, nous étudierons donc des cas particuliers de la règle de fusion générale (3.2).

Enfin, notons que les contraintes sur les coefficients de fusion nous permettent de garder la même interprétation probabiliste pour le masque fusionné $\tilde{\xi}_{j,fn}$, à savoir qu'il exprime la probabilité *a posteriori* que le point temps-fréquence (f, n) appartienne à la source j.

3.2 Cas particuliers

Afin d'appliquer la règle de fusion (3.2) en pratique, nous nous intéresserons par la suite aux moyens de déterminer les coefficients de fusion. Avant d'aborder ces aspects, nous pouvons d'ores et déjà déterminer plusieurs cas particuliers de la règle de fusion (3.2).

3.2.1 Fusion invariante, fusion variant en temps et fusion variant en fréquence

Les cas de fusion peuvent être distingués selon que les coefficients de fusion $\alpha_{m,fn}$ dépendent ou non de la bande de fréquence f et/ou de la trame temporelle n.

Fusion invariante

Le premier cas particulier consiste à considérer que les coefficients de fusion $\alpha_{m,fn}$ sont indépendants du point temps-fréquence. Dans cecas, la règle de fusion (3.2) proposée peut être plus simplement exprimée dans le domaine temporel :

$$\forall j, t,\ \tilde{\mathbf{s}}_j(t) = \sum_{m=1}^{M} \alpha_m \tilde{\mathbf{s}}_{jm}(t). \tag{3.10}$$

Pour le cas invariant, cette formulation est en effe parfaitement équivalente à notre règle générale (3.2) et est obtenue par transformée inverse des signaux $\tilde{\mathbf{s}}_{j,fn}$ et $\tilde{\mathbf{s}}_{jm,fn}$.

Fusion variant en temps ou en fréquence

La règle de fusion générale peut être également simplifiée en ne choisissant de conserver la dépendance des coefficients de fusion qu'en temps ou qu'en fréquence. De même que pour la fusion invariante, nous pouvons alors réexprimer cette règle dans le domaine temporel.

Nous parlerons de *fusion variant en temps* lorsque les coefficients de fusion seront indépendants de la fréquence. La règle de fusion correspondante s'exprime alors comme :

$$\forall j, n, t,\ \tilde{\mathbf{s}}_j^n(t) = \sum_{m=1}^{M} \alpha_{m,n} \tilde{\mathbf{s}}_{jm}^n(t), \tag{3.11}$$

où $\tilde{\mathbf{s}}_{jm}^{n}(t)$ et $\tilde{\mathbf{s}}_{j}^{n}(t)$ font respectivement référence aux sources estimées par les M séparateurs et à la source fusionnée pour la trame n. Ces signaux peuvent être simplement obtenus par transformée inverse de la trame n des signaux correspondants exprimés dans le plan temps-fréquence, à savoir $\{\tilde{\mathbf{s}}_{jm,fn}\}_{f=1..F}$ et $\{\tilde{\mathbf{s}}_{j,fn}\}_{f=1..F}$. La source fusionnée complète $\tilde{\mathbf{s}}_j(t)$ est alors obtenue par une opération classique d'addition-recouvrement.

De la même manière, la *fusion variant en fréquence* peut s'exprimer à l'aide des estimées temporelles $\tilde{\mathbf{s}}_{jm}^{f}(t)$ et $\tilde{\mathbf{s}}_{j}^{f}(t)$ représentant les signaux, estimés et fusionnés, filtrés par le filtre correspondant à la bande de fréquence f de la transformée temps-fréquence utilisée. Ces signaux filtrés sont obtenus par transformée inverse des signaux correspondants exprimés dans le plan temps-fréquence, à savoir $\{\tilde{\mathbf{s}}_{jm,fn}\}_{n=1..N}$ et $\{\tilde{\mathbf{s}}_{j,fn}\}_{n=1..N}$. La règle de fusion variant en fréquence s'exprime alors dans le domaine temporel comme :

$$\forall j, f, t,\ \tilde{\mathbf{s}}_{j}^{f}(t) = \sum_{m=1}^{M} \alpha_{m,f} \tilde{\mathbf{s}}_{jm}^{f}(t). \tag{3.12}$$

La source fusionnée complète $\tilde{\mathbf{s}}_j(t)$ est alors reconstruite en sommant dans le domaine temporel les contributions de toutes les bandes de fréquence f :

$$\forall j, t,\ \tilde{\mathbf{s}}_{j}(t) = \sum_{f=1}^{F} \tilde{\mathbf{s}}_{j}^{f}(t). \tag{3.13}$$

3.2.2 Fusion statique et fusion adaptative

Deux autres cas particuliers, complémentaires des précédents, peuvent être distingués selon que les coefficients de fusion dépendent ou non du mélange $\mathbf{x}(t)$ à séparer. Ainsi, dans la suite, nous nous intéresserons à des moyens de déterminer les coefficients de fusion indépendamment du mélange à traiter. Nous parlerons alors de *fusion statique*. Par opposition, nous parlerons de *fusion adaptative* lorsque les coefficients de fusion seront déterminés en fonction du mélange à traiter. Notons par avance que dans le cas de la fusion statique, les règles de fusion invariante (3.10) et variant en temps (3.11) sont strictement identiques.

3.2.3 Performance oracle de fusion

Nous avons évoqué dans la partie 2.1.5 différentes métriques permettant d'évaluer la qualité de séparation de sources. Dans nos expériences, nous donnerons plusieurs de ces métriques afin d'évaluer nos méthodes de fusion. Parmi elles, nous porterons une attention particulière au rapport signal à distortion (SDR) [Vincent et al., 2006] qui est sans nul doute la mesure la plus utilisée dans les publications de séparation de sources audio car elle donne une mesure globale de la qualité de séparation. En particulier, cette métrique va nous permettre d'évaluer le potentiel de nos méthodes de fusion.

Pour cela, nous proposons de déterminer les coefficients de fusion $\tilde{\alpha}_{m,fn}$ qui permettent, pour chacun des types de fusion définis dans la partie 3.2.1, de maximiser le SDR de la source fusionnée $\tilde{\mathbf{s}}_j$:

$$\mathrm{SDR}\left[\tilde{\mathbf{s}}_j\right] = 10\log_{10}\frac{\sum_t \|\mathbf{s}_j(t)\|^2}{\sum_t \|\mathbf{s}_j(t) - \tilde{\mathbf{s}}_j(t)\|^2}. \tag{3.14}$$

Rappelant que $\mathbf{s}_j(t)$ est la $j^{\text{ième}}$ source vraie composant le mélange $\mathbf{x}(t)$, il est évident que la maximisation de (3.14) ne peut être menée que dans un cadre expérimental où cette source vraie est connue. C'est pourquoi nous qualifierons d'*oracles* les coefficients ainsi déterminés, et leurs résultats associés, afin de souligner qu'il ne s'agit pas de résultats qui peuvent être obtenus en

pratique. Toutefois, les résultats oracles permettront d'évaluer les performances de nos méthodes de fusion puisqu'ils constituent une borne supérieure des performances que l'on peut attendre d'une règle de fusion dans un cas pratique.

Oracle de fusion invariante

Les coefficients de fusion invariante oracle $\tilde{\alpha}_m$ sont obtenus en maximisant le SDR de la source fusionnée $\tilde{\mathbf{s}}_j$ définie à l'équation (3.10). En injectant cette expression dans la définition du SDR (3.14), le problème d'optimisation est formulé comme un problème de maximisation sous contraintes d'égalité et d'inégalité :

$$\begin{array}{ll} \underset{\{\alpha_m\}_{m=1..M}}{\operatorname{argmax}} & 10\log_{10}\dfrac{\sum_t \|\mathbf{s}_j(t)\|^2}{\sum_t \|\mathbf{s}_j(t)-\tilde{\mathbf{s}}_j(t)\|^2} \\ \text{avec} & \left\{\begin{array}{l} \forall m,\ \alpha_m \geq 0 \\ \sum_{m=1}^{M} \alpha_m = 1 \end{array}\right. . \end{array} \tag{3.15}$$

En omettant le numérateur $\sum_t \|\mathbf{s}_j(t)\|^2$ qui est constant, ce problème revient à minimiser l'erreur quadratique moyenne (EQM) entre la source vraie $\mathbf{s}_j(t)$ et son estimée fusionnée $\sum_{m=1}^{M} \alpha_m \tilde{\mathbf{s}}_{jm}(t)$. Il peut donc être formulé comme un programme quadratique (PQ) standard [Bertsekas, 1999] sous forme matricielle :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} & c + \boldsymbol{\alpha}^\intercal \widetilde{\mathbf{G}}\, \boldsymbol{\alpha} - 2\,\tilde{\mathbf{d}}^\intercal \boldsymbol{\alpha} \\ \text{avec} & \left\{\begin{array}{l} \forall m,\ \alpha_m \geq 0 \\ \sum_{m=1}^{M} \alpha_m = 1 \end{array}\right. \end{array} \tag{3.16}$$

où $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_m, ..., \alpha_M]^\intercal$ représente le vecteur de fusion oracle et $\boldsymbol{\alpha}^\intercal$ sa transposée. La matrice $\widetilde{\mathbf{G}} = \{\tilde{g}_{m_1 m_2}\}$ de taille $M \times M$ est communément appelée *matrice de Gram* car elle est composée des produits scalaires entre signaux estimés, de sorte que :

$$\forall m_1, m_2,\ \tilde{g}_{m_1 m_2} = \sum_t \langle \tilde{\mathbf{s}}_{jm_1}(t), \tilde{\mathbf{s}}_{jm_2}(t) \rangle = \sum_t \tilde{\mathbf{s}}_{jm_1}^\intercal(t)\, \tilde{\mathbf{s}}_{jm_2}(t). \tag{3.17}$$

De façon similaire, le vecteur $\tilde{\mathbf{d}} = \left\{\tilde{d}_m\right\}$ de longueur M est composé des produits scalaires entre les signaux estimés et la source vraie et le scalaire c est le carré de la norme euclidienne de la source vraie :

$$\begin{array}{ll} \forall m, & \tilde{d}_m = \sum_t \langle \mathbf{s}_j(t), \tilde{\mathbf{s}}_{jm}(t) \rangle = \sum_t \mathbf{s}_j^\intercal(t)\, \tilde{\mathbf{s}}_{jm}(t) \\ \text{et} & c = \sum_t \|\mathbf{s}_j(t)\|^2 = \sum_t \mathbf{s}_j^\intercal(t)\, \mathbf{s}_j(t). \end{array} \tag{3.18}$$

Oracle de fusion variant en temps ou en fréquence

Les coefficients de fusion oracle variant en temps et variant en fréquence peuvent être obtenus de manière similaire au cas de fusion invariante.

Ainsi, les coefficients de fusion variant en temps $\boldsymbol{\alpha}_n = \{\alpha_{m,n}\}_{m=1..M}$ sont obtenus en remplaçant dans le problème PQ (3.16) la source vraie $\mathbf{s}_j(t)$ par la $n^{\text{ième}}$ trame correspondante $\mathbf{s}_j^n(t)$ et les sources estimées $\tilde{\mathbf{s}}_{jm}(t)$ par les trames correspondantes $\tilde{\mathbf{s}}_{jm}^n(t)$:

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}_n}{\operatorname{argmin}} & c_n + \boldsymbol{\alpha}_n{}^\intercal \widetilde{\mathbf{G}}_n\, \boldsymbol{\alpha}_n - 2\,\tilde{\mathbf{d}}_n^\intercal \boldsymbol{\alpha}_n \\ \text{avec} & \left\{\begin{array}{l} \forall m,\ \alpha_{m,n} \geq 0 \\ \sum_{m=1}^{M} \alpha_{m,n} = 1 \end{array}\right. \end{array} \tag{3.19}$$

avec

$$\forall m_1, m_2, \quad \tilde{g}_{n,m_1m_2} = \sum_t \left\langle \tilde{\mathbf{s}}^n_{jm_1}(t), \tilde{\mathbf{s}}^n_{jm_2}(t) \right\rangle, \tag{3.20}$$

$$\forall m, \quad \tilde{d}_{n,m} = \sum_t \left\langle \mathbf{s}^n_j(t), \tilde{\mathbf{s}}^n_{jm}(t) \right\rangle \tag{3.21}$$

$$\text{et} \quad c_n = \sum_t \|\mathbf{s}^n_j(t)\|^2. \tag{3.22}$$

Il s'agit donc de résoudre N problèmes PQ simultanés.

De la même façon, les coefficients de fusion variant en fréquence $\boldsymbol{\alpha}_f = \{\alpha_{m,f}\}_{f=1..F}$ sont obtenus en remplaçant dans le problème PQ (3.16) la source vraie $\mathbf{s}_j(t)$ par sa version filtrée $\mathbf{s}^f_j(t)$ sur la bande de fréquence f correspondante et les sources estimées $\tilde{\mathbf{s}}^f_{jm}(t)$ par leurs versions filtrées $\tilde{\mathbf{s}}^f_{jm}(t)$ sur cette même bande de fréquence f. Ainsi le problème est formulé comme :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}_f}{\operatorname{argmin}} & c_f + \boldsymbol{\alpha}_f{}^\intercal \, \widetilde{\mathbf{G}}_f \, \boldsymbol{\alpha}_f - 2\, \tilde{\mathbf{d}}_f^\intercal \, \boldsymbol{\alpha}_f \\ \text{avec} & \left\{ \begin{array}{l} \forall m,\, \alpha_{m,f} \geq 0 \\ \sum_{m=1}^M \alpha_{m,f} = 1 \end{array} \right. \end{array} \tag{3.23}$$

avec

$$\forall m_1, m_2, \quad \tilde{g}_{f,m_1m_2} = \sum_t \left\langle \tilde{\mathbf{s}}^f_{jm_1}(t), \tilde{\mathbf{s}}^f_{jm_2}(t) \right\rangle, \tag{3.24}$$

$$\forall m, \quad \tilde{d}_{f,m} = \sum_t \left\langle \mathbf{s}^f_j(t), \tilde{\mathbf{s}}^f_{jm}(t) \right\rangle \tag{3.25}$$

$$\text{et} \quad c_f = \sum_t \|\mathbf{s}^f_j(t)\|^2. \tag{3.26}$$

Il s'agit donc de résoudre F problèmes PQ simultanés.

Fusion ou sélection

Comme nous l'avons introduit dans la partie 2.3, la fusion est une alternative à la simple sélection. Là où la fusion cherche à combiner plusieurs séparateurs, la sélection n'en retient qu'un seul. Ainsi, dans nos expériences, nous proposerons de comparer les résultats de fusion oracle aux résultats de sélection oracle correspondants.

Par exemple, dans le cas invariant, les résultats de sélection oracle seront obtenus en résolvant le même problème PQ que pour la fusion (3.16) mais avec la contrainte supplémentaire qu'un seul des coefficients α_m peut avoir une valeur non-nulle, soit :

$$\exists!\, m' \in [1, M] \text{ tel que } \left\{ \begin{array}{l} \alpha_{m'} = 1 \\ \forall m \neq m', \alpha_m = 0. \end{array} \right. \tag{3.27}$$

Il en sera de même pour les cas variant en temps et variant en fréquence où cette contrainte sera exprimée respectivement sur les coefficients $\alpha_{m,n}$ pour le problème (3.19) et sur les coefficients $\alpha_{m,f}$ pour le problème (3.23).

3.3 Fusion homogène : application au rehaussement de la parole monocanal

Nous introduisons dans cette partie le premier cas d'application qui sera étudié tout au long de ce document. Il concerne le problème de rehaussement de la parole dans un mélange monocanal

bruité et sera traité à l'aide de séparateurs qualifiés d'*homogènes* en ce qu'ils utiliseront tous un même algorithme d'estimation (le cadre général de factorisation en matrices non-négatives présenté dans la partie 2.1) mais des modèles de structure et de nombre de paramètres différents. Dans la partie 3.3.1, nous rappellerons les principes du problème de rehaussement de la parole déjà introduit dans la partie 2.2.1 puis nous présenterons dans la partie 3.3.2 le corpus choisi pour tester nos méthodes de fusion. Les parties 3.3.3 et 3.3.4 seront elles consacrées à la description des méthodes d'apprentissage des modèles utilisés pour la séparation. Ces modèles seront ensuite évalués sur notre corpus dans la partie 3.3.5. Enfin, nous évaluerons dans la partie 3.3.6 le potentiel de nos méthodes de fusion sur ce corpus en discutant les résultats oracles obtenus.

3.3.1 Réhaussement de la parole monocanal

Nous nous intéressons donc ici au problème du rehaussement de la parole monocanal, qui consiste, comme nous l'avons déjà présenté dans la partie 2.2.1, à nettoyer un signal de parole que nous noterons $s_1(t)$ d'un signal de bruit $s_2(t)$, à partir de l'observation de leur mélange $x(t)$ défini dans le plan temps-fréquence comme :

$$x_{fn} = s_{1,fn} + s_{2,fn}. \tag{3.28}$$

Pour ce faire, nous proposons d'utiliser l'un des modèles de source les plus populaires [Geiger et al., 2013; Mohammadiha et al., 2012; Moritz et al., 2013; Raj et al., 2011] : la NMF, que nous avons déjà introduite dans la partie 2.1.3 et qui, de plus, a atteint de très bonnes performances sur le corpus CHiME que nous allons utiliser [Geiger et al., 2013; Vincent et al., 2013a].

3.3.2 Corpus CHiME

Le corpus *CHiME* (*Computational Hearing in Multisource Environments*) a été initialement proposé dans [Christensen et al., 2010] afin de fournir des données pour la recherche en traitement de la parole en environnement très bruité. En particulier, l'objectif annoncé des auteurs était d'améliorer les technologies de reconnaissance automatique de la parole pour les rendre robustes aux nombreuses sources de bruit pouvant interférer dans un environnement domestique quotidien souvent réverbérant.

Deux défis alliant à la fois séparation et reconnaissance de parole ont alors été proposés sur la base de ce corpus [Barker et al., 2013; Vincent et al., 2013b]. Le plus récent de ces deux défis [Vincent et al., 2013b] proposait deux tâches de reconnaissance, l'une sur un vocabulaire de petite taille issu du corpus de parole *Grid* [Cooke et al., 2006], et l'autre sur un vocabulaire de taille moyenne issu du corpus *Wall Street Journal (WSJ0)* [Garofalo et al., 2007]. Pour nos expériences, nous utiliserons les données fournies pour la première tâche dont le contenu est décrit ci-après.

Description

Les signaux de parole sont tirés du corpus *Grid* [Cooke et al., 2006]. Chaque phrase est une commande composée de 6 mots et prononcée par un locuteur parmi 34 locuteurs différents. Ces signaux de voix ont dans un premier temps été convolués aux réponses impulsionnelles binaurales de la pièce choisie pour l'enregistrement afin d'ajouter un effet de réverbération et de simuler de petits mouvements de tête du locuteur.

Par la suite, ces signaux de parole réverbérés ont été mélangés à des enregistrements de bruits réalisés dans cette même pièce à l'aide d'une tête artificielle. Deux microphones placés dans les oreilles permettent de simuler les signaux acoustiques que percevrait un adulte moyen. 14 heures d'enregistrement ont été réalisées avec ce dispositif au sein de la pièce principale d'une maison

habitée par deux adultes et deux enfants. Les bruits ainsi enregistrés sont donc typiques d'une maison familiale et comprennent entre autres : des bruits de pas, le son de la télévision, des conversations, des bruits d'appareils électroniques, *etc*.

La position temporelle des séquences de mots dans le bruit a été choisie afin de produire des mélanges à six rapports signal-à-bruit (RSBs) différents exprimés en décibels (dB) parmi $\{-6, -3, 0, 3, 6, 9\}$. Les mélanges stéréos obtenus ont été échantillonnés à 16 kHz. Enfin, ces données ont été réparties en trois ensembles de données : un ensemble d'apprentissage, un ensemble de développement et un ensemble de test.

Organisation

Pour nos expériences, les ensembles sus-cités ont été scindés en quatre ensembles distincts : un ensemble d'apprentissage, un ensemble de développement, un ensemble de validation et un ensemble de test.

L'*ensemble d'apprentissage* est composé de 500 séquences de mots pour chacun des 34 locuteurs. Chacune de ces séquences est disponible avec et sans réverbération. Cet ensemble correspond à l'ensemble d'apprentissage tel que défini dans le corpus original et sera ici utilisé afin d'apprendre des modèles spectraux de parole pour chacun des locuteurs.

L'*ensemble de développement* est composé de 600 mélanges bruités pour chacun des 6 RSBs définis plus tôt, soit un total de 3600 mélanges bruités. Nous disposons en plus pour chacun de ces mélanges de 10 secondes de bruit seul récupérées avant et après les signaux de parole, et destinés à l'apprentissage des modèles de bruit. Les signaux de parole sont également disponibles sans bruit. Ils permettront d'évaluer la qualité de séparation. Cet ensemble correspond à l'ensemble de développement du corpus original et sera dédié ici à l'apprentissage des coefficients de fusion.

Les *ensembles de validation et de test* ont été constitués à partir de l'ensemble de test du corpus original. Ils sont tous deux composés d'un total de 204 mélanges bruités (un par locuteur et par RSB) choisis au hasard. De même que pour l'ensemble de développement, 10 secondes de bruit seul sélectionnées avant et après les séquences de mots sont disponibles afin de modéliser le bruit pour chacun de ces mélanges. Les signaux de parole sont également disponibles sans bruit. Ils serviront à évaluer les performances de séparation et de fusion pour chaque exemple de ces ensembles. Ces deux ensembles seront exclusivement dédiés à l'évaluation de nos méthodes et ne serviront à aucun moment à un quelconque apprentissage.

Conversion stéréo vers mono

Originellement, le corpus CHiME est distribué en stéréo. En revanche, nous avons mené nos expériences en configuration monocanale. Pour convertir le corpus de stéréo à mono, deux solutions peuvent être envisagées. La première consiste simplement à sommer les signaux temporels relatifs aux canaux droit et gauche et à travailler sur le signal résultant. Cette méthode bien qu'immédiate risque d'endommager le signal de parole par effet de filtrage en peigne lorsque ce dernier n'est pas centré. Étant donné que les signaux de parole ont été modifiés afin de simuler de petits mouvements de tête du locuteur, ce phénomène pourrait se produire même si nous n'en avons pas vérifié l'existence sur ce corpus.

Par précaution, nous avons donc choisi une deuxième méthode de conversion stéréo vers mono. Cette méthode consiste simplement à sommer les spectrogrammes de puissance des canaux gauche et droit, plutôt que les signaux temporels. En effet, les spectrogrammes de puissance étant découplés de l'information de phase, le phénomène de filtrage en peigne ne peut plus avoir lieu. Des méthodes de séparation monocanale peuvent alors être appliquées au spectrogramme résultant. Cette deuxième méthode requiert toutefois de reconstruire les signaux estimés en stéréo, afin de pouvoir notamment procéder à l'évaluation de la qualité de séparation. Pour cela, nous proposerons

d'appliquer le même masque temps-fréquence mono estimé au canal droit et au canal gauche du mélange.

Séparateurs envisagés

Afin d'évaluer nos méthodes de fusion, nous devons définir un ensemble de M séparateurs à étudier. Nous avons décidé de dédier le corpus CHiME à l'étude de la fusion de séparateurs dits *homogènes*. Plus précisément, nous proposons ici de modéliser la source de parole et la source de bruit par des modèles NMF comme présenté à la partie 2.2.1. Le modèle de bruit sera appris pour chaque mélange et la méthode employée sera décrite dans la partie 3.3.3. Nous proposerons de fusionner les résultats obtenus pour différents modèles de parole dont l'apprentissage sera décrit dans la partie 3.3.4.

L'implémentation de nos modèles de NMF a été réalisée grâce à la boîte à outil *MATLAB* nommée *FASST*[1] (pour *Flexible Audio Source Separation Toolbox*), basée sur le cadre général de factorisation en matrices non-négatives [Ozerov et al., 2012], dont le moteur a également été codé en *C++* [Salaün et al., 2014].

3.3.3 Apprentissage du modèle de bruit

Comme présenté dans la partie 2.2.1, nous proposons de modéliser la source de bruit s_2 à l'aide d'un modèle de NMF simple dont les caractéristiques spectrales seront apprises au préalable sur les segments de bruit seul disponibles avant et après la séquence de mots. Ce signal de bruit seul, noté $x_b(t)$ et dont le spectrogramme de log-puissance QERB (pour *Quadratic Equivalent Rectangular Bandwidth*, voir annexe A) est illustré à la figure 3.1a, est modélisé par NMF. La seule source qui le compose est donc distribuée selon une loi normale univariée dont la variance est le résultat d'une NMF telle que :

$$v_{2,fn} = \sum_{k=1}^{K_2} w_{2,fk} h_{2,kn}. \tag{3.29}$$

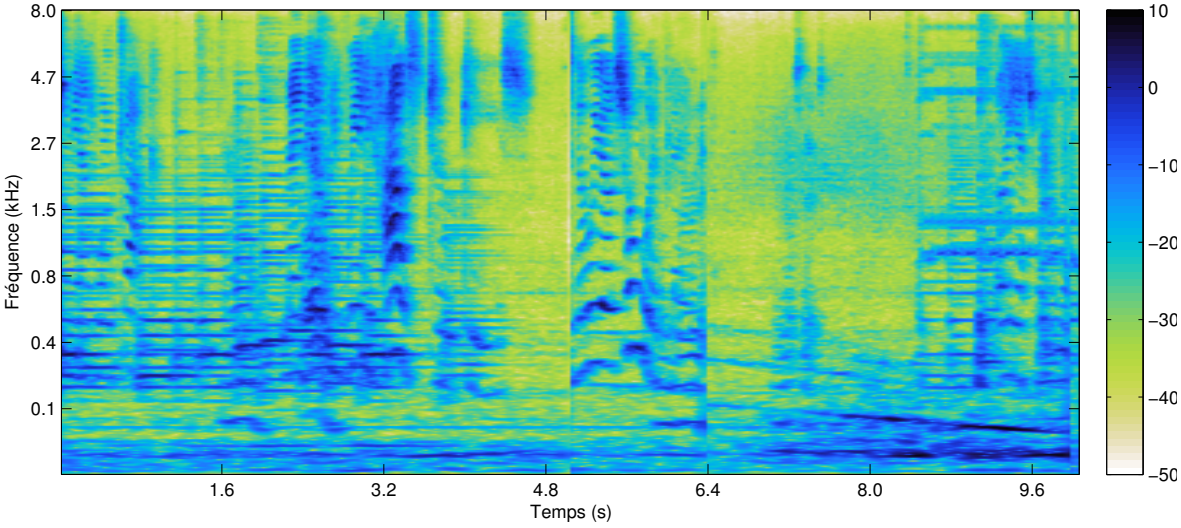
L'estimation des paramètres $w_{2,fk}$ et $h_{2,kn}$ est effectuée par mises à jour multiplicatives, comme présenté dans la partie 2.1.3.

La NMF étant particulièrement sensible à l'initialisation, le dictionnaire $\mathbf{W}_2 = \{w_{2,fk}\}_{k=1..K_2}^{f=1..F}$ est d'abord initialisé par quantification vectorielle en K_2 clusters du spectrogramme d'amplitude du signal de bruit seul $|\mathbf{X}_b| = \{|x_{b,fn}|\}_{n=1..N'}^{f=1..F}$, comme illustré sur la figure 3.1b. Les K_2 vecteurs prototypes ainsi estimés composent alors les K_2 colonnes du dictionnaire $\mathbf{W}_2$ à l'initialisation. Les coefficients d'activation $h_{2,fn}$ sont eux tous initialisés à 1.

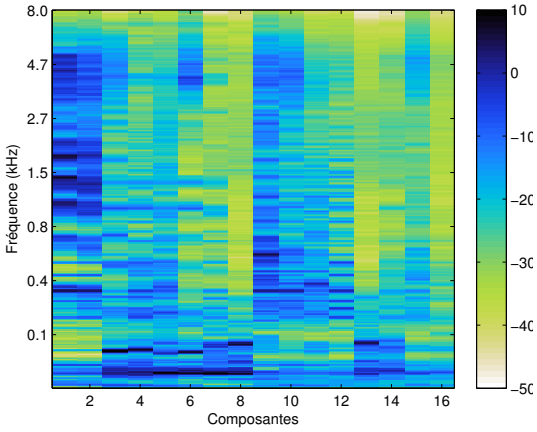
Après optimisation, le dictionnaire $\widetilde{\mathbf{W}}_2$ ainsi appris (figure 3.1c) est retenu comme description spectrale du bruit et sera fixé lors de la séparation du mélange bruité. La matrice d'activation $\widetilde{\mathbf{H}}_2$ apprise (figure 3.1d) sera elle aussi utilisée pour initialiser la matrice d'activation $\mathbf{H}_2$ du bruit lors de la séparation du mélange bruité. Précisément, les lignes de $\widetilde{\mathbf{H}}_2$ seront moyennées afin d'initialiser les lignes de $\mathbf{H}_2$ de sorte que :

$$\forall n \in [1, N]\,,\ \forall k \in [1, K_2]\,,\ h_{2,kn} = \frac{1}{N} \sum_{n'} \tilde{h}_{2,kn'}. \tag{3.30}$$

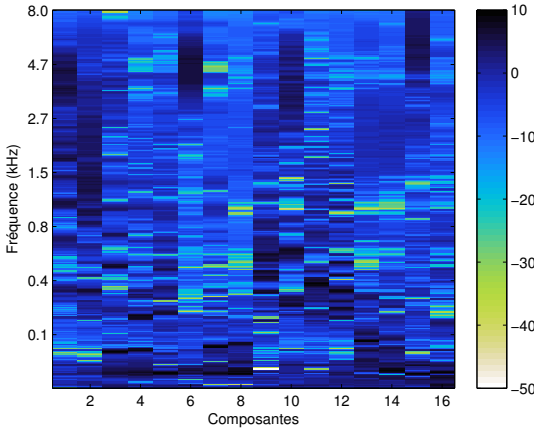
1. http://bass-db.gforge.inria.fr/fasst/



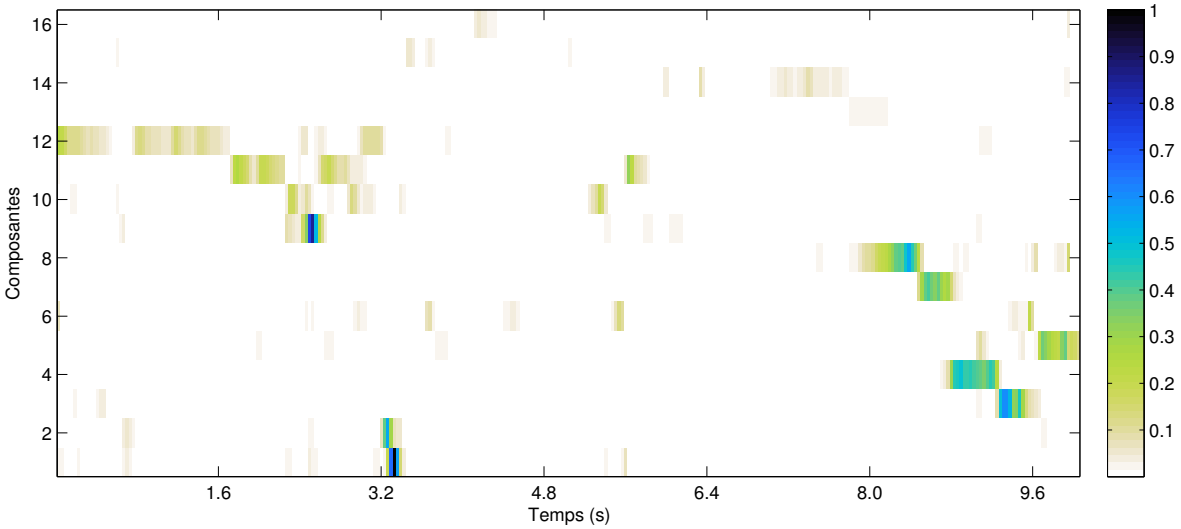
(a) Spectrogramme de log-puissance QERB du signal de bruit seul $x_b(t)$.



(b) Initialisation du dictionnaire $\mathbf{W}_2$ par quantification vectorielle en $K_2 = 16$ composantes.



(c) Dictionnaire $\widetilde{\mathbf{W}}_2$ appris par NMF sur le bruit seul.



(d) Matrice d'activation $\widetilde{\mathbf{H}}_2$ apprise par NMF sur le bruit seul.

FIGURE 3.1 – Étapes de l'apprentissage du modèle de bruit.

3.3.4 Apprentissage du modèle de parole

Comme présenté dans la partie 2.2.1, nous proposons de modéliser la parole par un modèle NMF. Toutefois, contrairement au modèle de bruit, nous envisageons deux formes possibles de factorisation de sa structure spectrale.

Quelque soit la structure spectrale choisie, NMF ou EF (excitation-filtre), pour caractériser un locuteur, nous proposons d'apprendre ses caractéristiques à l'aide de l'ensemble d'apprentissage défini plus tôt. Dans le cadre de nos expériences, nous supposerons que le locuteur à séparer est connu. Nous pourrons donc pour chaque mélange choisir le modèle dédié à ce locuteur.

Pour un locuteur donné, l'apprentissage est réalisé sur l'ensemble des occurrences de ce locuteur présentes dans la base d'apprentissage, soit un total de 500 séquences de mots par locuteur. Ces séquences sont concaténées afin de ne former qu'une seule observation que nous noterons $x_p(t)$. Notons bien que cette observation n'est composée que des séquences de mots prononcées par un seul et même locuteur, sans bruit et avec réverbération. Cette observation stéréo est ensuite convertie en une observation mono en sommant les spectrogrammes de puissance gauche et droit, comme justifié plus tôt.

Structure spectrale de type NMF simple

Pour le cas d'une structure spectrale de type NMF simple, la variance de la source de parole est exprimée par

$$v_{1,fn} = \sum_{k=1}^{K_1} w_{1,fk} h_{1,kn}. \tag{3.31}$$

L'apprentissage des paramètres $w_{1,fk}$ et $h_{1,kn}$ est mené de manière similaire au modèle de bruit. Le dictionnaire $\mathbf{W}_1$ est d'abord initialisé par quantification vectorielle en K_1 clusters du spectrogramme d'amplitude de l'observation $|\mathbf{X}_p| = \{|\mathbf{x}_{p,fn}|\}$. Les coefficients d'activation de $\mathbf{H}_1$ sont eux initialisés à 1. Ces paramètres sont ensuite mis à jour à l'aide des formules multiplicatives présentées dans la partie 2.1.3, relativement à l'observation x_p formée des séquences concaténées propres à ce locuteur. Comme illustré sur la figure 3.2a, le dictionnaire $\widetilde{\mathbf{W}_1}$ ainsi estimé sera alors utilisé comme description spectrale du locuteur lors de la séparation dans un mélange bruité. La matrice d'activation correspondante $\mathbf{H}_1$ sera quant à elle initialisée avec la moyenne des valeurs de la matrice d'activation $\widetilde{\mathbf{H}_1}$ apprise de sorte que :

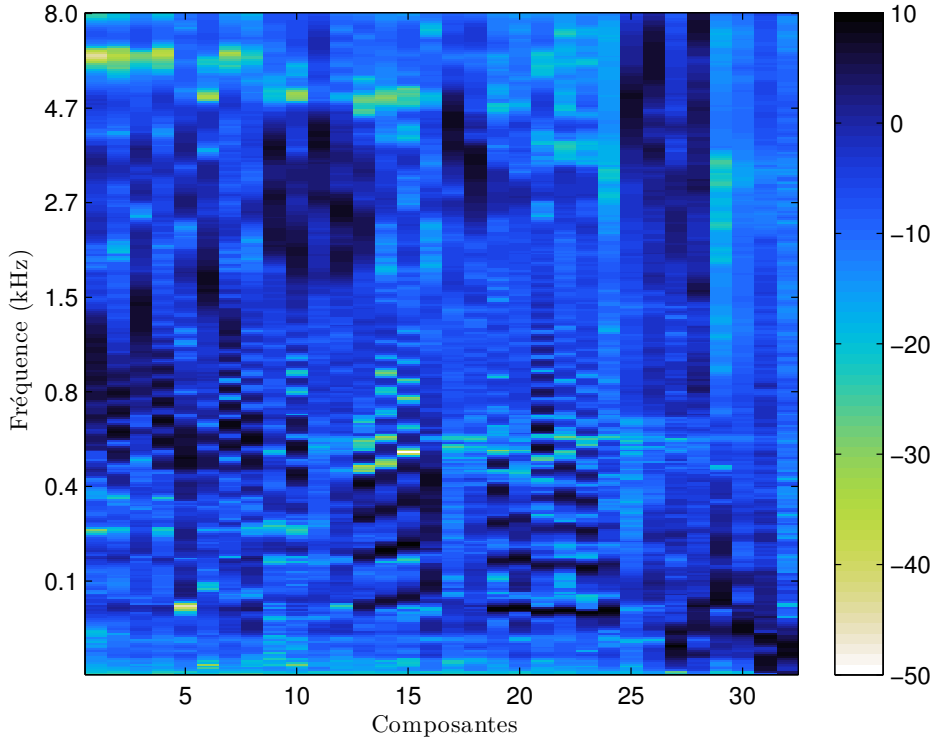
$$\forall n \in [1, N]\,,\ \forall k \in [1, K_1]\,,\ h_{1,kn} = \frac{1}{N} \sum_{n'} \tilde{h}_{1,kn'}. \tag{3.32}$$

Structure spectrale de type Excitation-Filtre (EF)

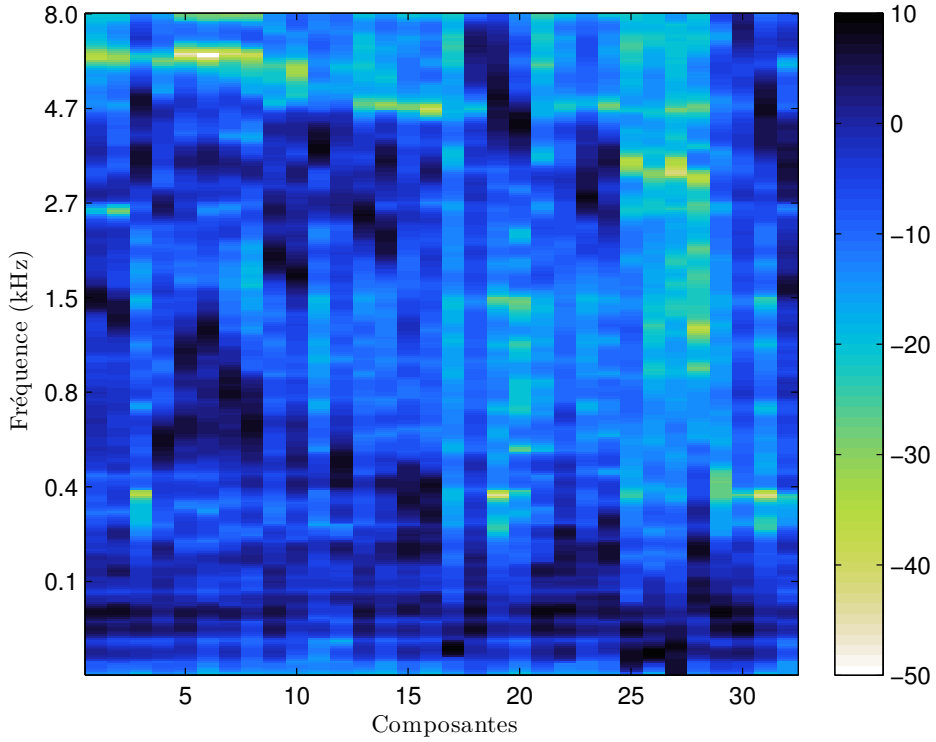
Pour le cas de la structure EF, la variance de la source de parole est exprimée par

$$v_{1,fn} = \left(\sum_{k=1}^{K_1^{\mathsf{ex}}} w_{1,fk}^{\mathsf{ex}} h_{1,kn}^{\mathsf{ex}} \right) \left(\sum_{k=1}^{K_1^{\mathsf{ft}}} \sum_{k'=1}^{K_1'^{\mathsf{ft}}} b_{1,fk'}^{\mathsf{ft}} u_{1,k'k}^{\mathsf{ft}} h_{1,kn}^{\mathsf{ft}} \right). \tag{3.33}$$

Nous procédons de manière similaire au cas NMF simple pour l'apprentissage du modèle EF. Le dictionnaire de la partie excitation $\mathbf{W}_1^{\mathsf{ex}}$ est fixé grâce au modèle *KLGLOTT88* [Klatt et Klatt, 1990]. Afin de prendre en compte les composantes non-voisées de la parole, une colonne de bruit blanc, représentée par des uns, est ajoutée au dictionnaire de l'excitation. La matrice d'activation correspondante $\mathbf{H}_1^{\mathsf{ex}}$ est initialisée avec des uns.



(a) Modèle de NMF simple appris : dictionnaire $\widetilde{\mathbf{W}}_1$.



(b) Modèle EF appris : dictionnaire de la partie filtre $\widetilde{\mathbf{W}}_1^{\text{ft}} = \widetilde{\mathbf{B}}_1^{\text{ft}} \widetilde{\mathbf{U}}_1^{\text{ft}}$.

FIGURE 3.2 – Modèles de voix appris pour le locuteur 10 du corpus *CHiME*.

Côté filtre, le dictionnaire de spectres à bande étroite $\mathbf{B}_1^{\text{ft}}$ est fixé avec $K'^{\text{ft}}_1 = 230$. Sa matrice d'activation associée $\mathbf{U}_1^{\text{ft}}$ est initialisée au moyen d'une quantification vectorielle en K'^{ft}_1 clusters de la matrice des MFCCs de l'observation, comme expliqué en annexe B. Enfin, la matrice d'activation des filtres lisses $\mathbf{H}_1^{\text{ft}}$ est initialisée avec des 1.

Par application des mises à jours décrites dans la partie 2.1.3, les matrices $\widetilde{\mathbf{H}}_1^{\text{ex}}$, $\widetilde{\mathbf{U}}_1^{\text{ft}}$ et $\widetilde{\mathbf{H}}_1^{\text{ft}}$ sont alors estimées relativement à l'observation x_p formée des séquences concaténées propres à ce locuteur. Comme illustré par la figure 3.2, ces matrices apprises seront utilisées pour initialiser les matrices du modèle de parole lors de la séparation du mélange bruité. Précisément, les valeurs apprises de la matrices $\widetilde{\mathbf{U}}_1^{\text{ft}}$ seront fixées de sorte que le produit $\widetilde{\mathbf{W}}_1^{\text{ft}} = \widetilde{\mathbf{B}}_1^{\text{ft}}\widetilde{\mathbf{U}}_1^{\text{ft}}$ décrive les caractéristiques spectrales des filtres relatifs au locuteur. Les matrices d'activation $\mathbf{H}_1^{\text{ex}}$ et $\mathbf{H}_1^{\text{ft}}$ seront elles initialisées par la moyenne de leurs valeurs à l'apprentissage de sorte que :

$$\begin{aligned} \forall n \in [1, N], \quad \forall k \in \left[1, K_1^{\text{ft}}\right], \quad & h_{1,kn}^{\text{ft}} = \tfrac{1}{N} \textstyle\sum_{n'} \tilde{h}_{1,kn'}^{\text{ft}}, \\ \forall k \in [1, K_1^{\text{ex}}], \quad & h_{1,kn}^{\text{ex}} = \tfrac{1}{N} \textstyle\sum_{n'} \tilde{h}_{1,kn'}^{\text{ex}}. \end{aligned} \tag{3.34}$$

3.3.5 Performances individuelles de séparation

Nous nous intéressons dans cette partie à une tâche de rehaussement de parole. Par conséquent, parmi les sources que nous cherchons à séparer, nous portons notre intérêt sur la source de parole uniquement. Afin d'évaluer le potentiel des règles de fusion proposées, nous devons dans un premier temps définir un ensemble de séparateurs dont nous étudierons par la suite la fusion. Pour cette tâche, nous nous bornerons à étudier des séparateurs qui ne varient que par la structure spectrale du modèle de parole et par son nombre de paramètres.

Paramètres et hyperparamètres

Comme précisé en introduction, tous les séparateurs que nous envisagerons dans nos expériences disposeront du même modèle de bruit. Ce modèle de bruit, est un modèle NMF dont le dictionnaire est appris pour chaque mélange $x(t)$ sur les sections sans parole qui précédent et suivent la séquence de mots à séparer. Le seul hyperparamètre à choisir est le nombre de composantes de ce modèle NMF. Pour toutes nos expériences, nous avons fixé ce nombre de composantes à $K_2 = 16$.

Pour le modèle de parole présenté dans la partie 3.3.4, nous allons envisager plusieurs variations de paramètres. D'ores et déjà, nous savons que la variance de la source de parole peut être modélisée soit par une structure spectrale de type NMF, soit par une structure spectrale de type EF. Par ailleurs, nous proposons dans nos expériences de faire varier la taille des matrices utilisées pour ces différentes structures. En effet, il a déjà été montré que la qualité de séparation dépendait du nombre de composantes de la NMF [Bertin et al., 2007]. De la même manière, nous montrerons ici que la dimension de la factorisation EF est elle aussi déterminante pour la qualité de séparation.

Ainsi, pour la structure spectrale de type NMF, nous étudierons l'influence du nombre de composantes pour des valeurs variant de $K_1 = 2$ à $K_1 = 128$ par puissance de 2, soit $M = 7$ nombres de composantes différents tels que $K_{1m} = 2^m$ avec $m \in [1, M]$. De la même manière, nous étudierons l'influence du nombre de composantes de la partie filtre K_1^{ft} dans le cas d'une structure spectrale de type EF. Nous envisagerons également $M = 7$ nombres de composantes distincts tels que $K_{1m}^{\text{ft}} = 2^m$ avec $m \in [1, M]$. Le tableau 3.1 reprécise les caractéristiques de l'intégralité des modèles que nous allons étudier, et propose pour chacun un nom court dans la colonne *Désignation* qui permettra de les identifier plus simplement dans la suite de ce document.

Enfin, notons que les paramètres d'analyse ont eux été gardés fixes tout au long de nos expériences. Nous avons utilisé la transformée QERB avec une fenêtre sinusoïdale de 1024 échantillons et $F = 350$ bandes de fréquence.

Désignation	Modèle de parole		Modèle de bruit	
	Structure spectrale	Nombre de composantes	Structure spectrale	Nombre de composantes
NMF-2	NMF	2	NMF	16
NMF-4		4		
NMF-8		8		
NMF-16		16		
NMF-32		32		
NMF-64		64		
NMF-128		128		
EF-2	EF	2		
EF-4		4		
EF-8		8		
EF-16		16		
EF-32		32		
EF-64		64		
EF-128		128		

TABLEAU 3.1 – Désignations et caractéristiques des séparateurs envisagés pour le rehaussement de la parole.

Évaluation

Nous proposons d'évaluer la performance des séparateurs à l'aide des mesures objectives introduites dans la partie 2.1.5. Nous nous intéressons ici uniquement à la qualité des signaux de parole estimés. Les mesures ont été moyennées sur chacun des ensembles de validation et de test. Nous discutons ci-après ces résultats en détail. Les valeurs numériques peuvent être trouvées dans le tableau 3.2

Influence de la structure spectrale et du nombre de composantes

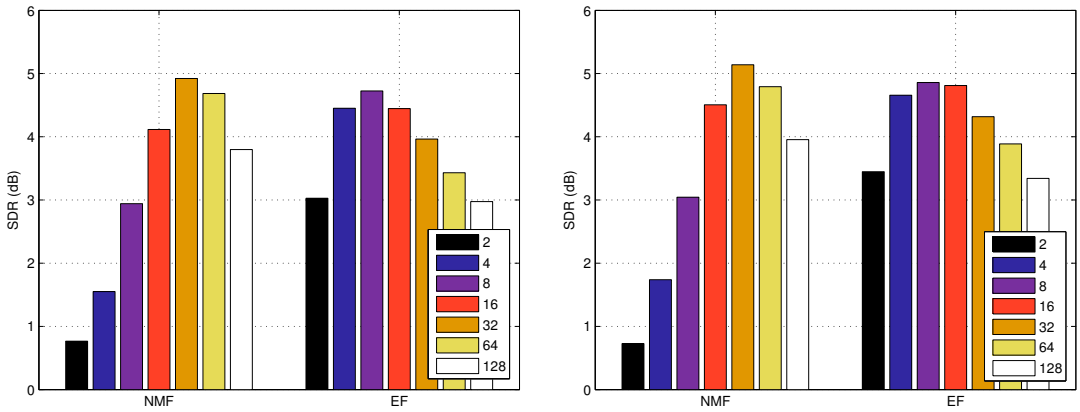
La figure 3.3 représente le SDR des signaux de voix estimés pour l'ensemble des $M = 14$ modèles considérés, pour l'ensemble de validation à la figure 3.3a et l'ensemble de test à la figure 3.3b. Comme nous pouvons le constater, la performance des séparateurs dépend à la fois de la structure spectrale, NMF ou EF, et du nombre de composantes, et ce quelque soit l'ensemble de données. Notons d'ailleurs que les résultats entre ensemble de validation et ensemble de test sont très similaires, ce qui souligne la bonne homogénéité de ces deux ensembles ainsi que la bonne généralisation des modèles de parole et de bruit que nous avons appris.

Dans le cas d'une structure NMF, le meilleur SDR moyen est obtenu pour un nombre de composantes $K_1 = 32$, soit un SDR de 4.92 dB pour l'ensemble de validation et un SDR de 5.14 dB pour l'ensemble de test. Pour des nombres de composantes de 2 à 32, le SDR de la source de voix croît de façon monotone, puis semble décroître de la même manière pour les nombres de composantes supérieurs à 32.

Pour le cas d'une structure EF, le meilleur SDR moyen est obtenu pour $K_1^{\text{ft}} = 8$, soit un SDR de 4.72 dB pour l'ensemble de validation et un SDR de 4.86 dB pour l'ensemble de test. De même que pour le cas d'une structure NMF, le SDR croît de façon monotone pour les nombres de composantes inférieurs à 8 puis décroît pour les nombres de composantes supérieurs.

Il est à noter que le modèle *NMF-32* est le meilleur séparateur en moyenne sur nos deux ensembles de données. Ce séparateur apporte un gain moyen de 0.3 dB (respectivement, 0.2 dB) sur l'ensemble de test (respectivement, l'ensemble de validation) par rapport au meilleur modèle

de type EF.



(a) Ensemble de validation (b) Ensemble de test

FIGURE 3.3 – Performance (SDR) individuelle des séparateurs en fonction de la structure spectrale et du nombre de composantes.

La figure 3.4 représente les performances des séparateurs en fonction de la structure spectrale considérée et de son nombre de composantes. Cette fois-ci, les résultats ont été tracés indépendamment pour chacun des rapports signal-à-bruit du corpus et ont été moyennés sur les ensembles de validation et de test, soit un total de 68 mélanges pour chaque RSB. Nous nous permettons ici de moyenner les performances des ensembles de validation et de test puisque leurs performances respectives sont comparables. On constate que quelque soit la structure spectrale adoptée, le meilleur nombre de composantes dépend du RSB. On peut aussi constater que logiquement, plus le RSB est grand, plus le SDR de la source de voix estimée est élevé. Ainsi, pour un RSB de -6 dB, le meilleur séparateur est le modèle *NMF-8* alors que pour le RSB de 9 dB, les meilleurs résultats sont obtenus pour le séparateur *EF-64*. En outre, il semble que plus le RSB est grand, plus le nombre de composantes optimal est lui aussi grand. De ce fait, les grands nombres de composantes, supérieurs à 32, montrent de piètres performances (SDRs négatifs) pour les mélanges à -6 dB de RSB alors que ces mêmes nombres de composantes sont les plus performants pour les exemples aux RSBs plus élevés.

Le fait que des modèles simples avec peu de composantes soient plus à même de distinguer la parole dans un bruit de niveau élevé n'est pas surprenant. En effet, dans un mélange très bruité, si l'on considère que la taille du modèle de bruit est fixée comme c'est le cas dans nos expériences, certaines composantes du modèle de parole risquent de contribuer à la modélisation du bruit au détriment de la parole. Lorsque le modèle de parole a moins de composantes, ce risque est plus faible. En d'autres termes, un modèle de parole avec peu de composantes est plus robuste au bruit. On peut d'ailleurs constater que les performances des modèles NMF avec peu de composantes (c'est-à-dire 2, 4 et 8) varient peu en fonction du RSB.

Enfin, le SIR et le SAR moyennés sur les ensembles de validation et de test ont été reportés sur la figure 3.5. On notera en particulier que comme le montre la figure 3.5b les modèles qui génèrent le moins d'artefacts (c'est-à-dire, ceux dont le SAR est le plus élevé) sont les modèles ayant le plus de composantes, que ce soit pour une structure spectrale NMF (*NMF-128*) ou EF (*EF-128*). Ce résultat semble logique puisqu'avec un nombre de composantes élevé, il est plus aisé de représenter la richesse spectrale d'une voix. Les nombres de composantes plus faibles donnent en effet une représentation plus grossière de la parole qui peut alors expliquer la création d'artefacts. A l'inverse, le SIR représenté sur la figure 3.5a a tendance à être plus faible pour les nombres de composantes élevés que pour les petits nombres de composantes. Comme nous l'avons expliqué

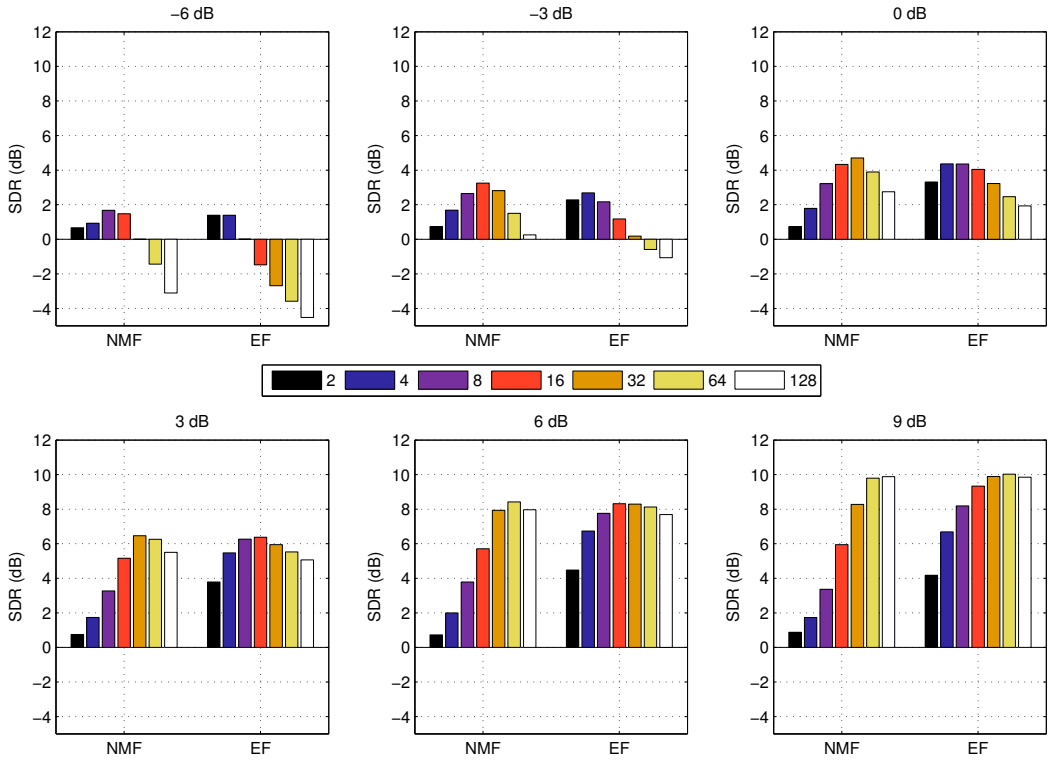


FIGURE 3.4 – Performance (SDR) individuelle des séparateurs en fonction du rapport signal-à-bruit du mélange, moyennée sur les ensembles de test et de validation.

plus tôt, un modèle avec un grand nombre de composantes est capable de décrire avec précision le contenu spectral d'une voix, au risque de parfois contribuer aussi à la description du bruit. Ce phénomène donne alors naissance aux interférences et engendre donc un SIR plus faible.

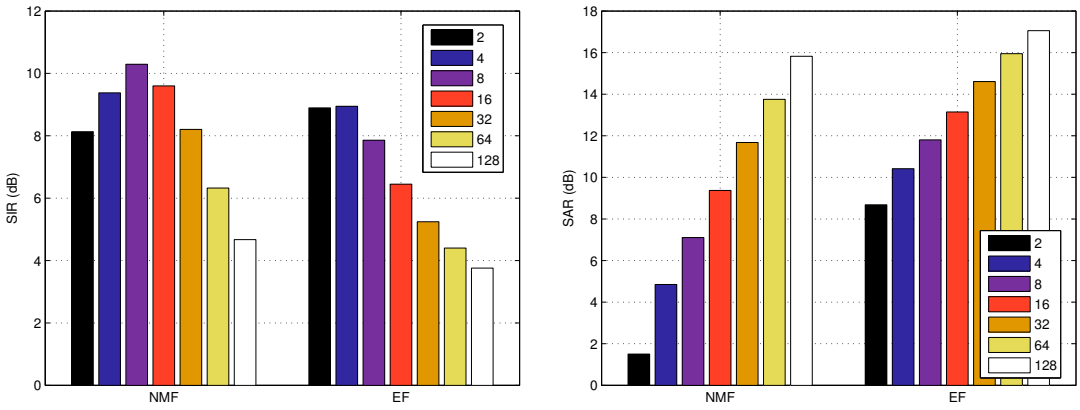
3.3.6 Performance oracle de fusion

Nous présentons ci-après les résultats oracles des cas particuliers de fusion introduits dans la partie 3.2. Ces résultats ont été obtenus en résolvant les problèmes PQ définis dans la partie 3.2.3. Comme pour les performances individuelles, nous ne nous intéressons ici qu'à la qualité des signaux de parole estimés. Nous discutons plus en détail ces résultats ci-après. Toutes les valeurs numériques des SDRs ont été reportées dans le tableau 3.3.

Fusion oracle invariante

Les SDRs, SIRs et SARs moyennés sur les ensembles de validation et de test sont représentés sur la figure 3.6 pour le cas des fusions oracles invariantes de modèles NMF uniquement, de modèles EF uniquement ou de modèles NMF et EF, et ce pour tous les nombres de composantes que nous avons proposés plus tôt. Les coefficients de fusion oracle invariante sont obtenus en résolvant le problème PQ (3.16). Ces résultats de fusion représentés par les barres blanches de la figure 3.6 sont comparés aux résultats de sélection de modèle représentés par les barres noires.

Ces premiers résultats nous permettent de confirmer l'intérêt de la fusion par rapport à une simple sélection pour notre problème de séparation de sources. En effet, quelque soit les modèles fusionnés (NMF, EF, NMF et EF), le SDR de fusion oracle invariante est toujours supérieur au SDR de sélection oracle invariante. Les gains sont respectivement de 0.66 dB, 0.38 dB et 0.72 dB de SDR pour la fusion de modèles NMF, de modèles EF et de modèles NMF et EF. On constate



(a) SIR (b) SAR

FIGURE 3.5 – Performance (SIR et SAR) individuelle des séparateurs en fonction de la structure spectrale et du nombre de composantes, moyennée sur les ensembles de test et de de validation.

Modèle	SDR		ISR		SIR		SAR	
	valid	test	valid	test	valid	test	valid	test
NMF-2	0.77	0.73	1.00	0.98	8.74	7.52	1.68	1.33
NMF-4	1.55	1.74	2.39	2.45	9.07	9.68	4.72	4.98
NMF-8	2.94	3.04	4.50	4.63	**10.21**	**10.38**	6.88	7.34
NMF-16	4.11	4.51	7.28	7.60	9.35	9.85	9.27	9.47
NMF-32	**4.92**	**5.14**	10.78	10.83	8.04	8.36	11.52	11.83
NMF-64	4.69	4.79	13.95	13.57	6.17	6.48	13.76	13.74
NMF-128	3.80	3.95	**16.31**	**16.43**	4.60	4.75	**15.70**	**15.94**
EF-2	3.03	3.45	5.61	5.67	8.43	**9.35**	8.64	8.73
EF-4	4.45	4.66	8.72	8.44	**8.67**	9.22	10.45	10.38
EF-8	**4.72**	**4.86**	10.88	10.59	7.60	8.12	11.77	11.83
EF-16	4.45	4.81	12.75	12.85	6.12	6.78	13.06	13.24
EF-32	3.96	4.32	14.42	14.42	5.00	5.49	14.53	14.69
EF-64	3.43	3.89	15.42	15.66	4.16	4.64	15.77	16.13
EF-128	2.97	3.34	**16.11**	**16.27**	3.58	3.93	**16.91**	**17.20**

TABLEAU 3.2 – Performance individuelle (SDR, ISR, SIR et SAR) des séparateurs envisagés pour le rehaussement de la parole, calculée sur les sources de parole estimées sur les ensembles de validation et de test

donc que la fusion de modèles EF uniquement est la moins prometteuse en terme de qualité globale de séparation et que les fusions de modèles NMF et de modèles NMF et EF ont à peu près le même potentiel comparées aux approches par sélection correspondantes. Toutefois, la fusion de modèles NMF et EF atteint une performance moyenne plus élevée, soit un SDR de 7.96 dB contre un SDR de 7.42 dB pour la fusion de modèles NMF uniquement.

Nous pouvons aussi constater que nos méthodes de fusion permettent de réduire significativement les artefacts comparativement à la simple sélection. Toutefois, comme nous l'avions déjà noté dans l'analyse des résultats individuels dans la partie 3.3.5, cette diminution des artefacts se fait au détriment des interférences qui sont moins présentes lorsqu'on ne sélectionne qu'un seul séparateur.

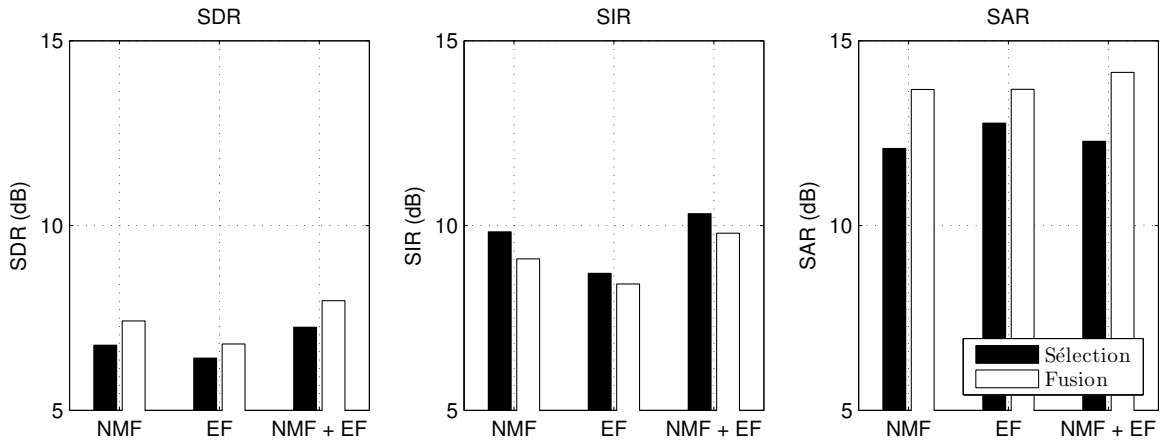


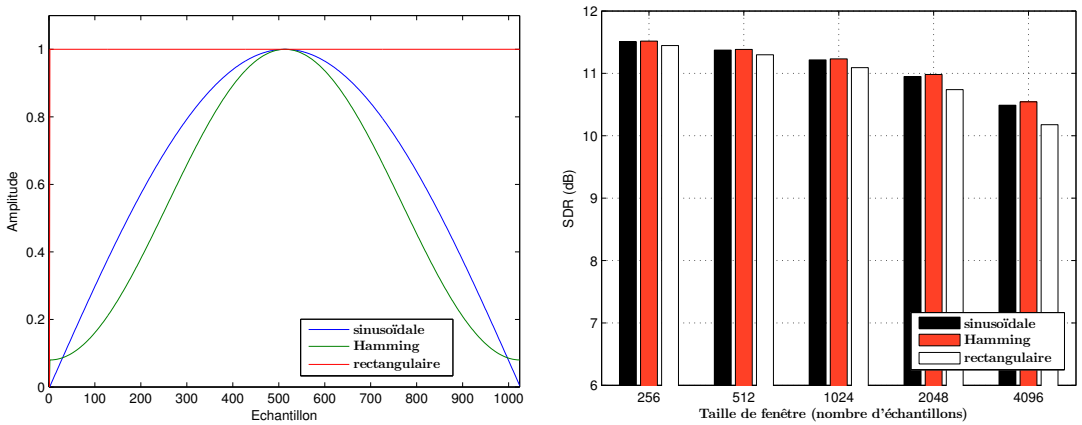
FIGURE 3.6 – Performance (SDR, SIR, SAR) de fusion oracle invariante, en fonction des modèles fusionnés et en moyenne sur les ensembles de validation et de test.

Fusion oracle variant en temps

Les résultats de fusion oracle variant en temps sont obtenus par la résolution du problème PQ défini en (3.19) sur chacune des trames du mélange. Il convient donc dans un premier temps de choisir une fenêtre d'analyse afin de découper en trames les signaux du problème. Nous avons ici envisagé trois types de fenêtre, à savoir une fenêtre sinusoïdale, une fenêtre de Hamming et une fenêtre rectangulaire. Toutes trois sont représentées sur la figure 3.7a pour une taille de 1024 échantillons. Nous avons par ailleurs calculé les coefficients de fusion oracle variant en temps pour plusieurs tailles de fenêtre, à savoir 256, 512, 1024, 2048 et 4096 échantillons. A chaque fois, nous avons considéré un recouvrement des fenêtres adjacentes de 50%. Les SDRs obtenus pour le cas de la fusion de modèles NMF et EF sont représentés sur la figure 3.7 pour chaque couple type/taille de fenêtre. Ces mesures ont été moyennées sur tous les exemples des ensembles de validation et de test.

Comme nous pouvions nous y attendre, nous constatons que plus la fenêtre est petite, plus le SDR est grand. Ceci s'explique très simplement en notant que plus la fenêtre est petite, plus le nombre de trames est élevé. En conséquence, le nombre de coefficients de fusion est plus grand et la fusion opère avec une meilleure précision. Entre la plus petite fenêtre (256 échantillons) et la plus grande fenêtre (4096 échantillons), la différence de SDR est de 1 dB environ quelque soit le type de fenêtre choisi. Enfin, on remarque que le choix du type de fenêtre n'a pas une influence capitale sur la qualité de séparation. Fenêtre de Hamming et sinusoïdale donnent des résultats très proches. Seule la fenêtre rectangulaire engendre une perte de 0.1 à 0.3 dB de SDR environ.

La figure 3.8 compare les résultats de fusion oracle variant en temps et de sélection oracle variant



(a) Forme des fenêtres d'analyse

(b) SDR (dB) en fonction du type et de la taille de la fenêtre d'analyse

FIGURE 3.7 – Influence de la fenêtre d'analyse sur la performance (SDR) de fusion oracle variant en temps. Le SDR est moyenné sur les ensembles de validation et de test, pour le cas de la fusion de modèles NMF et EF.

en temps, pour le cas d'une fenêtre sinusoïdale de 1024 échantillons avec 50% de recouvrement. Comme pour le cas invariant, les résultats de sélection oracle variant en temps sont obtenus en résolvant le même problème PQ (3.19) avec la contrainte supplémentaire que seul un coefficient de fusion $\alpha_{m,n}$ peut être non-nul et donc égal à 1, et ce pour tout n. Ici encore, la fusion est plus performante que la sélection puisque qu'elle permet un gain de 0.32 dB pour la fusion de modèles NMF et la fusion de modèles NMF et EF, et un gain de 0.14 dB pour la fusion de modèles EF uniquement. Il convient enfin de noter que les SIR et SAR suivent le même comportement que pour le cas invariant.

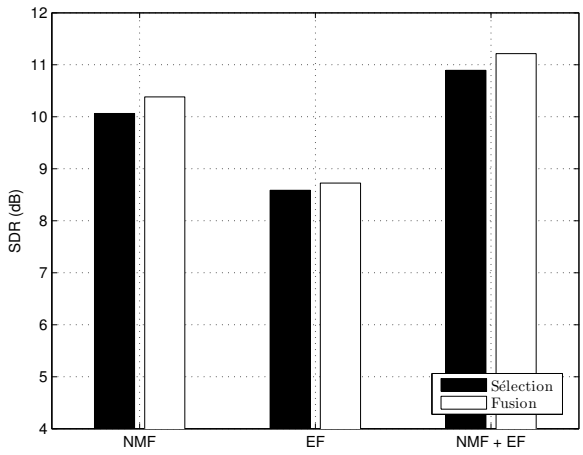


FIGURE 3.8 – Performance (SDR) de fusion oracle variant en temps et sélection oracle variant en temps, en fonction des modèles fusionnés et en moyenne sur les ensembles de validation et de test.

Fusion oracle variant en fréquence

Les résultats de fusion oracle variant en fréquence sont obtenus par la résolution du problème PQ défini en (3.23) sur chacune des bandes de fréquence. Il convient donc dans un premier temps de calculer les versions filtrées des signaux en jeu dans le problème PQ (3.23), à savoir les sources

estimées $\tilde{s}_{jm}^{f}(t)$ et les sources vraies $s_{j}^{f}(t)$. Comme indiqué dans la partie 3.2.1, nous proposons de calculer ces signaux par transformée inverse des signaux temps-fréquence correspondants $\{\tilde{s}_{jm,fn}\}_{n=1..N}$ et $\{s_{jm,fn}\}_{n=1..N}$. Nous avons choisi pour ces expériences d'utiliser une TFCT avec une fenêtre sinusoïdale de 1024 échantillons et 50 % de recouvrement. Ainsi, pour chacun des signaux, nous obtenons $F = 513$ signaux filtrés $\left\{\tilde{s}_{jm}^{f}(t)\right\}_{f=1..F}$ et $\left\{s_{jm}^{f}(t)\right\}_{f=1..F}$.

Afin d'analyser l'effet du nombre de bandes de fréquence sur le résultats de fusion, nous proposons de regrouper certaines bandes des signaux filtrés et d'estimer les coefficients de fusion sur chacun de ces groupes de bandes. Nous avons envisager pour cela trois possibilités :

— regrouper les bandes de manière uniforme,

— regrouper les bandes par octave,

— ou regrouper les bandes par bandes de *Bark*.

Pour le cas uniforme, nous proposons de grouper les signaux filtrés en F_b bandes de telle sorte que les signaux filtrés à fusionner sont exprimés comme :

$$\forall t,\, \forall f_b \in [1, F_b],\, \begin{cases} \forall m,\, \tilde{s}_{jm}^{f_b}(t) = \sum_{f=\frac{(f_b-1)}{F_b}F+1}^{\frac{f_b}{F_b}F} \tilde{s}_{jm}^{f}(t) \\ s_{j}^{f_b}(t) = \sum_{f=\frac{(f_b-1)}{F_b}F+1}^{\frac{f_b}{F_b}F} s_{j}^{f}(t) \end{cases} \tag{3.35}$$

Nous avons testé plusieurs nombres de bandes, à savoir $F_b \in \{2, 4, 8, 16, 32, 64, 513\}$, le cas $F_b = 513$ correspondant à un coefficient de fusion par bande de fréquence de la TFCT. Nous avons aussi calculé des coefficients pour des bandes regroupées par octave et par bandes de Bark, ce qui nous donne respectivement $F_b = 10$ et $F_b = 22$.

La figure 3.9 présente les valeurs des SDRs en fonction du type et du nombre de bandes considérées, selon que l'on fusionne des modèles NMF uniquement, des modèles EF uniquement ou des modèles NMF et EF simultanément. Sans surprise, plus le nombre de bandes à fusionner est grand, plus la performance est élevée. De même que précédemment, les performances sont meilleures pour la fusion de modèles NMF et EF, et moins bonnes pour la fusion de modèles EF uniquement. Enfin, on peut constater que le découpage en bandes de Bark donne de meilleures performances qu'un découpage uniforme avec un nombre de bandes comparable. En effet, quels que soient les modèles fusionnés, le SDR de fusion en 22 bandes de Bark se situe entre le SDR de fusion en 32 bandes uniformes et le SDR de fusion en 64 bandes uniformes. Le même constat peut être fait pour le découpage en 10 bandes d'octave dont le SDR dépasse de quelques centièmes de décibels celui de la fusion en 16 bandes uniformes. Ceci peut sans doute s'expliquer par la le fait que les bandes de Bark sont plus fines dans les basses fréquences, là où l'énergie du signal de parole est la plus importante, et plus grossières dans les hautes fréquences où l'energie du signal de parole est plus faible.

Sur la figure 3.10, nous comparons la fusion oracle variant en fréquence à la sélection variant en fréquence pour un découpage en 513 bandes uniformes. Nous constatons que l'apport de la fusion variant en fréquence par rapport à la simple sélection est plus faible que dans le cas invariant et variant en temps (voir figures 3.6 et 3.8). En effet, le gain apporté par la fusion est respectivement de 0.15, 0.07 et 0.16 dB pour la fusion de modèles NMF, de modèles EF, et de modèles NMF et EF. Ce résultat peut s'expliquer de deux façons. Il se peut que sur certaines bandes de fréquences, tous les séparateurs donnent des estimées très proches. Sur ces bandes de fréquences, ne choisir qu'une estimée ou les fusionner toutes revient quasiment au même. A l'inverse, il se peut que sur certaines bandes de fréquences, seul un séparateur ait de bonnes performances de séparation. Ce faisant, ne sélectionner que ce séparateur pour cette bande de fréquence ou lui attribuer un coefficient fort par rapport aux autres séparateurs revient en pratique quasiment au même.

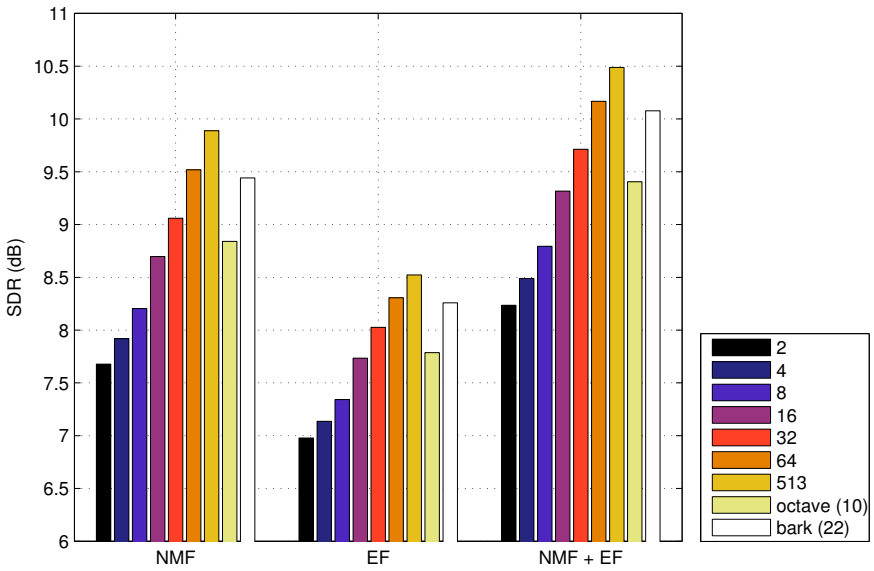


FIGURE 3.9 – Performance (SDR) de fusion oracle variant en fréquence en fonction du nombre de bandes et des modèles fusionnés, en moyenne sur les ensembles de validation et de test.

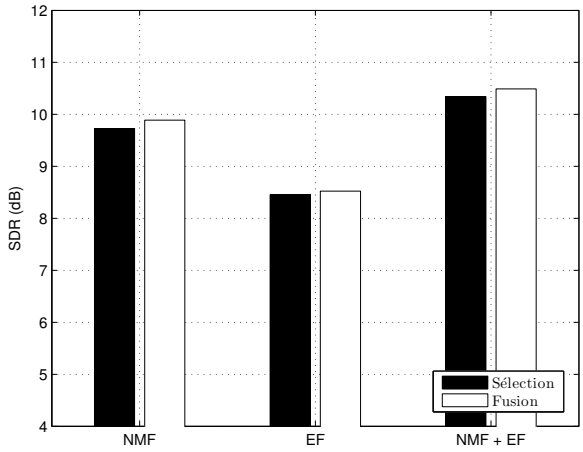


FIGURE 3.10 – Performance (SDR) de fusion oracle variant en fréquence et performance de sélection oracle variant en fréquence, en fonction des modèles fusionnés et en moyenne sur les ensembles de validation et de test.

Type de fusion			Modèles fusionnés					
			NMF		EF		NMF + EF	
Invariante			*valid*	*test*	*valid*	*test*	*valid*	*test*
			7.26	7.58	6.50	7.09	**7.82**	**8.11**
	Fenêtre							
	type	taille	*valid*	*test*	*valid*	*test*	*valid*	*test*
		256	**10.59**	**10.62**	**8.58**	**9.16**	**11.47**	**11.54**
		512	10.50	10.52	8.52	9.09	11.34	11.41
	sinusoïdale	1024	10.37	10.40	8.44	9.01	11.18	11.25
		2048	10.14	10.17	8.31	8.87	10.92	10.98
Variant		4096	9.69	9.77	8.06	8.63	10.44	10.54
			valid	*test*	*valid*	*test*	*valid*	*test*
en		256	**10.60**	**10.63**	**8.59**	**9.17**	**11.48**	**11.55**
		512	10.50	10.53	8.52	9.10	11.35	11.42
temps	Hamming	1024	10.38	10.41	8.45	9.02	11.20	11.27
		2048	10.18	10.20	8.32	8.88	10.95	11.01
		4096	9.73	9.83	8.08	8.65	10.49	10.60
			valid	*test*	*valid*	*test*	*valid*	*test*
		256	**10.55**	**10.58**	**8.55**	**9.13**	**11.41**	**11.48**
		512	10.43	10.46	8.48	9.06	11.27	11.33
	rectangulaire	1024	10.26	10.29	8.38	8.95	11.06	11.12
		2048	9.93	9.98	8.20	8.76	10.70	10.78
		4096	9.37	9.48	7.91	8.47	10.13	10.23
	Nombre de bandes		*valid*	*test*	*valid*	*test*	*valid*	*test*
	2		7.56	7.79	6.69	7.27	8.13	8.34
Variant	4		7.86	7.97	6.86	7.41	8.42	8.56
	8		8.17	8.24	7.07	7.61	8.74	8.85
en	16		8.63	8.76	7.45	8.01	9.23	9.40
	32		9.00	9.12	7.74	8.31	9.65	9.77
fréquence	64		9.48	9.56	8.03	8.58	10.12	10.22
	513		**9.85**	**9.93**	**8.23**	**8.81**	**10.45**	**10.53**
	22 (Bark)		9.39	9.49	7.98	8.53	10.03	10.12
	10 (octave)		8.77	8.91	7.52	8.06	9.35	9.47

TABLEAU 3.3 – Performance (SDR) de fusion oracle pour le rehaussement de la parole, calculée sur les sources de parole fusionnées sur les ensembles de validation et de test.

Conclusion

La figure 3.11 propose enfin de comparer les SDRs , SIRs et SARs des fusions invariante, variant en temps (pour une fenêtre sinusoïdale de 1024 échantillons) et variant en fréquence (pour un découpage en 513 bandes). Quels que soient les modèles fusionnés, la fusion variant en temps est celle qui donne la meilleure performance globale, exprimée par le SDR. Pour le cas de la fusion de modèles NMF et EF, elle permet un gain potentiel de 3.25 dB par rapport à la fusion invariante et de 0.72 dB par rapport à la fusion variant en fréquence. De la même manière, c'est elle qui présente le meilleur SIR, puisqu'il est respectivement supérieur de 5.72 dB et 2.48 dB par rapport aux fusions invariante et variant en fréquence. Toutefois, c'est la fusion variant en fréquence qui donne le meilleur SAR.

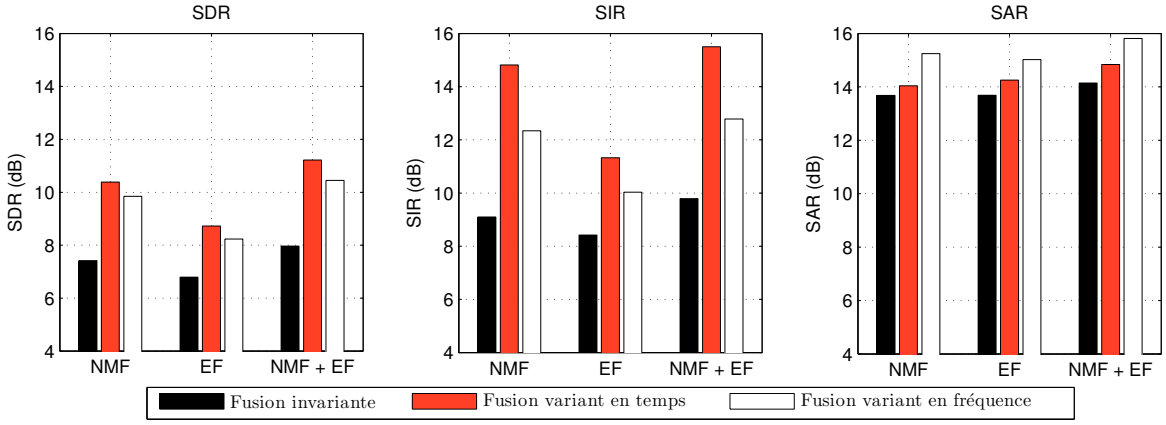


FIGURE 3.11 – Performance (SDR, SIR et SAR) de fusions oracles invariante, variant en temps et variant en fréquence pour la fusion de modèles NMF, de modèles EF et de modèles NMF et EF, en moyenne sur les ensembles de validation et de test.

Dans la suite de ce rapport, nous proposerons plusieurs solutions afin d'appliquer en pratique ces méthodes de fusion. Même si la fusion variant en temps est celle qui donne les meilleurs résultats oracles, nous verrons en pratique qu'elle est plus difficile à mettre en œuvre que les fusions invariante et variant en fréquence.

3.4 Fusion hétérogène : application à l'extraction de voix chantée

Dans cette partie, nous introduisons un deuxième cadre expérimental, complémentaire du premier, et portant cette fois sur l'extraction de voix chantée. Comme introduit dans la partie 2.2.2, le problème d'extraction de voix chantée consiste à séparer un signal de voix que nous noterons $\mathbf{s}_1(t)$ d'un signal d'accompagnement musical $\mathbf{s}_2(t)$ à partir de l'observation de leur mélange $\mathbf{x}(t)$. Nous proposons d'étudier la fusion de quatre méthodes de la littérature initialement utilisées dans [LIUTKUS et al., 2014] sur le corpus que nous nous proposons d'exploiter. Ce corpus sera présenté dans la partie 3.4.1 et les paramètres des quatre séparateurs ici envisagés, dont les principes ont déjà été introduits dans la partie 2.2.2, seront décrits dans la partie 3.4.2. Ces séparateurs mettent en œuvre des méthodes bien distinctes pour résoudre le problème posé. C'est pourquoi nous parlons ici de *fusion hétérogène*. Comme pour le cas homogène, nous étudierons les performances individuelles des séparateurs dans la partie 3.4.3 puis les performances de fusion oracle dans la partie 3.4.4.

3.4.1 Corpus ccMixter

Le corpus *ccMixter* a été initialement proposé dans [Liutkus et al., 2014]. Il est composé de 49 morceaux de musique complets tirés du site communautaire de *remix* nommé *ccMixter*[2]. Tous les morceaux sélectionnés sont en stéréo et sont des productions semi-professionnelles dans des genres musicaux divers. Nous proposons en annexe C un descriptif succinct de chacun des morceaux. L'objectif initial de ce corpus était d'évaluer différentes méthodes d'extraction de voix chantée.

Description

Originellement, chacun des morceaux a été séparé à l'aide de quatre algorithmes de séparation distincts [Durrieu et al., 2011; Huang et al., 2012; Liutkus et al., 2014; Rafii et Pardo, 2012] dont les principes ont été présentés dans la partie 2.2.2. Les séparations ont été effectuées sur des extraits non-recouvrants de 30 secondes maximum. Pour chacun des extraits de chaque morceau, nous disposons donc :

— du mélange original $\mathbf{x}(t)$,

— du signal de voix chantée $\mathbf{s}_1(t)$ et du signal d'accompagnement musical $\mathbf{s}_2(t)$ vrais qui ont permis d'obtenir le mélange original, selon un modèle de mélange linéaire, $\forall t, \mathbf{x}(t) = \mathbf{s}_1(t) + \mathbf{s}_2(t)$,

— des signaux de voix et d'accompagnement musical estimés pour chaque algorithme de séparation, que nous noterons $\tilde{\mathbf{s}}_{1m}(t)$ et $\tilde{\mathbf{s}}_{2m}(t)$ pour $m \in [1, 4]$.

Tous ces signaux sont échantillonnés à 44.1 kHz.

Organisation

Pour nos expériences, nous avons conservé un total de 208 extraits. En effet, certains extraits de moins de 20 secondes ont été mis de côté afin d'homogénéiser le corpus. Par ailleurs, les 49 morceaux ont été répartis aléatoirement en 5 groupes de taille similaire afin de pouvoir constituer par la suite des ensembles d'apprentissage, de validation et de test, selon le principe de validation croisée.

3.4.2 Séparateurs envisagés

Comme indiqué en introduction, nous proposons de résoudre le problème d'extraction de voix chantée à partir de quatre modèles intrinsèquement différents, à savoir : le modèle de mélange instantané (IMM pour *Instantaneous Mixture Model*) [Durrieu et al., 2011], l'analyse en composantes principales robuste (RPCA pour *Robust Principal Component Analysis*) [Huang et al., 2012], la technique d'extraction par similarité de motifs répétés (REPETsim pour *Repeating Pattern Extraction Technique based on similarity*) [Rafii et Pardo, 2012] et le modèle additif à noyaux (KAM pour *Kernel Additive Model* [Liutkus et al., 2014]. Pour chacun de ces séparateurs, nous précisons ci-après les paramètres choisis, qui sont les paramètres par défaut des algorithmes distribués sur les sites des auteurs.

IMM

L'implémentation du séparateur IMM que nous avons utilisée est disponible sur le site[3] associé à l'article [Durrieu et al., 2011]. Par défaut, l'algorithme utilise une TFCT calculée avec une

2. http://www.ccmixter.org/
3. http://www.durrieu.ch/research/jstsp2010.html

fenêtre sinusoïdale de 2048 échantillons pour un recouvrement de 256 échantillons, soit 87.5 %. Le modèle d'accompagnement musical a $K_2 = 40$ composantes. Pour le modèle de voix chantée, le dictionnaire de l'excitation a été construit selon le modèle *KLGLOTT88* [Klatt et Klatt, 1990] pour couvrir un intervalle de hauteur entre 100 et 800 Hz et compte 20 spectres harmonique par demi-ton. Pour la partie filtre, le dictionnaire $\mathbf{W}_1^{\text{ft}} = \mathbf{B}_1^{\text{ft}}\mathbf{U}_1^{\text{ft}}$ a $K_1^{\text{ft}} = 10$ filtres, chaque filtre étant exprimé comme la combinaison de $K'^{\text{ft}}_1 = 30$ formes spectrales lisses. L'estimation suit les trois étapes proposées par l'auteur et les signaux estimés sont obtenus par filtrage de Wiener du mélange.

RPCA

Une implémentation de la RPCA [Huang et al., 2012] est disponible sur le site de l'auteur[4]. Notons que, le code fourni ne prenant pas en compte les mélanges stéréophoniques, nous avons appliqué la même méthode aux canaux droit et gauche indépendamment. L'algorithme utilise par défaut une TFCT avec fenêtre de Hann de 1024 échantillons et recouvrement de 75 %. La matrice à décomposer, notée $\mathbf{M}$ dans l'équation (2.53), est le spectrogramme d'amplitude du mélange $\mathbf{M} = \{|x_{fn}|\}$. Le paramètre de compromis λ entre le rang de la matrice $\mathbf{L}$ et la parcimonie de $\mathbf{P}$ est fixé tel que :

$$\lambda = \frac{1}{\sqrt{N}}. \tag{3.36}$$

Enfin, les sources estimées sont obtenues par masquage binaire de la TFCT du mélange.

REPETsim

L'algorithme REPETsim [Rafii et Pardo, 2012] peut être téléchargé sur le site de l'auteur[5]. Les paramètres par défaut incluent notamment une TFCT avec fenêtre de Hamming de 2048 échantillons avec recouvrement de 50 %. Pour chaque trame, il peut être sélectionné un nombre maximal de $\#\mathcal{N}_n = 100$ trames similaires pour calculer la médiane (2.55). Les signaux estimés sont obtenus par filtrage doux du mélange original. De même que pour la RPCA, REPETsim a été développé pour des signaux monophoniques. Par conséquent, nous avons appliqué cette méthode indépendamment aux canaux gauche et droit.

KAM

L'auteur du modèle KAM [Liutkus et al., 2014] met à disposition une implémentation sur son site personnel[6]. Comme nous l'avons évoqué dans la partie 2.2.2, l'accompagnement musical est représenté par deux noyaux : l'un périodique (figure 2.5b) dont la période est fixée par estimation du tempo du mélange et l'autre horizontal (figure 2.5a) de durée égale à 2 secondes. La voix chantée est elle modélisée à l'aide d'un noyau en forme de croix (figure 2.5c) dont la hauteur est fixée par défaut à 15 Hz et la largeur à 20 ms. La TFCT du mélange a été calculée pour une fenêtre de Hamming de taille 90 ms avec un recouvrement de 80 %. Les sources estimées sont obtenues par filtrage de Wiener.

3.4.3 Performances individuelles de séparation

Tout comme pour le rehaussement de la parole, nous présentons ici les performances individuelles des séparateurs envisagés, évaluées à l'aide des mesures objectives introduites dans la partie

4. https://sites.google.com/site/singingvoiceseparationrpca/
5. http://zafarrafii.com/repet.html
6. http://www.loria.fr/~aliutkus/kam/

2.1.5. Nous nous focalisons ici sur la qualité de la voix chantée extraite. Nous n'indiquerons donc que les mesures relatives aux sources de voix chantée estimées. Pour commencer, les SDRs calculés sur chacun des groupes du corpus sont reportés dans le tableau 3.4.

	G1	G2	G3	G4	G5	Moyenne
IMM	**3.49**	**4.33**	**3.16**	**2.55**	**2.83**	**3.30**
RPCA	-0.92	-1.69	-3.65	-2.18	-1.22	-1.90
KAM	2.17	2.03	0.07	0.11	1.57	1.24
REPETsim	3.19	2.44	1.12	1.78	2.38	2.21
Sélection oracle invariante	4.37	4.42	3.31	2.85	3.33	3.69
Fusion oracle invariante	**5.07**	**5.18**	**4.03**	**3.53**	**4.22**	**4.44**
Sélection oracle variant en temps	6.33	6.61	5.36	4.78	5.62	5.78
Fusion oracle variant en temps	**6.88**	**7.07**	**5.89**	**5.28**	**6.08**	**6.28**
Sélection oracle variant en fréquence	5.08	5.19	4.18	3.72	4.34	4.53
Fusion oracle variant en fréquence	**5.37**	**5.61**	**4.61**	**4.14**	**4.72**	**4.92**

TABLEAU 3.4 – Performance individuelle (SDR) des séparateurs envisagés pour l'extraction de voix chantée et performance de sélection oracle et de fusion oracle, calculées sur la source de voix chantée estimée et moyennées sur les extraits de chaque groupe puis sur l'ensemble (dernière colonne).

Nous remarquons dans un premier temps que les résultats individuels entre les cinq groupes sont similaires. En effet, quelque soit le groupe, le meilleur séparateur en terme de SDR est le modèle IMM, suivi par le modèle REPETsim, le modèle KAM puis le modèle RPCA. En moyenne, 1.1 dB de SDR sépare les trois premiers modèles. En revanche, la RPCA donne des résultats bien inférieurs aux autres modèles puisque le SDR moyen n'atteint que -1.9 dB. En observant les autres mesures comme illustrées sur la figure 3.12 qui les représente moyennées sur l'ensemble du corpus (les valeurs numériques ont également été reportées dans le tableau 3.5), on constate que le SIR pour la RPCA est tout aussi faible que le SDR alors que le SAR est plutôt élevé comparativement aux autres séparateurs. Ceci tend à montrer que la RPCA échoue totalement à séparer la voix chantée et que par conséquent la source de voix estimée contient aussi en grande partie la source d'accompagnement musical.

Nous noterons par ailleurs qu'en plus d'être le meilleur séparateur en terme de SDR, le séparateur IMM donne le meilleur SIR, suivi par le modèle KAM. Enfin, REPETsim, qui donne une bonne séparation globale, a l'avantage de ne créer que peu d'artefacts alors que les séparateurs KAM et IMM ont des SARs comparables mais plus faibles. Ainsi, on constate que les quatre séparateurs considérés pour ce corpus ont des performances très hétérogènes, ce qui peut nous laisser espérer de bonnes performances de fusion oracle.

3.4.4 Performance oracle de fusion

Nous proposons à présent d'évaluer le potentiel de nos méthodes de fusion sur le corpus ccMixter. Comme pour le corpus CHiME, nous comparons ici les fusions invariante, variant en temps et variant en fréquence. Toutefois, nous avons fixé cette fois-ci les paramètres de fusion. Ainsi, pour la fusion variant en temps, les signaux ont été découpés en trames à l'aide d'une fenêtre sinusoïdale de 2048 échantillons avec recouvrement de 50 %. Pour la fusion variant en fréquence, nous avons découpé les signaux en 513 bandes à l'aide d'une TFCT de 1024 échantillons.

Les SDRs pour chaque groupe sont reportés dans le tableau 3.4 et les autres mesures, moyennées sur l'ensemble du corpus, sont reportées dans le tableau 3.5. La figure 3.13 illustre ces mesures pour chacune des méthodes de fusion. Comme pour le rehaussement de la parole, nous comparons chacun des cas de fusion à son équivalent en sélection.

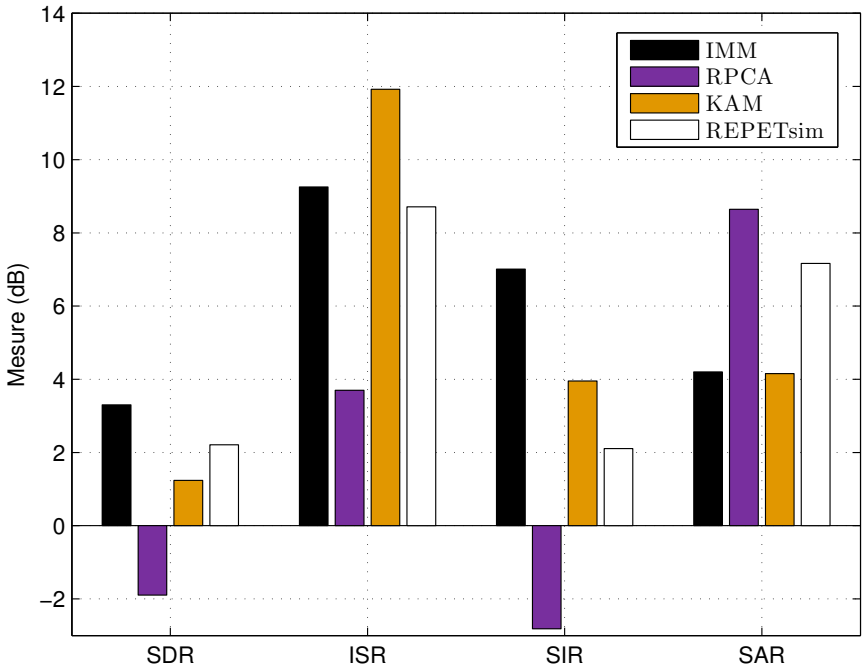


FIGURE 3.12 – Performance (SDR, ISR, SIR, SAR) des séparateurs sur le corpus ccMixter.

	SDR	ISR	SIR	SAR
IMM	**3.30**	9.26	**7.01**	4.20
RPCA	-1.90	3.70	-2.82	**8.65**
KAM	1.24	**11.92**	3.95	4.15
REPETsim	2.21	8.71	2.10	7.16
Sélection oracle invariante	3.69	**10.30**	**6.90**	5.63
Fusion oracle invariante	**4.44**	10.11	6.26	**7.27**
Sélection oracle variant en temps	5.78	**13.15**	**10.33**	6.93
Fusion oracle variant en temps	**6.28**	12.72	9.86	**8.08**
Sélection oracle variant en fréquence	4.53	**12.41**	**8.26**	6.48
Fusion oracle variant en fréquence	**4.92**	12.33	7.99	**7.34**

TABLEAU 3.5 – Performance individuelle (SDR, ISR, SAR, SIR) des séparateurs envisagés pour l'extraction de voix chantée et performance de sélection oracle et de fusion oracle, calculées sur la source de voix chantée estimée et moyennées sur l'ensemble.

Nous constatons, comme ce fut le cas pour le rehaussement de la parole, que la fusion oracle variant en temps est celle qui donne les meilleurs résultats globaux, permettant un gain de 0.5 dB de SDR par rapport à la sélection oracle variant en temps, et des gains de 1.84 dB et 1.36 dB par rapport aux fusions oracles invariante et variant en fréquence. De la même manière, parmis les trois cas de fusion étudiés, la fusion variant en temps est celle qui présente les meilleurs ISR, SIR et SAR. Comparativement au meilleur séparateur (IMM), les fusions invariantes, variant en temps et variant en fréquence peuvent donc faire gagner jusqu'à 1.1 dB, 3 dB et 1.6 dB respectivement.

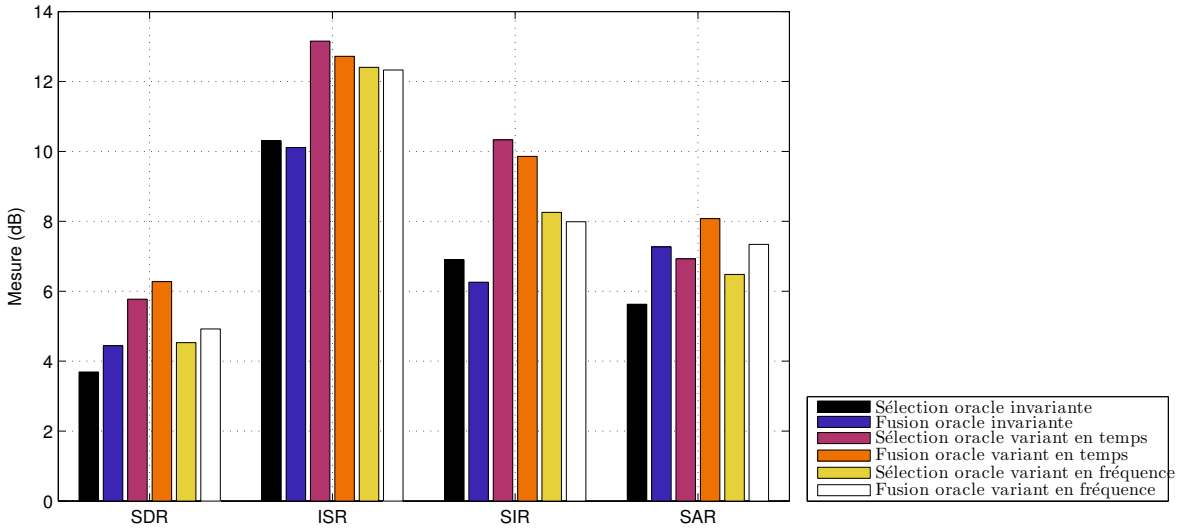


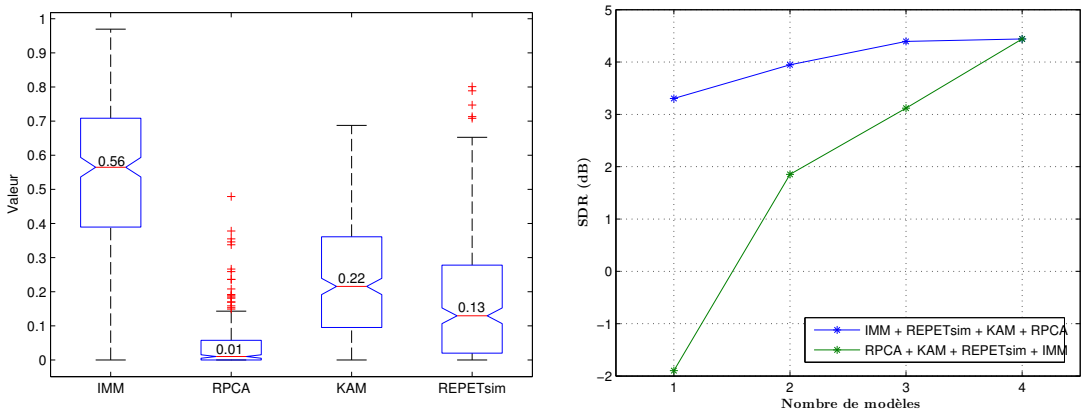
FIGURE 3.13 – Performance (SDR, ISR, SIR, SAR) des méthodes de fusion et de sélection oracles sur le corpus ccMixter

Aux vues de la faible performance du séparateur RPCA, il est judicieux de se poser la question de son utilité dans la fusion. La figure 3.14a représente la distribution des valeurs des coefficients relatifs à chacun des séparateurs pour le cas de la fusion oracle invariante et sur l'ensemble du corpus. Les distributions sont représentées sous forme de boîtes à moustache. Pour chaque modèle, ces diagrammes représentent la médiane des valeurs de coefficients de fusion oracle invariante associés à ce modèle (trait horizontal surmonté d'un chiffre indiquant la valeur de cette médiane). Cette médiane se situe dans une *boîte* délimitée par deux traits horizontaux, l'un en-dessous de la médiane (premier quartile), l'autre au-dessus (troisième quartile). Entre ces deux extrémités sont donc situés 50 % des coefficients de fusion oracle invariante pour le modèle considéré. Cette boîte se prolonge de part et d'autre par un segment tracé en pointillés et terminé de chaque côté par un nouveau trait horizontal. Ces deux nouveaux traits appelés *moustaches* définissent les valeurs extrémales des coefficients de fusion oracle invariante pour le modèle considéré. Enfin, les croix rouges se situant au-delà représentent des données dites *marginales* (*outliers* en anglais) au vu de la distribution des autres valeurs.

Ainsi, on constate que le modèle IMM, qui est le meilleur séparateur individuel en moyenne, est aussi le séparateur auquel sont associés les plus grands coefficients de fusion puisque 50% de ces coefficients ont des valeurs comprises entre 0.4 et 0.7. A l'inverse, pour la RPCA dont les performances individuelles sont faibles, les coefficients de fusion oracle ont des valeurs faibles, inférieures à 0.1 pour la majeure partie. On remarque toutefois que certaines valeurs marginales, représentées par les croix rouges, prennent quelques valeurs plus grandes, jusqu'à 0.5 pour le coefficient le plus élevé. Comme nous pouvions nous y attendre, la RPCA n'a qu'une faible influence pour la fusion oracle invariante.

Ceci est confirmé par la figure 3.14b représentant l'évolution du SDR de fusion oracle invariante

en fonction du nombre de séparateurs fusionnés. Pour cette expérience, nous avons dans un premier temps ordonné les quatre séparateurs en fonction de leurs performances individuelles puis nous avons calculé consécutivement les coefficients oracle pour un nombre de séparateurs variant de 1 à 4, en conservant l'ordre des séparateurs. Sur la courbe bleue, les séparateurs ont été ajoutés dans l'ordre croissant des performances individuelles moyennes, à savoir : IMM, REPETsim, KAM puis RPCA. A l'inverse, sur la courbe verte, les séparateurs ont été ajoutés dans l'ordre décroissant de leurs performances individuelles moyennes, à savoir : RPCA, KAM, REPETsim puis IMM. Ceci nous permet donc de confirmer la faible influence du modèle RPCA sur les résultats de fusion oracle invariante. En effet, ne pas utiliser le modèle RPCA n'engendre qu'une perte de 0.05 dB de SDR (différence entre le 3ème et le 4ème point de la courbe bleue). En comparaison, sur cette même courbe bleue, les modèles REPETsim et KAM permettent de gagner respectivement 0.7 dB et 0.5 dB de SDR. Toutefois, nous noterons que ce résultat observé sur le cas de fusion oracle invariante n'indique pas forcément qu'en pratique, le modèle RPCA sera inutile pour opérer la fusion. Pour la suite de notre étude, sur des cas non-oracles, nous conserverons donc le modèle RPCA.



(a) Distribution des coefficients de fusion oracle invariante

(b) Influence du nombre de séparateurs sur le SDR de fusion oracle invariante

FIGURE 3.14 – Distribution des coefficients de fusion oracle invariante et influence du nombre de séparateurs sur les résultats de fusion oracle invariante.

3.5 Conclusion

Dans ce chapitre, nous avons introduit notre cadre général de fusion dédié à la séparation de sources. La formulation de notre règle de fusion ne fait que peu d'hypothèses sur les séparateurs à fusionner, ce qui lui permet d'être très générale. De cette règle générale, nous avons dérivé plusieurs cas particuliers, à savoir les cas de fusion invariante, variant en temps et variant en fréquence. De même, nous avons pu distinguer la fusion statique, où les coefficients de fusion sont indépendants du signal à séparer, de la fusion adaptative, où les coefficients de fusion sont dépendants du signal à séparer.

Dans la suite de ce manuscrit, nous proposerons d'étudier ces cas particuliers de fusion sur des tâches pratiques de séparation. Dans ce chapitre, nous avons donc introduit ces tâches ainsi que les corpus et séparateurs envisagés pour chacune. Cette introduction nous a également permis d'évaluer le potentiel de nos règles de fusion, comparativement aux performances individuelles des séparateurs considérés. Que ce soit dans le cas d'une fusion homogène, où les séparateurs sont des modèles identiques dont les hyperparamètres sont différents, ou dans le cas d'une fusion hétérogène, où les séparateurs sont des modèles différents, les performances de fusion oracle se

sont toujours révélées meilleures que les performances obtenues par sélection oracle. En particulier, la fusion oracle variant en temps semble la plus prometteuse, bien que nous verrons que cette dernière est plus difficile à implémenter en pratique, comparativement aux fusions invariante et variant en fréquence.

Chapitre 4

Fusion statique : approches préliminaires

Sommaire

Dans le chapitre 3 précédent, nous avons proposé un cadre général pour la fusion en séparation de sources. De la règle de fusion générale (3.2), nous avons dérivé trois cas particuliers dont nous avons évalué le potentiel sur nos deux corpus. Toutefois, pour le moment, nous n'avons pas encore envisagé d'application pratique de ces règles de fusion puisque les résultats oracles que nous avons commentés nécessitent toujours la connaissance des sources vraies composant chaque mélange à séparer.

Dans ce chapitre, nous proposons donc d'étudier un premier cas d'application pratique de nos règles de fusion. Afin de pouvoir utiliser notre cadre de fusion dans des conditions réelles, nous n'exploiterons plus ci-après la connaissance des sources vraies dans le processus d'estimation des coefficients de fusion. Nous n'emploierons donc plus le terme d'*oracle*. De plus, nous proposerons dans ce chapitre d'estimer les coefficients de fusion sans tenir compte du mélange à séparer. À ce titre, nous qualifierons les méthodes envisagées de *statiques*.

Ainsi, dans la partie 4.1, nous proposerons des opérateurs simples permettant de formuler des règles de fusion statique sans étape préalable d'apprentissage. À l'inverse, nous proposerons dans la partie 4.2 de recourir à une étape d'apprentissage des coefficients de fusion, avant d'utiliser ces coefficients appris dans nos règles de fusion statique.

4.1 Fusion statique par opérateurs simples

Nous proposons ici des moyens simples pour estimer les coefficients de fusion pour les cas particuliers présentés dans la partie 3.2. L'idée est de proposer des opérateurs qui sont applicables sans aucun *a priori* sur le mélange à traiter ou sur les signaux à fusionner. Comme nous allons le voir, ces opérateurs peuvent être appliqués indifféremment aux règles de fusion invariante (3.10) ou de fusion variant en fréquence (3.12).

4.1.1 Fusion statique par moyenne

La manière la plus simple et la plus immédiate pour déterminer les coefficients de fusion consiste simplement à prendre la moyenne des signaux estimés à fusionner. Cela revient dans notre règle générale de fusion temps-fréquence (3.2) à poser :

$$\forall f, n, m,\ \alpha_{m,fn} = \frac{1}{M}. \tag{4.1}$$

En reprenant notre comparaison avec les problèmes de classification exposée dans la partie 3.1.3, cela revient à accorder le même degré de confiance à chacun des M séparateurs considérés.

Il est important de noter que calculer la moyenne des signaux estimés $\tilde{\mathbf{s}}_{jm,fn}$ dans le plan temps-fréquence ou des signaux estimés dans le domaine temporel $\tilde{\mathbf{s}}_{jm}(t)$ selon la règle de fusion invariante (3.10) est strictement équivalent. De la même manière, calculer la moyenne des signaux estimés par trames $\tilde{\mathbf{s}}^n_{jm}(t)$ en injectant (4.1) dans la règle de fusion variant en temps (3.11) ou la moyenne des signaux estimés filtrés par bandes de fréquence $\tilde{\mathbf{s}}^f_{jm}(t)$ en injectant (4.1) dans la règle de fusion variant en fréquence (3.12) mènera à la même source fusionnée. Par conséquent, nous désignerons dans la suite cette règle de fusion par le terme unique de *fusion statique par moyenne*. L'expression de la source ainsi fusionnée est donnée dans le tableau 4.1.

4.1.2 Fusion statique par médiane

La moyenne est un opérateur statistique simple et linéaire qui s'inscrit parfaitement dans le cadre général de fusion que nous proposons. Une autre opérateur tout aussi simple peut être utilisé pour fusionner différentes estimées : la médiane. Contrairement à la moyenne, cet opérateur

Fusion statique	Par moyenne	Par médiane
Invariante	$\tilde{\mathbf{s}}_j(t) = \frac{1}{M}\sum_{m=1}^{M} \tilde{\mathbf{s}}_{jm}(t)$	$\tilde{\mathbf{s}}_j(t) = \text{médiane}\left(\{\tilde{\mathbf{s}}_{jm}(t)\}_{m=1..M}\right)$
Variant en fréquence		$\tilde{\mathbf{s}}^f_j(t) = \text{médiane}\left(\left\{\tilde{\mathbf{s}}^f_{jm}(t)\right\}_{m=1..M}\right)$

TABLEAU 4.1 – Règles de fusion statique par opérateurs simples.

n'est pas linéaire et son application à notre problème de fusion ne respecte pas la formulation générale introduite dans la partie 3.1. Cette non-linéarité nous donne cependant l'avantage de pouvoir espérer des estimées fusionnées par médiane différentes selon le cas de fusion, invariant ou variant en fréquence. En revanche, calculer la médiane par trame ou sur l'intégralité des signaux estimés est strictement équivalent. En conséquence, dans la suite, nous distinguerons seulement la *fusion statique par médiane invariante*, et la *fusion statique par médiane variant en fréquence*. Leurs expressions respectives sont données dans le tableau 4.1.

4.2 Fusion statique par apprentissage

Pour aller plus loin, nous proposons ci-après d'apprendre les coefficients de fusion statique sur un ensemble d'apprentissage représentatif du mélange à séparer. Nous restons dans le domaine de la fusion statique car les coefficients de fusion vont être appris de manière parfaitement indépendante du mélange à séparer. Nous supposons simplement que nous disposons d'un ensemble d'apprentissage et proposons de déterminer les coefficients de fusion par minimisation sur ce dernier d'une fonction de coût dépendant des coefficients de fusion. Précisément, nous proposerons dans la partie 4.2.1 de minimiser l'erreur quadratique moyenne de reconstruction des sources sur l'ensemble d'apprentissage alors que dans la partie 4.2.2, nous proposerons plutôt de maximiser le SDR moyen sur ce même ensemble. Dans ces deux cas, nous préciserons comment appliquer ces méthodes aux cas de fusions statiques invariante et variant en fréquence (le cas variant en temps étant de toute façon identique au cas invariant).

Dans la suite, nous supposerons que l'ensemble d'apprentissage est composé de L mélanges $\mathbf{x}^{(l)}(t)$ indexés par l et que pour chacun de ces exemples nous disposons :

— des J sources vraies $\mathbf{s}_j^{(l)}(t)$ composant le mélange,

— des estimées $\tilde{\mathbf{s}}_{jm}^{(l)}(t)$ de chacune de ces sources par chacun des M séparateurs.

4.2.1 Par minimisation de l'erreur quadratique moyenne

La fusion statique invariante par minimisation de l'erreur quadratique moyenne a été initialement proposée dans [JAUREGUIBERRY et al., 2013] pour le cas de la fusion invariante. Nous proposons ici de présenter d'abord sa formulation dans le cas invariant puis de l'étendre ensuite à la fusion statique variant en fréquence.

Cas invariant

Dans la partie 3.2.3, nous avons présenté comment déterminer, pour un mélange donné, les coefficients de fusion oracle invariante par la minimisation du problème PQ défini à l'équation (3.16). Pour rappel, résoudre ce problème revient à minimiser l'erreur quadratique moyenne (EQM)

entre la source vraie $\mathbf{s}_j(t)$ et son estimée fusionnée $\tilde{\mathbf{s}}_j(t) = \sum_{m=1}^{M} \alpha_m \tilde{\mathbf{s}}_{jm}(t)$. La connaissance des sources vraies qui composent le mélange original n'étant pas disponible en pratique, nous proposons ici de résoudre un problème PQ similaire mais visant cette fois à minimiser l'EQM sur l'ensemble d'apprentissage. Les coefficients de fusion sont ainsi appris en résolvant le problème suivant :

$$\begin{array}{ll} \underset{\{\alpha_m\}_{m=1..M}}{\operatorname{argmin}} & \sum_{l=1}^{L} \sum_t \left\| \mathbf{s}_j^{(l)}(t) - \tilde{\mathbf{s}}_j^{(l)}(t) \right\|^2 \\ \text{avec} & \begin{cases} \forall m,\, \alpha_m \geq 0 \\ \sum_{m=1}^{M} \alpha_m = 1, \end{cases} \end{array} \tag{4.2}$$

où $\tilde{\mathbf{s}}_j^{(l)}(t) = \sum_{m=1}^{M} \alpha_m \tilde{\mathbf{s}}_{jm}^{(l)}(t)$ représente la $j^{\text{ième}}$ source fusionnée pour l'exemple l. Tout comme pour la fusion oracle invariante, ce problème PQ peut être reformulé sous une forme matricielle standard similaire à (3.16) telle que :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} & \left(\sum_l c^{(l)}\right) + \boldsymbol{\alpha}^\intercal \left(\sum_l \widetilde{\mathbf{G}}^{(l)}\right) \boldsymbol{\alpha} - 2 \left(\sum_l \tilde{\mathbf{d}}^{(l)}\right)^\intercal \boldsymbol{\alpha} \\ \text{avec} & \begin{cases} \forall m,\, \alpha_m \geq 0 \\ \sum_{m=1}^{M} \alpha_m = 1. \end{cases} \end{array} \tag{4.3}$$

Cette fois, la matrice de Gram $\widetilde{\mathbf{G}} = \left(\sum_l \widetilde{\mathbf{G}}^{(l)}\right)$, le vecteur $\tilde{\mathbf{d}} = \left(\sum_l \tilde{\mathbf{d}}^{(l)}\right)$ et le scalaire $c = \left(\sum_l c^{(l)}\right)$ sont obtenus en sommant leurs valeurs calculées pour chaque exemple l selon les expressions déjà exprimées aux équations (3.17) et (3.18), soit :

$$\forall l,\, \forall m_1, m_2, \quad \tilde{g}_{m_1 m_2}^{(l)} = \sum_t \left\langle \tilde{\mathbf{s}}_{jm_1}^{(l)}(t), \tilde{\mathbf{s}}_{jm_2}^{(l)}(t) \right\rangle, \tag{4.4}$$

$$\forall m, \quad \tilde{d}_m^{(l)} = \sum_t \left\langle \mathbf{s}_j^{(l)}(t), \tilde{\mathbf{s}}_{jm}^{(l)}(t) \right\rangle \tag{4.5}$$

$$\text{et} \quad c^{(l)} = \sum_t \|\mathbf{s}_j^{(l)}(t)\|^2. \tag{4.6}$$

Les coefficients de fusion ainsi obtenus par minimisation de l'EQM sur l'ensemble d'apprentissage peuvent alors être utilisés pour n'importe quel mélange n'en faisant pas partie afin de fusionner les estimées données par les mêmes M séparateurs que ceux exploités pour l'apprentissage. Plus tard, nous ferons référence à ces coefficients de fusion comme étant les *coefficients de fusion statique invariante minimisant l'EQM*.

Rappelons enfin qu'en pratique, les fusions statiques invariante et variant en temps sont exactement identiques puisqu'appliquer un même vecteur de coefficients de fusion statique à chacune des trames des signaux estimés $\tilde{\mathbf{s}}_{jm}^{n}(t)$ selon l'équation (3.11) ou directement aux signaux estimés entiers $\tilde{\mathbf{s}}_{jm}(t)$ selon l'équation (3.10) revient strictement au même.

Cas variant en fréquence

Les coefficients de fusion statique variant en fréquence peuvent être obtenus de façon similaire au cas invariant présenté ci-dessus. En effet, nous proposons de les déterminer en minimisant l'EQM par bandes de fréquence sur l'ensemble d'apprentissage. Le problème PQ correspondant a exactement la même forme que celui relatif au cas oracle (3.23), soit :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}_f}{\operatorname{argmin}} & \left(\sum_l c_f^{(l)}\right) + \boldsymbol{\alpha}_f^\intercal \left(\sum_l \widetilde{\mathbf{G}}_f^{(l)}\right) \boldsymbol{\alpha}_f - 2 \left(\sum_l \tilde{\mathbf{d}}_f^{(l)}\right)^\intercal \boldsymbol{\alpha}_f \\ \text{avec} & \begin{cases} \forall m,\, \alpha_{m,f} \geq 0 \\ \sum_{m=1}^{M} \alpha_{m,f} = 1. \end{cases} \end{array} \tag{4.7}$$

La matrice de Gram $\widetilde{\mathbf{G}}_f = \left(\sum_l \widetilde{\mathbf{G}}_f^{(l)}\right)$, le vecteur $\tilde{\mathbf{d}}_f = \left(\sum_l \tilde{\mathbf{d}}_f^{(l)}\right)$ et la constante $c_f = \left(\sum_l c_f^{(l)}\right)$ sont ici définis par rapport aux versions filtrées des sources estimées $\tilde{\mathbf{s}}_{jm}^{f\,(l)}(t)$ et des sources vraies $\mathbf{s}_j^{f\,(l)}(t)$ tels que

$$\forall l, \quad \forall m_1, m_2, \quad \tilde{g}_{f,m_1m_2}^{(l)} = \sum_t \left\langle \tilde{\mathbf{s}}_{jm_1}^{f\,(l)}(t), \tilde{\mathbf{s}}_{jm_2}^{f\,(l)}(t) \right\rangle, \tag{4.8}$$

$$\forall m, \quad \tilde{d}_{f,m}^{(l)} = \sum_t \left\langle \mathbf{s}_j^{f\,(l)}(t), \tilde{\mathbf{s}}_{jm}^{f\,(l)}(t) \right\rangle \tag{4.9}$$

$$\text{et} \quad c_f^{(l)} = \sum_t \|\mathbf{s}_j^{f\,(l)}(t)\|^2. \tag{4.10}$$

Dans la suite, nous nommerons les coefficients de fusion ainsi appris les *coefficients de fusion statique variant en fréquence minimisant l'EQM*.

4.2.2 Par maximisation du SDR

La fusion statique invariante par maximisation du SDR a été initialement présentée dans [Jaureguiberry et al., 2015] comme une alternative à l'apprentissage par minimisation de l'EQM. En effet, contrairement au cas oracle, minimiser l'EQM ne revient pas à maximiser le SDR moyen sur l'ensemble d'apprentissage. Lorsque calculés sur un seul signal, EQM et SDR sont équivalents à un logarithme et une constante près, comme nous l'avons évoqué dans la partie 3.2.3 pour la fusion oracle. En revanche, lorsque moyennés sur un ensemble de signaux, EQM et SDR ont un comportement différent. L'apprentissage par minimisation de l'EQM va avoir tendance à compenser l'erreur la plus grande sur l'ensemble d'apprentissage. À l'inverse, l'introduction d'un terme de normalisation ainsi que la présence du logarithme dans le calcul du SDR vont plutôt avoir pour effet d'équilibrer le poids des différentes erreurs. En ce sens, nous pouvons espérer des résultats de fusion différents selon que l'on maximise le SDR moyen sur l'ensemble d'apprentissage ou que l'on minimise l'EQM sur ce même ensemble.

Nous proposons donc ci-après deux nouvelles méthodes d'apprentissage des coefficients de fusion statique invariants et variant en fréquence basées sur la maximisation du SDR moyen sur l'ensemble d'apprentissage.

Cas invariant

Maximiser le SDR moyen sur l'ensemble d'apprentissage revient à résoudre le problème suivant :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}}{\operatorname{argmax}} & \sum_l 10\log_{10} \frac{\sum_t \|\mathbf{s}_j^{(l)}(t)\|^2}{\sum_t \|\mathbf{s}_j^{(l)}(t) - \tilde{\mathbf{s}}_j^{(l)}(t)\|^2} \\ \text{avec} & \begin{cases} \forall m,\, \alpha_m \geq 0 \\ \sum_{m=1}^M \alpha_m = 1 \end{cases} \end{array}. \tag{4.11}$$

À cause de la sommation sur l devant le logarithme, il est impossible de simplifier cette expression sous forme d'un problème de minimisation de l'EQM comme nous l'avions fait auparavant pour le cas oracle. Ceci étant dit, il nous est possible en omettant le numérateur constant de transformer ce problème en un problème de minimisation. Nous obtenons :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} & \sum_l \log_{10}\left(c^{(l)} + \boldsymbol{\alpha}^\intercal \widetilde{\mathbf{G}}^{(l)} \boldsymbol{\alpha} - 2\,\tilde{\mathbf{d}}^{(l)\intercal} \boldsymbol{\alpha}\right) \\ \text{avec} & \begin{cases} \forall m,\, \alpha_m \geq 0 \\ \sum_{m=1}^M \alpha_m = 1, \end{cases} \end{array} \tag{4.12}$$

où les matrices de Gram $\widetilde{\mathbf{G}}^{(l)}$, les vecteurs $\tilde{\mathbf{d}}^{(l)}$ et les scalaires $c^{(l)}$ sont définis comme aux équations (4.4), (4.5) et (4.6).

Contrairement au problème (4.3), le problème (4.12) n'est plus un programme quadratique sous contraintes linéaires d'égalité et d'inégalité, à cause de la non-linéarité introduite par le logarithme dans la fonction de coût. Sa résolution est donc plus compliquée mais des techniques d'optimisation telles que les algorithmes *active-set* [Wright et Nocedal, 1999] permettent d'atteindre un minimum local de la fonction de coût. Dans nos expériences, nous avons utilisée la fonction *fmincon* de la boîte à outil *Optimization* du logiciel *MATLAB*. Afin d'améliorer la solution donnée, nous avons calculé et fourni le gradient de la fonction. Son expression est donnée en annexe D.1. Dans la suite, nous ferons référence aux coefficients ainsi appris par le terme de *coefficients de fusion statique invariante maximisant le SDR*.

Cas variant en fréquence

Des coefficients de fusion statique variant en fréquence peuvent être obtenus de façon similaire au cas invariant ci-dessus. Cette fois-ci, nous proposons de résoudre le problème de minimisation (4.12) mais indépendamment sur chaque bande de fréquence f, tel que :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}_f}{\operatorname{argmin}} & \sum_l \log_{10}\left(c_f^{(l)} + \boldsymbol{\alpha}_f^\intercal\, \widetilde{\mathbf{G}}_f^{(l)}\, \boldsymbol{\alpha}_f - 2\,\tilde{\mathbf{d}}_f^{(l)\intercal}\, \boldsymbol{\alpha}_f\right) \\ \text{avec} & \left\{ \begin{array}{l} \forall m,\, \alpha_{m,f} \geq 0 \\ \sum_{m=1}^{M} \alpha_{m,f} = 1, \end{array} \right. \end{array} \tag{4.13}$$

où les matrices de Gram $\widetilde{\mathbf{G}}_f^{(l)}$, les vecteurs $\tilde{\mathbf{d}}_f^{(l)}$ et les scalaires $c_f^{(l)}$ sont définis comme aux équations (4.8), (4.9) et (4.10). Ce problème non-linéaire sous contraintes linéaires d'égalité et d'inégalité peut être également résolu à l'aide d'algorithme *active-set* [Wright et Nocedal, 1999]. Nous avons là aussi utilisé la fonction *MATLAB fmincon* en fournissant l'expression du gradient de la fonction de coût comme précisée en annexe D.2. Les coefficients ainsi obtenus seront appelés *coefficients de fusion statique variant en fréquence maximisant le SDR par bande*.

Notons que les coefficients de fusion obtenus en résolvant le problème (4.13) ne sont pas les coefficients qui maximisent le SDR moyen sur l'ensemble d'apprentissage, comme c'est le cas pour la fusion invariante. En effet, pour maximiser le SDR moyen, il faudrait en pratique estimer les coefficients de fusion conjointement pour chacune des bandes de fréquence en résolvant le problème suivant :

$$\begin{array}{ll} \underset{\{\boldsymbol{\alpha}_f\}_{f=1..F}}{\operatorname{argmin}} & \sum_l \log_{10}\left(\sum_{f=1}^{F}\left(c_f^{(l)} + \boldsymbol{\alpha}_f^\intercal\, \widetilde{\mathbf{G}}_f^{(l)}\, \boldsymbol{\alpha} - 2\,\tilde{\mathbf{d}}_f^{(l)\intercal}\, \boldsymbol{\alpha}_f\right)\right) \\ \text{avec} & \forall f,\ \left\{ \begin{array}{l} \forall m, \alpha_{m,f} \geq 0 \\ \sum_{m=1}^{M} \alpha_{m,f} = 1 \end{array} \right. . \end{array} \tag{4.14}$$

On constate alors que le problème à résoudre prend une dimension potentiellement conséquente puisqu'il compte $F \times M$ inconnues. Par exemple, dans nos expériences présentées plus tard dans la partie 4.3, cela nécessiterait l'estimation d'un total de $513 \times 7 = 3591$ coefficients de fusion. Pour limiter la dimension du problème, nous préférons donc résoudre F fois le problème (4.13) indépendamment, ce qui revient à minimiser le SDR moyen des signaux filtrés sur chacune des bandes de fréquence f.

4.3 Expériences

Dans cette partie, les fusions statiques par moyenne et médiane introduites dans la partie 4.1 et les approches par apprentissage de la partie 4.2 sont mises en œuvres et évaluées sur nos deux

corpus. La partie 4.3.1 suivante est dédiée aux résultats obtenus sur le corpus CHiME pour la fusion de séparateurs homogènes (NMF). La partie 4.3.2 sera elle consacrée aux résultats sur le corpus ccMixter pour la fusion de séparateurs hétérogènes.

4.3.1 Fusion homogène : corpus CHiME

Nous présentons ci-après les résultats de fusion statique par moyenne, par médiane et par apprentissage obtenus sur les ensembles de validation et de test du corpus CHiME, pour le cas de la fusion homogène de NMFs de différents ordres. Quelle que soit la fonction de coût choisie pour l'apprentissage (EQM ou SDR), nous avons utilisé l'ensemble d'apprentissage décrit dans la partie 3.3.2. Nous avons arbitrairement fixé le nombre de bandes de fréquence à $F = 513$. Les sources vraies filtrées $\mathbf{s}_j^f(t)$ et les sources estimées filtrées $\tilde{\mathbf{s}}_{jm}^f(t)$ ont été obtenues par masquage et inversion de leur STFT, calculée avec une fenêtre de 1024 échantillons et un recouvrement de moitié. Nous considérons ici seulement les modèles NMF définis dans le tableau 3.1, soit $M = 7$ séparateurs.

Comme pour les résultats oracles présentés dans la partie 3.3.6, nous proposons de comparer nos approches par fusion à leurs équivalents par sélection. Pour cela, il suffit d'imposer une contrainte supplémentaire à la résolution des problèmes (4.3), (4.12), (4.7) et (4.13). Dans le cas invariant, cette contrainte impose qu'un seul des coefficients de fusion α_m peut être non-nul et donc égal à 1. Il en est de même pour le cas variant en fréquence sauf que cette contrainte est exprimée indépendamment sur chacune des bandes de fréquence.

Approche de référence par sélection

La sélection invariante, obtenue par minimisation de l'EQM ou par maximisation du SDR sur l'ensemble d'apprentissage selon les problèmes (4.3) et (4.12) et avec la contrainte supplémentaire d'un seul coefficient non-nul, constitue une référence particulièrement intéressante pour la suite de nos expériences. En effet, elle simule la méthode de sélection la plus couramment utilisée en séparation de sources qui consiste à baser son choix sur l'expérience. Lorsque confronté en pratique à un nouveau mélange à séparer, il est coutume de sélectionner un séparateur dont on sait, par expérience, qu'il fonctionne *bien* sur des mélanges similaires. Cette notion de *bon fonctionnement* peut être tout à fait subjective et basée par exemple sur des critères d'écoute mais peut être aussi fondée sur des critères objectifs tels que le SDR ou l'EQM, comme c'est le cas pour la sélection invariante ici envisagée.

Fusions statiques par moyenne et par médiane invariante

Nous avons regroupé sur la figure 4.1 les SDRs obtenus en moyenne sur les ensembles de validation et de test pour chacune des méthodes de fusion proposées et leur équivalent en sélection. Les valeurs numériques correspondantes sont reportées dans le tableau 4.2.

En premier lieu, nous constatons que la fusion statique par moyenne et la fusion statique invariante par médiane donnent des résultats satisfaisants puisque leurs SDRs moyens sont supérieurs à toutes les performances individuelles des séparateurs envisagés, données plus tôt dans le tableau 3.2, excepté le meilleur séparateur (*NMF-32*) qui donnait un SDR moyen de 5.03 dB en moyenne sur les ensembles de validation et de test. Ainsi, on constate qu'il peut être intéressant, lorsque l'on ne dispose d'aucune connaissance pour guider le choix d'un séparateur, d'en choisir plusieurs et de les fusionner par médiane ou moyenne. Toutefois, on notera qu'en disposant d'un ensemble d'apprentissage représentatif, comme c'est le cas ici, il est possible de déterminer le meilleur séparateur par sélection du séparateur qui maximise le SDR moyen sur l'ensemble d'apprentissage. En

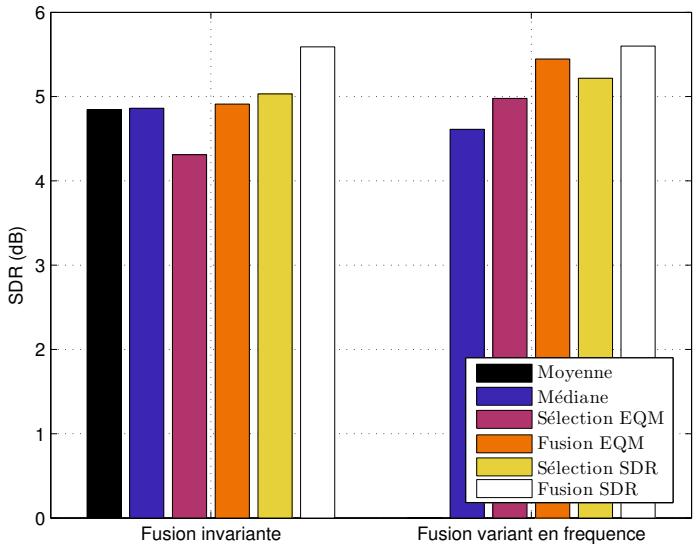


FIGURE 4.1 – Performance de fusion statique en moyenne, de fusion statique invariante et de fusion statique variant en fréquence, pour le rehaussement de la parole. Dans les barres relatives à la fusion variant en fréquence, la fusion par moyenne n'apparaît pas puisqu'elle est équivalente à la fusion invariante par moyenne.

Fusion statique			SDR	ISR	SIR	SAR
Par moyenne			4.84	7.31	7.67	13.21
Invariante	Par médiane		4.86	7.88	9.37	10.34
	EQM	sél.	4.31	7.44	9.60	9.37
		fus.	4.91	7.30	8.11	12.81
	SDR	sél.	5.03	10.80	8.20	11.68
		fus.	**5.59**	10.81	7.58	13.77
Variant en fréquence	Par médiane		4.61	7.74	9.81	10.10
	EQM	sél.	4.98	8.47	9.90	10.81
		fus.	5.45	8.06	9.09	13.23
	SDR	sél.	5.22	11.61	7.82	12.95
		fus.	**5.60**	11.73	7.96	13.82

TABLEAU 4.2 – Performance (SDR, ISR, SIR et SAR) de fusion statique pour le rehaussement de la parole, calculée sur les sources de parole et moyennée sur les ensembles de validation et de test (*sél.* et *fus.* signifient respectivement *sélection* et *fusion)*.

effet, la sélection statique invariante par maximisation du SDR moyen (cinquième ligne du tableau 4.2) mène au choix du séparateur *NMF-32* et nous donne donc la même performance.

Par ailleurs, si la performance globale de la fusion par moyenne et celle de la fusion statique invariante par médiane sont très proches, on constate que la fusion par moyenne engendre moins d'artefacts que la fusion invariante par médiane (13.21 dB de SAR contre 10.34 dB). À l'inverse, la fusion par moyenne souffre de plus d'interférences comparativement à la fusion invariante par médiane (7.67 dB de SIR contre 9.37 dB).

Fusion statique invariante par apprentissage

Nos approches par apprentissage confirment l'intérêt de la fusion par rapport à la simple sélection. En effet, quelle que soit la fonction de coût optimisée, le SDR moyen de fusion est supérieur au SDR moyen de l'approche par sélection correspondante. On constate un gain de 0.6 dB de SDR que ce soit pour la fusion statique invariante par minimisation de l'EQM ou pour la fusion statique invariante par maximisation du SDR. La fusion statique invariante par maximisation du SDR permet un gain de presque 0.7 dB par rapport à l'approche par minimisation de l'EQM. Les autres mesures (ISR, SIR et SAR) sont aussi plus élevées pour l'approche par maximisation du SDR. Toutefois, comme nous l'avions remarqué dans le cas oracle, le gain en qualité globale (SDR) apporté par la fusion se fait au détriment du SIR mais s'accompagne par un gain très net en SAR. Pour notre cas de rehaussement de la parole, cela signifie que la qualité de la source de voix fusionnée se trouve moins altérée par les méthodes de séparation utilisées mais qu'en contrepartie, cette source fusionnée contient un peu plus de la source de bruit. A l'écoute, cette perte de SIR semble principalement s'expliquer par la récupération de sons non-voisés mal modélisés par le modèle de parole et qui, lors des séparations initiales, sont représentés par le modèle de bruit. La fusion invariante ne permet alors pas de discriminer assez précisément les sons non-voisés des bruits environnementaux.

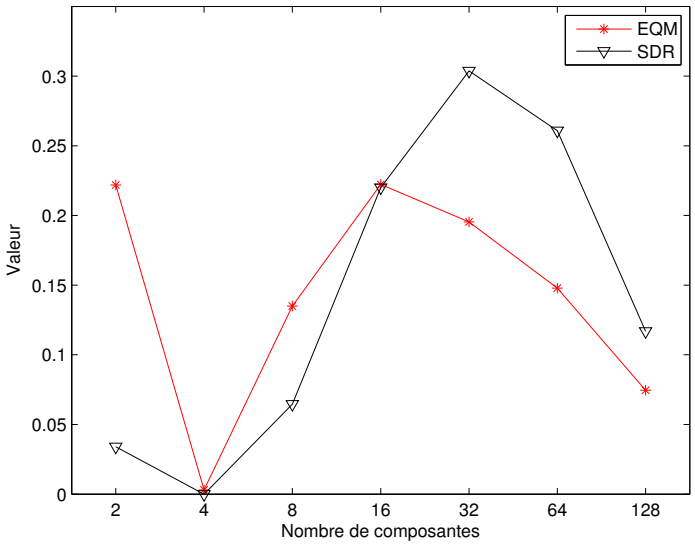


FIGURE 4.2 – Coefficients de fusion statique invariante, en fonction de la fonction de coût.

La figure 4.2 représente les coefficients de fusion statique invariante obtenus par minimisation de l'EQM en rouge et par maximisation du SDR en noir. Il est a noter que lorsque la sélection est préférée à la fusion, alors le séparateur sélectionné correspond au séparateur ayant le plus grand coefficient pour la fusion correspondante. Ainsi, le séparateur *NMF-16* est sélectionné par minimisation de l'EQM alors que c'est le séparateur *NMF-32* qui est sélectionné lorsque la fonction

de coût choisie est le SDR. On notera par ailleurs que les valeurs des cinq premiers coefficients de fusion statique invariante obtenus par maximisation du SDR (courbe noire) sont ordonnées dans le même sens que les SDRs des séparateurs correspondants (voir tableau 3.2), c'est-à-dire que le meilleur séparateur en terme de SDR a le plus grand coefficient de fusion et ainsi de suite. Toujours en comparant les valeurs des coefficients appris aux performances individuelles des séparateurs données dans le tableau 3.2, on comprend pourquoi la méthode par maximisation du SDR a un SIR sensiblement moins bon que la méthode par minimisation de l'EQM mais un meilleur SAR. En effet, les coefficients de fusion obtenus par maximisation du SDR donnent plus de poids aux séparateurs ayant des nombres de composantes élevés, ces séparateurs ayant de faibles SIR mais de bons SAR pour les raisons invoquées dans la partie 3.3.5.

Fusion statique variant en fréquence par apprentissage

Dans la figure 4.1 et le tableau 4.2 figurent aussi les résultats de fusion statique variant en fréquence. Bien que la fusion statique variant en fréquence par minimisation de l'EQM permet de gagner 0.54 dB de SDR par rapport à son équivalent invariant, il n'en est pas de même pour l'approche par maximisation du SDR. En effet, les performances obtenues par maximisation du SDR dans le cas variant en fréquence sont très proches de celles obtenues dans le cas invariant (seulement 0.01 dB de SDR en plus). Pourtant, comme le montre la figure 4.3, les coefficients de fusion variant en fréquence par maximisation du SDR, représentés sur la sous-figure en bas à droite, sont très différents de ceux du cas invariant et ne donnent pas majoritairement le poids au meilleur séparateur (*NMF-32*), ce qui aurait pu expliquer les performances proches entre cas invariant et variant en fréquence. Ceci peut sans doute s'expliquer par les raisons déjà évoquées dans la partie 4.2.2. En effet, rappelons que le problème (4.13) que nous proposons de résoudre ne revient pas à maximiser le SDR moyen sur l'ensemble d'apprentissage mais le SDR par bande de fréquence. En considérant plutôt le problème (4.14), nous pourrions espérer un véritable gain de SDR par rapport au cas invariant, au prix d'un problème de dimension bien supérieure. Il serait alors probable que la majorité des bandes de fréquence ait un coefficient maximal pour le meilleur séparateur *NMF-32*.

Plus généralement, que ce soit lorsque l'on minimise l'EQM ou lorsque l'on maximise le SDR, il est difficile de déduire, à la vue de la figure 4.3, une interprétation intuitive des valeurs de coefficients en fonction de la bande de fréquence. En effet, par exemple, dans le cas de la minimisation de l'EQM, l'apprentissage a tendance à favoriser le plus petit nombre de composantes (le séparateur *NMF-2*) alors même que c'est le séparateur qui a les moins bonnes performances individuelles. De la même manière, on constate que le meilleur séparateur (*NMF-32*) en terme de SDR individuel n'a que peu de coefficients à valeur élevée, que ce soit pour la fusion par minimisation de l'EQM ou par maximisation du SDR.

4.3.2 Fusion hétérogène : corpus ccMixter

Nous avons mené les mêmes expériences de fusion statique sur notre corpus ccMixter dédié à l'extraction de voix chantée. Pour rappel, les séparateurs envisagés ici sont ceux déjà présentés dans la partie 3.4.1 ($M = 4$). Ces séparateurs étant par nature différents, nous parlons ici de fusion hétérogène. Nous présentons ci-après les résultats de fusion statique par moyenne, par médiane et par apprentissage obtenus sur chacun des groupes du corpus, ainsi qu'en moyenne sur tout le corpus. Les ensembles d'apprentissage ont été définis en suivant le principe de validation croisée, comme indiqué dans la partie 3.4.1. Pour la fusion variant en fréquence, nous avons arbitrairement fixé le nombre de bandes de fréquence à $F = 513$ comme pour le rehaussement de la parole. Les sources vraies filtrées $\mathbf{s}_j^f(t)$ et les sources estimées filtrées $\tilde{\mathbf{s}}_{jm}^f(t)$ ont été obtenues par masquage et inversion de leur STFT, calculée avec une fenêtre de 1024 échantillons et un recouvrement de

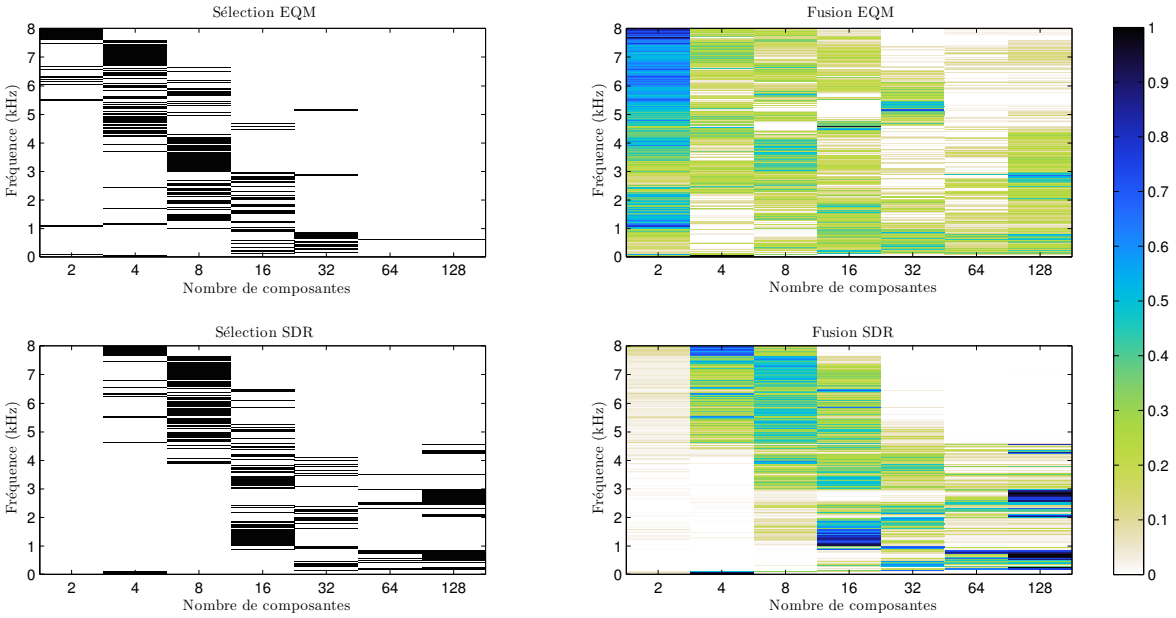


FIGURE 4.3 – Coefficients de fusion statique variant en fréquence, en fonction de la méthode de sélection ou de fusion.

moitié. Enfin, comme pour le rehaussement de la parole, nous proposons de comparer toutes nos approches par fusion à leurs équivalents par sélection.

Fusion et sélection

Comme le montre la figure 4.4, nous obtenons sur ce corpus des résultats assez comparables à ceux obtenus sur le corpus de rehaussement de la parole. En premier lieu, nous pouvons constater que la fusion par apprentissage est encore toujours plus performante que l'approche par sélection correspondante. En effet, selon les valeurs reportées dans le tableau 4.3, nous constatons que quelle que soit la fonction de coût choisie pour l'apprentissage des coefficients de fusion, la fusion donne toujours un meilleur SDR que la sélection. Ce fut aussi le cas sur le corpus CHiME. Ici, le gain atteint jusqu'à 1 dB pour les cas de fusions statiques invariante et variant en fréquence par minimisation de l'EQM, contre 0.6 et 0.8 dB pour les fusions statiques invariante et variant en fréquence respectivement, dans le cas de la maximisation du SDR.

Fusions statiques par moyenne et par médiane

Nous proposons ici encore de comparer fusions par moyenne et médiane à la sélection invariante, qui simule bien comment un séparateur est sélectionné classiquement. Au vu de la figure 4.4, nous pouvons constater que les fusions invariante et variant en fréquence par médiane permettent un gain moyen de 0.1 dB et 0.3 dB de SDR respectivement par rapport à la sélection invariante par maximisation du SDR (0.5 dB et 0.7 dB de SDR dans le cas de la sélection invariante par minimisation de l'EQM). Nous constatons que ce gain n'est toutefois pas systématique puisque, par exemple, pour le groupe 3, les fusions par médiane et par moyenne sont moins performantes que la sélection invariante, et ce quelle que soit la fonction de coût considérée.

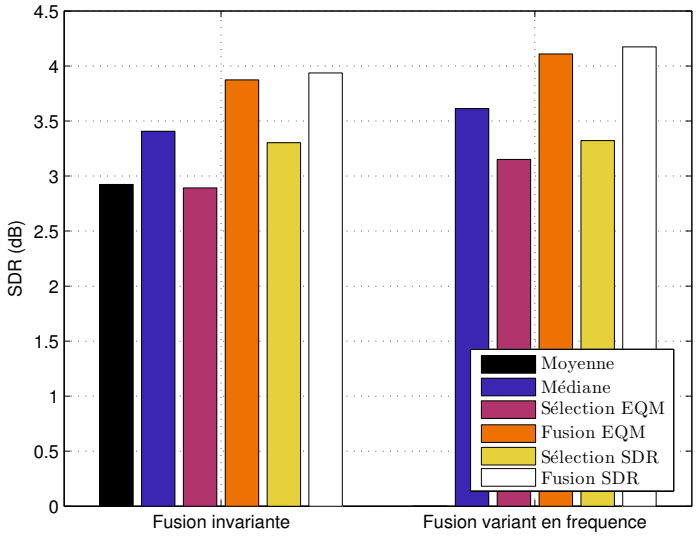


FIGURE 4.4 – Performance (SDR) de fusion statique par moyenne, de fusion statique par médiane, de sélection et de fusion statiques invariantes et de sélection et de fusion statiques variant en fréquence, pour l'extraction de voix chantée.

Fusion statique invariante par apprentissage

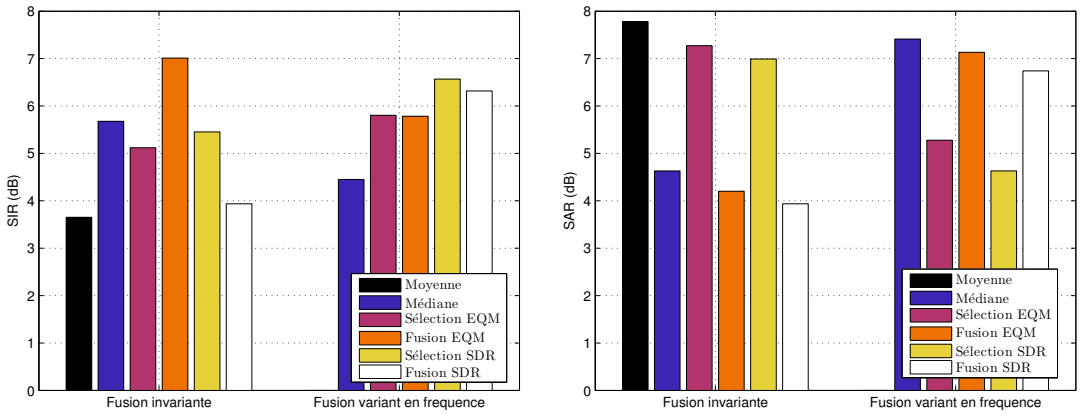
Comme nous l'avions constaté sur le corpus *CHiME*, les fusions statiques par apprentissage donnent ici encore les meilleures performances. Que ce soit dans le cas invariant ou dans le cas variant en fréquence, le choix de la fonction de coût semble avoir toutefois moins d'importance que nous avions pu le noter pour le rehaussement de la parole, puisque les fusions par maximisation du SDR ne devancent les fusions par minimisation de l'EQM que par à peine plus de 0.05 dB.

On pourra cependant remarquer que si la maximisation du SDR permet effectivement de sélectionner le meilleur séparateur individuel (le modèle IMM, d'après le tableau 3.4), la minimisation de l'EQM ne le permet que sur quatre groupes parmi les cinq. Dans ce cas, seul l'apprentissage pour le groupe 2 mène à la sélection d'un autre séparateur (le séparateur REPETsim, qui est tout de même le deuxième meilleur séparateur en terme de performance individuelle). Malgré cela, la fusion permet de pallier ce défaut de la sélection, puisque pour le groupe 2, la fusion par minimisation de l'EQM apporte un gain de 2.2 dB par rapport à la sélection correspondante, soit 0.3 dB par rapport au résultat individuel du meilleur séparateur (IMM).

Fusion statique variant en fréquence par apprentissage

La fusion statique variant en fréquence permet un gain par rapport à la fusion statique invariante. Toutefois, ce gain de 0.25 dB de SDR en moyenne ne paraît pas si important au regard de l'effort de calcul supplémentaire qu'elle requiert par rapport à la fusion invariante. Si l'on se réfère aux figures 4.5a et 4.5b, nous pouvons tout de même remarquer que la fusion statique variant en fréquence permet d'améliorer nettement le SIR et le SAR par rapport à la fusion statique invariante dans le cas de l'aprentissage par maximisation du SDR, de près de 2 dB pour chaque. Cette figure montre également que la fusion statique invariante par minimisation de l'EQM présente un SIR de 3 dB supérieur à la fusion statique invariante par maximisation du SDR, SIR qui est, de surcroît, le plus élevé de toutes les méthodes de fusion et de sélection présentées ici.

Enfin, comme nous l'avons fait sur le corpus *CHiME*, nous avons également tracé sur la figure 4.6 les coefficients de fusion appris dans le cas de la maximisation du SDR, en fonction du groupe considéré et pour chaque bande de fréquence. Cette fois , il est possible de dégager un compor-



(a) SIR (b) SAR

FIGURE 4.5 – Performance (SIR et SAR) de fusion statique en moyenne, de fusion statique invariante et de fusion statique variant en fréquence, pour l'extraction de voix chantée.

tement général des coefficients de fusion appris en fonction de la fréquence. On peut en effet constater que le modèle IMM, qui est le meilleur séparateur en terme de SDR individuel, est celui qui a les plus forts coefficients de fusion, notamment pour les très hautes fréquences supérieures à 10 kHz. Le modèle KAM semble lui prendre l'ascendant dans les basses fréquences, inférieures à 2 kHz. Bien que donnant un SDR moyen assez faible (voir tableau 3.4), le modèle RPCA semble être localement préféré dans de fines bandes de fréquence autour de 2.5 kHz et 5 kHz, ainsi que dans les très hautes fréquences supérieures à 20 kHz. La contribution du modèle REPETsim, bien que ce soit le deuxième meilleur séparateur en terme de performance individuelle, semble être très faible et difficilement généralisable. Il semble seulement contribuer dans les basses fréquences inférieures à 2.5 kHz.

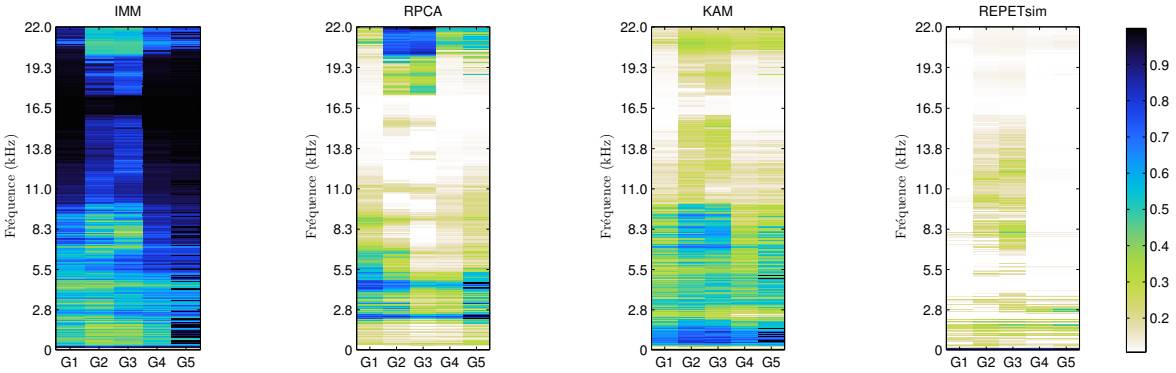


FIGURE 4.6 – Coefficients de fusion variant en fréquence appris par maximisation du SDR, en fonction du groupe d'exemples et de la bande de fréquence.

Fusion statique			G1	G2	G3	G4	G5	Moyenne
Par moyenne			3.62	3.47	1.94	2.34	3.08	2.92
Invariante	Par médiane		4.13	4.01	2.72	2.63	3.38	3.41
	EQM	sél.	3.49	2.44	3.16	2.55	2.83	2.89
		fus.	**4.44**	4.61	3.31	3.15	**3.70**	3.87
	SDR	sél.	3.49	4.33	3.16	2.55	2.83	3.30
		fus.	4.38	**4.75**	**3.55**	**3.17**	3.67	**3.94**
Variant en fréquence	Par médiane		4.28	4.25	3.00	2.83	3.54	3.61
	EQM	sél.	3.83	3.83	2.39	2.64	2.89	3.15
		fus.	**4.67**	4.83	3.58	**3.40**	**3.91**	4.11
	SDR	sél.	3.59	4.17	3.18	2.66	2.88	3.32
		fus.	4.58	**5.03**	**3.88**	3.39	3.84	**4.17**

TABLEAU 4.3 – Performance (SDR) de sélection statique et de fusion statique pour l'extraction de voix chantée, calculée sur la source de voix chantée estimée et moyennée sur les extraits de chaque groupe puis sur l'ensemble (*sél.* et *fus.* signifient respectivement *sélection* et *fusion)*.

4.4 Conclusion

Dans ce chapitre, nous avons proposé un premier ensemble de techniques simples pour opérer les fusions invariante et variant en fréquence présentées au chapitre 3. Nos expériences menées sur les deux corpus ont permis avant tout de confirmer l'intérêt de la fusion par rapport à la simple sélection.

Les fusions statiques par moyenne et par médiane se sont par ailleurs révélées être des moyens faciles pour fusionner des estimées et éviter le choix d'un seul séparateur lorsque nous ne disposons d'aucune expérience ou d'aucun *a priori* sur le mélange à séparer. Toutefois, nos expériences n'ont pas permis de définir une méthode à préférer parmi la fusion statique par moyenne, la fusion statique invariante par médiane et la fusion statique variant en fréquence par médiane. En pratique, il conviendra donc d'opérer ces trois fusions qui sont peu coûteuses en temps de calcul et de sélectionner la plus satisfaisante pour l'application considérée, par rapport à un critère subjectif par exemple.

Si en pratique nous disposons d'un ensemble d'apprentissage, il nous est alors possible de procéder à l'apprentissage préalable de coefficients de fusion statique sur cet ensemble. Nous avons proposé pour cela deux possibilités pour le choix de la fonction de coût. Là encore, nos expériences n'ont pas permis de déterminer quelle est la fonction de coût à favoriser. Ceci étant dit, nos expériences suggèrent qu'il est plus prometteur de choisir de maximiser le SDR moyen sur l'ensemble d'apprentissage, au risque de dégrader modérément le SIR. Quelle que soit la fonction de coût choisie, il reste en tout cas toujours préférable d'opérer une fusion plutôt qu'une simple sélection, les gains ainsi obtenus variant de 0.4 à 1 dB de SDR.

Nos résultats suggèrent aussi que l'apprentissage des coefficients de fusion statique dans le cas variant en fréquence selon les problèmes (4.7) et (4.13) n'est pas satisfaisant, au regard du faible gain qu'il apporte par rapport à l'approche invariante, plus simple. Comme nous l'avons déjà indiqué, les performances de cet apprentissage pourraient certainement être améliorées en minimisant la fonction de coût conjointement sur toutes les bandes de fréquence. Une autre piste plus générale pour améliorer les performances de l'apprentissage consisterait à envisager d'autres fonctions de coût à optimiser, comme l'une des autres mesures objectives (SIR, SAR ou ISR), une combinaison de ces mesures ou tout autre mesure liée à l'application de séparation souhaitée.

En conclusion, les méthodes de fusion statique que nous avons ici proposées sont donc suffisamment générales et flexibles pour pouvoir être appliquées à n'importe quel problème de séparation

de sources et à n'importe quel type de séparateur. Toutefois, et nous allons l'envisager dans les chapitres suivants, nous pouvons espérer des gains de qualité substantiels en proposant de surcroît l'adaptation de la fusion au mélange à séparer.

Chapitre 5

Fusion adaptative : approche bayésienne pour la fusion de NMFs

Sommaire

Dans le chapitre 4, nous avons proposé des moyens simples à mettre en œuvre pour estimer des coefficients de fusion dits *statiques*. Les méthodes envisagées ont l'avantage de pouvoir être appliquées à n'importe quel type de séparateur. De plus, les expériences réalisées sur nos deux corpus ont montré que la fusion apportait systématiquement un gain de qualité de séparation par rapport aux approches correspondantes par sélection. Toutefois, les résultats oracles que nous avons présentés dans le chapitre 3 nous laissent espérer des gains substantiels si nous parvenons à adapter les coefficients de fusion au signal à séparer.

Dans ce chapitre, nous introduisons donc une première approche de fusion adaptative, basée sur la modélisation statistique du signal à séparer, afin de donner une interprétation probabiliste au cadre de fusion que nous avons décrit dans le chapitre 3. Une approche probabiliste de notre cadre de fusion ne peut s'envisager qu'au prix d'une perte de généralité de ce dernier. En effet, seuls des séparateurs disposant eux-mêmes d'une formulation probabiliste peuvent être considérés dans un tel cadre.

Ainsi, dans ce chapitre, nous limiterons notre étude à la fusion de séparateurs NMFs, pour lesquels plusieurs formulations probabilistes sont disponibles dans la littérature. Nous introduirons dans un premier temps dans la partie 5.1 la formulation adoptée dans notre étude ainsi que la technique d'inférence mise en œuvre, en présentant les travaux de [Adiloğlu et Vincent, 2012]. Dans la partie 5.2 suivante, nous présenterons le principe de *moyennage bayésien de modèles* qui nous permettra de donner une interprétation probabiliste à nos coefficients de fusion. Nous dériverons ensuite de ce principe un nouveau modèle génératif pour la NMF permettant de modéliser une source par plusieurs NMFs d'ordres différents estimées conjointement. Ce modèle nommé *NMF à ordre multiple* sera présenté dans la partie 5.3. Au travers de résultats préliminaires synthétiques présentés dans la partie 5.4, nous montrerons que, bien que l'interprétation ainsi donnée des coefficients de fusion soit séduisante, elle ne suffit guère en pratique à opérer une véritable fusion. Pour y remédier, nous proposerons dans cette même partie une modification mineure du principe de moyennage bayésien que nous validerons sur données synthétiques. Enfin, nous consacrerons la dernière partie 5.5 de ce chapitre à l'analyse de résultats obtenus sur le corpus *CHiME* déjà exploité auparavant.

5.1 VB-NMF

Comme nous l'avons déjà introduit dans la partie 2.1.3, la formulation déterministe originelle de la NMF [Lee et Seung, 2001] peut être interprétée à l'aide d'un modèle statistique qui dépend de la fonction de coût choisie pour l'estimation des paramètres. Pour certaines de ces fonctions de coût, l'optimisation des paramètres de façon déterministe est alors équivalente à l'estimation par maximum de vraisemblance (MV) des paramètres dans le cadre probabiliste correspondant [Cichocki et al., 2009; Févotte et al., 2009; Virtanen et al., 2008a].

Par comparaison aux problèmes de classification, nous proposions dans la partie 3.1.3 de voir le coefficient de fusion invariant α_m comme reflétant le degré de confiance porté au séparateur $\mathcal{M}_m$. En termes probabilistes, nous pourrions donc dans un premier temps proposer d'identifier ce coefficient de fusion à la vraisemblance du modèle $p(\mathbf{X}|\mathbf{S}, \mathcal{M}_m)$. Toutefois, il est bien connu que la vraisemblance d'un modèle ne suffit pas à formuler un critère de sélection de modèles car, en particulier, la vraisemblance est toujours plus élevée pour un modèle d'ordre élevé comparativement à un modèle d'ordre plus faible. Pour autant, un modèle d'ordre trop grand cause souvent un surapprentissage. Comme nous l'avons déjà indiqué dans la partie 2.3.1, un bon critère de sélection résulte alors souvent d'un compromis entre complexité du modèle et adéquation aux données.

Ce compromis se trouve naturellement exprimé lorsqu'un problème est résolu dans un cadre bayésien complet [Bishop, 2006]. Pour ce faire, il convient de modéliser tous les paramètres du modèle à l'aide de variables aléatoires. La procédure consiste alors à inférer la distribution *a*

posteriori de l'ensemble de ces paramètres et à intégrer la vraisemblance du modèle par rapport à ces paramètres. La grandeur ainsi obtenue, nommée *vraisemblance marginale* (en anglais, *marginal likelihood* ou *evidence*), peut alors être utilisée comme critère de sélection, le meilleur modèle étant celui dont la vraisemblance marginale est maximale.

Pour aller dans ce sens, cette partie sera donc consacrée à la présentation des travaux de [Adiloğlu et Vincent, 2012] introduisant la NMF variationnelle bayésienne. Dans un premier temps, nous présenterons dans la partie 5.1.1 la formulation bayésienne de la NMF. La partie 5.1.2 introduira ensuite la méthode approchée choisie pour inférer ses paramètres, à savoir l'approche variationnelle bayésienne. Nous présenterons alors dans la partie 5.1.3 les règles de mise à jour des paramètres du modèle NMF ainsi obtenues. Puis, nous expliquerons dans la partie 5.1.4 comment estimer les sources séparées à partir de l'inférence des paramètres de NMF avant de donner dans la partie 5.1.5 l'expression de l'énergie libre qui nous sera plus tard utile pour formuler notre règle de fusion.

5.1.1 Formulation NMF bayésienne

Comme nous l'avons justifié auparavant, nous ne nous intéressons ici qu'à la NMF obtenue par minimisation de la divergence d'Itakura-Saito. De plus, en vue de l'application au rehaussement de la parole, nous ne détaillerons ici que le cas monocanal. La généralisation de notre travail au cas multicanal est toutefois possible, suivant [Adiloğlu et Vincent, 2012].

Pour rappel, dans le cadre de la NMF probabiliste, la transformée de Fourier à court terme (TFCT) du mélange à séparer $\mathbf{x}(t)$ est modélisée selon l'équation de mélange (2.2). Dans le cas monocanal, l'équation de mélange dans le plan temps fréquence se trouve être exprimée comme

$$\forall f,n,\;\; x_{fn} \;=\; \sum_{j=1}^{J} s_{j,fn} + \epsilon_{fn} \;=\; \mathbf{A}\mathbf{s}_{fn} + \epsilon_{fn} \tag{5.1}$$

où la matrice de mélange $\mathbf{A} = [1, ..., 1]$ est un vecteur de uns de longueur J, invariant en temps et en fréquence. Le vecteur $\mathbf{s}_{fn} = [s_{1,fn}, ..., s_{j,fn}, ..., s_{J,fn}]^\intercal$ regroupe lui les J sources monocanales qui composent le mélange. Chacune de ces sources est modélisée par une distribution normale complexe centrée telle que

$$\forall j,f,n,\;\; s_{j,fn} \sim \mathcal{N}(0, v_{j,fn}) \tag{5.2}$$

dont la variance $v_{j,fn}$, homogène à une densité spectrale de puissance, est modélisée par une NMF de sorte que

$$v_{j,fn} = \sum_{k=1}^{K_j} w_{j,fk} h_{j,kn}. \tag{5.3}$$

Rappelons que cette variance $v_{j,fn}$ peut aussi être décomposée en un nombre de facteurs plus grand comme proposé dans [Ozerov et al., 2012]. Dans la suite, nous ne garderons que deux facteurs par souci de concision et afin de respecter la proposition originelle de la NMF. Le bruit de capteur ϵ_{fn} est supposé distribué selon une loi normale centrée de variance constante σ^2 telle que

$$\epsilon_{fn} \sim \mathcal{N}(0, \sigma^2). \tag{5.4}$$

En notant $\mathbf{X} = \{x_{fn}\}_{f=1..F}^{n=1..N}$ et $\mathbf{S} = \{\mathbf{s}_{fn}\}_{f=1..F}^{n=1..N}$, la log-vraisemblance s'exprime comme

$$\log p(\mathbf{X}|\mathbf{S}) = \sum_{n=1}^{N}\sum_{f=1}^{F} \log \mathcal{N}\left(x_{fn}|\mathbf{A}\mathbf{s}_{fn}, \sigma^2\right). \tag{5.5}$$

La maximisation de cette log-vraisemblance par rapport aux paramètres du modèle mène alors aux règles de mise à jour déjà exposées dans la partie 2.1.3.

Pour aller plus loin, et notamment vers notre cadre d'inférence bayésienne, il nous faut également donner aux paramètres de la NMF $w_{j,fk}$ et $h_{j,kn}$ des distributions *a priori*. Plusieurs possibilités ont été envisagées dans la littérature. Il est proposé notamment dans [Hoffman et al., 2010] de donner aux paramètres de NMF une distribution *a priori* de type gamma de sorte que :

$$\forall j,f,k, \quad w_{j,fk} \sim \text{Gamma}(a,a) \quad \text{et} \quad \forall j,k,n, \quad h_{j,kn} \sim \text{Gamma}(b,b) \tag{5.6}$$

où a et b sont des hyperparamètres réels strictement positifs à choisir. La distribution gamma est définie par

$$\forall y,k,\theta \in \mathbb{R}^{+*}, \quad \text{Gamma}(y;k,\theta) = y^{k-1}\frac{e^{-y/\theta}}{\Gamma(k)\theta^k} \tag{5.7}$$

où Γ représente la fonction gamma [Artin, 1931]. Dans [Adiloğlu et Vincent, 2012], il est proposé de remplacer l'*a priori* gamma par un *a priori* non-informatif de Jeffreys $\mathcal{J}(y) \propto 1/y$ de sorte que

$$\forall j,f,k, \quad w_{j,fk} \sim \mathcal{J} \quad \text{et} \quad \forall j,k,n, \quad h_{j,kn} \sim \mathcal{J}. \tag{5.8}$$

On remarquera que l'*a priori* de Jeffreys est un cas limite de la distribution gamma, lorsque $k \to 0$ et $\theta \to +\infty$. Notons également que les mêmes distributions *a priori* pourront être utilisées pour les autres facteurs dans le cas d'une NMF avec plus de deux facteurs.

Notre modèle génératif bayésien ainsi défini peut être représenté par le modèle graphique illustré à la figure 5.1. Dans la suite, nous noterons $\mathbf{Z}$ l'ensemble des variables cachées du modèle :

$$\mathbf{Z} = \{\mathbf{S}, \mathbf{W}, \mathbf{H}\} \tag{5.9}$$

avec $\mathbf{W} = \{\mathbf{W}_j\}_{j=1..J}$ et $\mathbf{H} = \{\mathbf{H}_j\}_{j=1..J}$. La probabilité conjointe s'écrit alors

$$p(\mathbf{X},\mathbf{Z}) \;=\; p(\mathbf{X}|\mathbf{S})\,p(\mathbf{S}|\mathbf{W},\mathbf{H})\,p(\mathbf{W})\,p(\mathbf{H}). \tag{5.10}$$

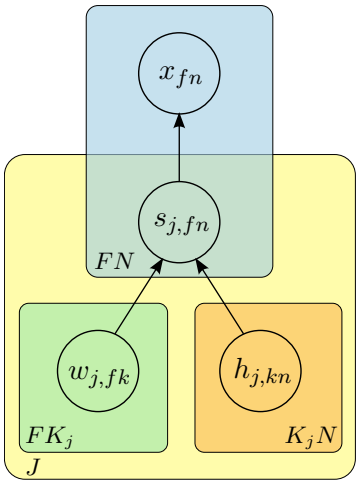


Figure 5.1 – Modèle graphique de la NMF bayésienne à ordre unique.

5.1.2 Inférence variationnelle bayésienne

Une fois le modèle défini, l'objectif de l'inférence bayésienne est de calculer la probabilité *a posteriori* de ses paramètres $p(\mathbf{Z}|\mathbf{X})$ ainsi que la vraisemblance marginale $p(\mathbf{X})$. Comme souvent, ces grandeurs n'ont pas de solution analytique et il faut donc recourir à une méthode de calcul approchée. Ici, nous proposons de mettre en œuvre l'inférence variationnelle bayésienne qui a déjà fait ses preuves sur ce modèle génératif [Adiloğlu et Vincent, 2012; Hoffman et al., 2010].

Principe de l'inférence variationnelle bayésienne

Le principe de l'inférence variationnelle bayésienne s'appuie sur la décomposition de l'expression de la log-vraisemblance marginale $\log p(\mathbf{X})$ en deux termes $\mathcal{L}[q]$ et $\mathsf{KL}[q\|p]$, selon

$$\log p(\mathbf{X}) = \mathcal{L}[q] + \mathsf{KL}[q\|p] \tag{5.11}$$

avec

$$\mathcal{L}[q] = \int q(\mathbf{Z}) \log \frac{p(\mathbf{X},\mathbf{Z})}{q(\mathbf{Z})} d\mathbf{Z} \tag{5.12}$$

et

$$\mathsf{KL}[q\|p] = -\int q(\mathbf{Z}) \log \frac{p(\mathbf{Z}|\mathbf{X})}{q(\mathbf{Z})} d\mathbf{Z}. \tag{5.13}$$

La distribution $q(\mathbf{Z})$ ici introduite est appelée *distribution variationnelle jointe* des paramètres et est utilisée comme approximation de la distribution *a posteriori* des paramètres $p(\mathbf{Z}|\mathbf{X})$. On constate en effet qu'en posant $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X})$ dans l'expression (5.11) de la log-vraisemblance marginale, le terme $\mathsf{KL}[q\|p]$, qui n'est autre que la distance de Kullback-Leibler entre la distribution variationnelle $q(\mathbf{Z})$ et la distribution *a posteriori* $p(\mathbf{Z}|\mathbf{X})$ des paramètres, est nul. Rappelant que la divergence de Kullback-Leibler (KL) satisfait toujours $\mathsf{KL}[q\|p] \geq 0$, il vient alors que le terme $\mathcal{L}[q]$, couramment appelé *énergie libre*, forme une borne inférieure de la log-vraisemblance marginale, telle que $\mathcal{L}[q] \leq \log p(\mathbf{X})$. Il en résulte naturellement que maximiser l'énergie libre $\mathcal{L}[q]$ par rapport à $q(\mathbf{Z})$ revient à minimiser la divergence $\mathsf{KL}[q\|p]$ entre $q(\mathbf{Z})$ et $p(\mathbf{Z}|\mathbf{X})$ et que la borne $\mathcal{L}[q]$ ainsi obtenue peut être considérée comme une approximation de la log-vraisemblance marginale $\log p(\mathbf{X})$.

Puisque la vraie distribution *a posteriori* des paramètres $p(\mathbf{Z}|\mathbf{X})$ ne peut être calculée, la solution $q^*(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X})$ n'est pas accessible. L'inférence variationnelle bayésienne préconise alors de considérer une famille restreinte de distributions $q(\mathbf{Z})$ et de chercher le membre de cette famille minimisant la divergence KL. En particulier, dans [Bishop, 2006], il est proposé de partitionner l'ensemble des paramètres $\mathbf{Z}$ en plusieurs ensembles disjoints $\mathbf{Z}_\delta$ de sorte que la distribution variationnelle totale soit factorisée suivant

$$q(\mathbf{Z}) = \prod_\delta q_\delta(\mathbf{Z}_\delta). \tag{5.14}$$

Ce type d'approximation est bien connu en physique sous le nom d'*approximation de champ moyen*. Il est alors démontré, notamment dans [Bishop, 2006], que la solution optimale q^*_δ qui maximise l'énergie libre par rapport à la distribution q_δ, tout en gardant les autres distributions $\{q_{\delta'}\}_{\delta' \neq \delta}$ constantes, est donnée par

$$\log q^*_\delta(\mathbf{Z}_\delta) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X},\mathbf{Z})\right] + \text{cte}, \tag{5.15}$$

où la constante de normalisation est telle que q^*_δ soit une distribution de probabilité valide. Le terme $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X},\mathbf{Z})\right]$ représente l'espérance de la log-probabilité conjointe $\log p(\mathbf{X},\mathbf{Z})$ par rapport

aux distributions variationnelles $\{q_{\delta'}(\mathbf{Z}_{\delta'})\}_{\delta' \neq \delta}$, c'est-à-dire tous les facteurs de la distribution variationnelle totale $q(\mathbf{Z})$ définis par l'équation (5.14) sauf $q_\delta(\mathbf{Z}_\delta)$. L'inférence des distributions est donc menée de manière itérative, successivement pour chaque facteur δ, à la manière de l'algorithme Espérance-Maximisation (EM). Pour commencer, il est donné une initialisation appropriée à chaque distribution $q_\delta(\mathbf{Z}_\delta)$. Ensuite, pour chaque facteur δ, l'étape E consiste à calculer l'espérance de la log-probabilité jointe par rapport aux valeurs courantes des autres distributions $q_{\delta'}$ pour $\delta' \neq \delta$ et l'étape M met à jour les paramètres de la distribution q_δ suivant l'équation (5.15).

Approximation de l'énergie libre

En pratique, il peut arriver que pour l'un des facteurs δ le terme $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X},\mathbf{Z})\right]$ n'ait pas de solution analytique ou ne s'identifie pas à une distribution paramétrique connue. Dans ce cas, il est proposé dans [Adiloğlu et Vincent, 2012] d'approximer le terme $p(\mathbf{X},\mathbf{Z})$ de l'équation (5.15) par une borne inférieure paramétrique que nous noterons $f(\mathbf{X},\mathbf{Z},\mathbf{\Omega})$ de sorte que

$$f(\mathbf{X},\mathbf{Z},\mathbf{\Omega}) \leq p(\mathbf{X},\mathbf{Z}), \tag{5.16}$$

où $\mathbf{\Omega}$ représente un ensemble de variables auxiliaires définissant la borne inférieure. En remplaçant $p(\mathbf{X},\mathbf{Z})$ par $f(\mathbf{X},\mathbf{Z},\mathbf{\Omega})$ dans la définition de l'énergie libre (5.12), nous pouvons définir une borne inférieure de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega})$ telle que

$$\mathcal{B}[q](\mathbf{\Omega}) = \int q(\mathbf{Z}) \log \frac{f(\mathbf{X},\mathbf{Z},\mathbf{\Omega})}{q(\mathbf{Z})} d\mathbf{Z} \leq \mathcal{L}[q]. \tag{5.17}$$

Le schéma d'inférence peut alors être modifié comme suit. Plutôt que de maximiser l'énergie libre par rapport à chaque distribution q_δ, il est proposé de d'abord maximiser la borne inférieure $\mathcal{B}[q](\mathbf{\Omega})$ par rapport à $\mathbf{\Omega}$ afin d'approcher au mieux la vraie énergie libre, puis de maximiser cette borne inférieure maximale $\mathcal{B}[q](\mathbf{\Omega}^*)$ par rapport à chaque distribution variationnelle q_δ. La distribution optimale q_δ^* est alors donnée par

$$\log q_\delta^*(\mathbf{Z}_\delta) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log f(\mathbf{X},\mathbf{Z},\mathbf{\Omega}^*)\right] + \text{cte}. \tag{5.18}$$

Dans la suite, nous utiliserons cette stratégie afin de calculer la mise à jour des paramètres NMF et des sources, car nous verrons que le terme $\mathbb{E}[\log p(\mathbf{S}|\mathbf{W},\mathbf{H})]$ n'a pas de forme analytique.

Application à la NMF bayésienne

Suivant [Adiloğlu et Vincent, 2012], la distribution variationnelle $q(\mathbf{Z})$ est factorisée en fonction de chacun des paramètres du modèle définis en (5.9) selon

$$q(\mathbf{Z}) = q(\mathbf{S})q(\mathbf{W})q(\mathbf{H}) = \prod_{fn} q(\mathbf{s}_{fn}) \prod_{j,fk} q(w_{j,fk}) \prod_{j,kn} q(h_{j,kn}). \tag{5.19}$$

Afin de simplifier l'écriture, nous avons volontairement omis de préciser que chaque probabilité q est unique. Pour être précis, nous aurions dû noter par exemple $q_{\mathbf{s}_{fn}}(\mathbf{s}_{fn})$ au lieu de seulement $q(\mathbf{s}_{fn})$. De même, la notation $q(\mathbf{S})$ pour désigner l'ensemble des distributions variationnelles $q_{\mathbf{s}_{fn}}(\mathbf{s}_{fn})$ sera préférée dans la suite afin d'alléger l'écriture.

La distribution variationnelle $q(\mathbf{Z})$ maximisant l'énergie libre est obtenue en appliquant la solution (5.15) à chacun des facteurs définis en (5.19). Ainsi, nous obtenons les trois équations suivantes :

$$\log q^*(\mathbf{s}_{fn}) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log p(\mathbf{X},\mathbf{Z})\right] + \text{cte}, \tag{5.20}$$

$$\log q^*(w_{j,fk}) = \mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[\log p(\mathbf{X},\mathbf{Z})\right] + \text{cte}, \tag{5.21}$$

$$\log q^*(h_{j,kn}) = \mathbb{E}_{\mathbf{Z}\setminus h_{j,kn}}\left[\log p(\mathbf{X},\mathbf{Z})\right] + \text{cte}. \tag{5.22}$$

Chacune de ces équations met en jeu l'espérance de la log-probabilité jointe déjà définie à l'équation (5.10). Cette espérance peut donc se développer sous la forme :

$$\begin{aligned}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X},\mathbf{Z})\right] = \; & \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X}|\mathbf{S})\right] + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{S}|\mathbf{W},\mathbf{H})\right] \\ & + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{W})\right] + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{H})\right].\end{aligned} \tag{5.23}$$

Parmi ces termes, nous verrons que le calcul de $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X}|\mathbf{S})\right]$, $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{W})\right]$ et $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{H})\right]$ ne pose pas de problème particulier. En revanche, nous allons détailler le calcul du terme $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{S}|\mathbf{W},\mathbf{H})\right]$ afin de préciser les approximations faites pour rendre la solution analytique.

En développant son expression et en y injectant l'expression de la distribution *a priori* des sources définie en (5.2), le terme $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{S}|\mathbf{W},\mathbf{H})\right]$ s'écrit :

$$\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{S}|\mathbf{W},\mathbf{H})\right] = \sum_{j,fn}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(s_{j,fn}|v_{j,fn})\right] \tag{5.24}$$

$$= \sum_{j,fn}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log \mathcal{N}(s_{j,fn}|0,v_{j,fn})\right] \tag{5.25}$$

$$\begin{aligned} = \; & -JFN\log\pi - \sum_{j,fn}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log v_{j,fn}\right] \\ & - \sum_{j,fn}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[|s_{j,fn}|^2\right]\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[v_{j,fn}^{-1}\right].\end{aligned} \tag{5.26}$$

Parmi les termes du développement, $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log v_{j,fn}\right]$ et $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}[v_{j,fn}^{-1}]$ n'ont pas de forme analytique du fait que $v_{j,fn}$ s'exprime comme une somme de composantes NMF selon (5.3). Nous devons donc leur trouver une borne inférieure afin de pouvoir mener à bien l'inférence.

D'après [Adilŏglu et Vincent, 2012; Hoffman et al., 2010], comme la fonction $v_{j,fn} \to -\log(v_{j,fn})$ est convexe, elle peut être bornée par son développement en série de Taylor au premier ordre autour d'un point positif $\omega_{j,fn}$ quelconque

$$-\log v_{j,fn} \geq -\log\omega_{j,fn} + 1 - \frac{1}{\omega_{j,fn}}v_{j,fn}. \tag{5.27}$$

Pour le second terme, la fonction $v_{j,fn} \to -v_{j,fn}^{-1}$ étant concave, et rappelant que $v_{j,fn} = \sum_{k=1}^{K_j} w_{j,fk}h_{j,kn}$, alors $\forall \phi_{j,fn,k} \geq 0$ tel que $\sum_{k=1}^{K_j}\phi_{j,fn,k} = 1$, nous avons

$$-\left(\sum_{k=1}^{K_j} w_{j,fk}h_{j,kn}\right)^{-1} \geq -\sum_{k=1}^{K_j}\phi_{j,fn,k}^2 w_{j,fk}^{-1}h_{j,kn}^{-1}. \tag{5.28}$$

En utilisant les inégalités (5.27) et (5.28), nous pouvons alors formuler une borne inférieure de $\log p(\mathbf{S}|\mathbf{W},\mathbf{H})$ telle que :

$$\log p(\mathbf{S}|\mathbf{W},\mathbf{H}) \geq \log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega},\boldsymbol{\phi}) \tag{5.29}$$

avec

$$\begin{aligned}\log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega},\boldsymbol{\phi}) = \; & -JFN\log\pi - \textstyle\sum_{j,fn}\left(-\log\omega_{j,fn} + 1 - \frac{1}{\omega_{j,fn}}v_{j,fn}\right) \\ & - \textstyle\sum_{j,fn}|s_{j,fn}|^2\sum_{k=1}^{K_j}\phi_{j,fn,k}^2 w_{j,fk}^{-1}h_{j,kn}^{-1}.\end{aligned} \tag{5.30}$$

où $\boldsymbol{\omega} = \{\omega_{j,fn}\}$ et $\boldsymbol{\phi} = \{\phi_{j,fn,k}\}$ désignent l'ensemble des paramètres auxiliaires définissant la borne inférieure.

Selon le principe exposé plus tôt dans cette partie, cette approximation nous permet alors de formuler une borne inférieure paramétrique de la probabilité conjointe $p(\mathbf{X}, \mathbf{Z})$ selon (5.16) et de l'énergie libre $\mathcal{L}[q]$ selon (5.17) avec

$$f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}) \;=\; p(\mathbf{X}|\mathbf{S})\, g(\mathbf{S}|\mathbf{W}, \mathbf{H}, \boldsymbol{\omega}, \boldsymbol{\phi})\, p(\mathbf{W})\, p(\mathbf{H}). \tag{5.31}$$

L'inférence sera donc menée en deux temps, en maximisant d'abord la borne inférieure de l'énergie libre par rapport aux variables auxiliaires $\mathbf{\Omega} = \{\boldsymbol{\omega}, \boldsymbol{\phi}\}$ puis par rapport aux distributions variationnelles. En pratique, les distributions variationnelles seront donc obtenues en résolvant les trois équations suivantes :

$$\log q^*(\mathbf{s}_{fn}) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}^*)\right] + \text{cte}, \tag{5.32}$$

$$\log q^*(w_{j,fk}) = \mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[\log f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}^*)\right] + \text{cte}, \tag{5.33}$$

$$\log q^*(h_{j,kn}) = \mathbb{E}_{\mathbf{Z}\setminus h_{j,kn}}\left[\log f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}^*)\right] + \text{cte}, \tag{5.34}$$

où $\mathbf{\Omega}^*$ représente l'ensemble des paramètres auxiliaires maximisant la borne inférieure de l'énergie libre (5.17).

5.1.3 Mises à jour

Comme nous l'avons expliqué dans la partie 5.1.2, à cause de l'approximation de la probabilité $p(\mathbf{S}|\mathbf{W}, \mathbf{H})$ rendue nécessaire pour atteindre une solution analytique, l'inférence variationnelle doit être désormais menée en deux temps. Dans un premier temps, il convient de maximiser par rapport aux paramètres auxiliaires $\mathbf{\Omega} = \{\boldsymbol{\omega}, \boldsymbol{\phi}\}$ la borne inférieure de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega})$ définie à l'équation (5.17) avec $f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega})$ défini selon (5.31). Dans un second temps, les distributions variationnelles optimales sont calculées selon (5.18).

Maximisation par rapport aux variables auxiliaires

Les paramètres $\omega^*_{j,fn}$ et $\phi^*_{j,fn,k}$ qui maximisent la borne inférieure de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega})$ sont obtenus de la manière suivante. Pour $\omega_{j,fn}$, il suffit de calculer la dérivée partielle de la borne inférieure et trouver en quel point $\omega^*_{j,fn}$ elle s'annule. La résolution, dont le détail est donné en annexe E.1, mène à la solution

$$\omega^*_{j,fn} = \mathbb{E}_{\mathbf{Z}}\left[v_{j,fn}\right] = \sum_{k=1}^{K_j} \mathbb{E}_{\mathbf{Z}}\left[w_{j,fk}\right] \mathbb{E}_{\mathbf{Z}}\left[h_{j,kn}\right]. \tag{5.35}$$

Pour plus de clarté, rappelons ici que la notation $\mathbb{E}_{\mathbf{Z}}$ fait référence à l'espérance calculée relativement à la distribution variationnelle $q(\mathbf{Z})$ définie à l'équation (5.19). De ce fait, nous avons $\mathbb{E}_{\mathbf{Z}}\left[w_{j,fk}\right] = \mathbb{E}_{w_{j,fk}}\left[w_{j,fk}\right]$, où l'espérance n'est en réalité calculée que relativement à la distribution $q(w_{j,fk})$. De la même manière, nous avons également $\mathbb{E}_{\mathbf{Z}}\left[h_{j,kn}\right] = \mathbb{E}_{h_{j,kn}}\left[h_{j,kn}\right]$.

Pour $\phi_{j,fn,k}$, il convient d'utiliser les multiplicateurs de Lagrange afin de tenir compte des contraintes. La résolution du système d'équations obtenu, dont le détail peut être trouvé en annexe E.1, mène alors à la solution

$$\phi^*_{j,fn,k} = \frac{1}{C_{j,fn}} \mathbb{E}_{\mathbf{Z}}\left[w^{-1}_{j,fk}\right]^{-1} \mathbb{E}_{\mathbf{Z}}\left[h^{-1}_{j,kn}\right]^{-1} \tag{5.36}$$

où $C_{j,fn}$ désigne la constante de normalisation définie par

$$C_{j,fn} = \sum_{k=1}^{K_j} \mathbb{E}_{\mathbf{Z}}\left[w^{-1}_{j,fk}\right]^{-1} \mathbb{E}_{\mathbf{Z}}\left[h^{-1}_{j,kn}\right]^{-1}. \tag{5.37}$$

On notera que ce terme exprime le résultat de la NMF par le biais de l'espérance des inverses des paramètres de la NMF. Il est par ailleurs homogène à une densité spectrale de puissance, tout comme le sont $v_{j,fn}$ et $\omega^*_{j,fn}$.

Calcul des distributions variationnelles optimales

Afin de déterminer les distributions variationnelles, nous devons à présent maximiser la borne inférieure de l'énergie libre $\mathcal{B}[q](\boldsymbol{\omega}^*, \boldsymbol{\phi}^*)$ en fonction de chaque distribution variationnelle q. La solution est alors donnée par l'équation (5.18) avec

$$\begin{aligned}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log f(\mathbf{X},\mathbf{Z},\boldsymbol{\Omega}^*)\right] = \; & \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X}|\mathbf{S})\right] + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega}^*,\boldsymbol{\phi}^*)\right] \\ & + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{W})\right] + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{H})\right].\end{aligned} \tag{5.38}$$

Distribution variationnelle des sources

En développant (5.32) et en injectant les expressions de $\omega^*_{j,fn}$ et $\phi^*_{j,fn,k}$ données aux équations (5.35) et (5.36), nous obtenons la solution suivante :

$$\log q^*(\mathbf{s}_{fn}) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log p(\mathbf{X}|\mathbf{S})\right] + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega}^*,\boldsymbol{\phi}^*)\right] + \text{cte} \tag{5.39}$$

$$= \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log \mathcal{N}(x_{fn}|\mathbf{A}\mathbf{s}_{fn},\sigma^2)\right] - \sum_j \frac{\mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[|s_{j,fn}|^2\right]}{C_{j,fn}} + \text{cte} \tag{5.40}$$

$$= -\frac{1}{\sigma^2}|x_{fn} - \mathbf{A}\mathbf{s}_{fn}|^2 - \sum_j \frac{|s_{j,fn}|^2}{C_{j,fn}} + \text{cte}. \tag{5.41}$$

En posant $\mathbf{C}_{fn} = \text{diag}\,(C_{j,fn})_{j=1..J}$ la matrice diagonale composée des J constantes de normalisation $C_{j,fn}$ définies à l'équation (5.37) et en développant la dernière ligne, nous obtenons :

$$\log q^*(\mathbf{s}_{fn}) = \mathbf{s}_{fn}^H \left(\mathbf{C}_{fn}^{-1} + \frac{1}{\sigma^2}\mathbf{A}^\intercal\mathbf{A}\right)\mathbf{s}_{fn} + \frac{2}{\sigma^2}\Re(x_{fn}\mathbf{A}\mathbf{s}_{fn}) + \text{cte}. \tag{5.42}$$

La distribution ainsi définie peut alors être identifiée à une distribution normale complexe multivariée telle que

$$q^*(\mathbf{s}_{fn}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{s}_{fn}}, \boldsymbol{\Sigma}_{\mathbf{s}_{fn}}) \tag{5.43}$$

avec

$$\boldsymbol{\mu}_{\mathbf{s}_{fn}} = \boldsymbol{\Sigma}_{\mathbf{s}_{fn}}\mathbf{A}\frac{1}{\sigma^2}x_{fn} \quad \text{et} \quad \boldsymbol{\Sigma}_{\mathbf{s}_{fn}} = \left(\mathbf{C}_{fn}^{-1} + \frac{1}{\sigma^2}\mathbf{A}^\intercal\mathbf{A}\right)^{-1}. \tag{5.44}$$

Distributions variationnelles des paramètres de NMF

En développant (5.33) et en regroupant les termes constants, la distribution variationnelle relative au paramètre de NMF $w_{j,fk}$ est donnée par

$$\log q^*(w_{j,fk}) = \mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[\log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega}^*,\boldsymbol{\phi}^*)\right] + \mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[\log p(w_{j,fk})\right] + \text{cte}. \tag{5.45}$$

La solution dépend nécessairement de l'*a priori* choisi pour les paramètres de NMF (Jeffreys ou gamma). Nous proposons ici d'utiliser l'*a priori* de Jeffreys, de sorte que $p(w_{j,fk}) = \frac{1}{w_{j,fk}}$ afin de simplifier le calcul. Les expressions des distributions variationnelles des paramètres NMF pour l'*a*

priori gamma seront données plus tard. Pour l'*a priori* de Jeffreys, nous obtenons :

$$\begin{aligned}
\log q^*(w_{j,fk}) &= \sum_n \left(-\log \omega^*_{j,fn} + 1 - \frac{1}{\omega^*_{j,fn}} \sum_{k=1}^{K_j} w_{j,fk}\, \mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[h_{j,kn}\right] \right) \\
&\quad - \sum_n \left(\mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[|s_{j,fn}|^2\right] \sum_{k=1}^{K_j} \phi^{*2}_{j,fn,k} w^{-1}_{j,fk} \mathbb{E}_{\mathbf{Z}\setminus w_{j,fk}}\left[h^{-1}_{j,kn}\right] \right) \qquad (5.46) \\
&\quad - \log w_{j,fk} + \text{cte} \\
&= -w_{j,fk} \sum_n \left(\omega^{*-1}_{j,fk}\, \mathbb{E}_{h_{j,kn}}\left[h_{j,kn}\right] \right) \\
&\quad - w^{-1}_{j,fk} \sum_n \left(\mathbb{E}_{s_{j,fn}}\left[|s_{j,fn}|^2\right]\, \phi^{*2}_{j,fn,k}\, \mathbb{E}_{h_{j,kn}}\left[h^{-1}_{j,kn}\right] \right) \qquad (5.47) \\
&\quad - \log w_{j,fk} + \text{cte}.
\end{aligned}$$

Nous pouvons constater que la solution présente un terme en $w_{j,fk}$, un terme en $w^{-1}_{j,fk}$ et un terme $\log w_{j,fk}$. La distribution $q^*(w_{j,fk})$ peut donc être identifiée à une distribution Gaussienne Inverse Généralisée (GIG) [JØRGENSEN, 1982] dont la densité de probabilité est définie par la fonction

$$\text{GIG}(y|\gamma,\rho,\tau) = \frac{\exp\left((\gamma-1)\log y - \rho y - \tau/y\right)\rho^{\gamma/2}}{2\tau^{\gamma/2}\mathcal{K}_\gamma(2\sqrt{\rho\tau})} \qquad (5.48)$$

où $y \geq 0$, $\rho \geq 0$ et $\tau \geq 0$. $\mathcal{K}_\gamma$ représente la fonction de Bessel de seconde espèce de paramètre γ. Les hyperparamètres ρ, τ et γ peuvent donc être identifiés aux termes suivant respectivement $w_{j,fk}$, $w^{-1}_{j,fk}$ et $\log w_{j,fk}$. Par ailleurs, la mise à jour de toutes les distributions $q^*(w_{j,fk})$ peut se formuler sous forme matricielle. En notant $\boldsymbol{\rho}_{\mathbf{W}_j} = \{\rho_{j,fk}\}^{f=1..F}_{k=1..K_j}$, $\boldsymbol{\tau}_{\mathbf{W}_j} = \{\tau_{j,fk}\}^{f=1..F}_{k=1..K_j}$, et $\boldsymbol{\gamma}_{\mathbf{W}_j} = \{\gamma_{j,fk}\}^{f=1..F}_{k=1..K_j}$, les mises à jour des paramètres sont données par :

$$\boldsymbol{\tau}_{\mathbf{W}_j} = \mathbb{E}_{\mathbf{Z}}\left[\frac{1}{\mathbf{W}_j}\right]^{.2} \circ \left[\left(\mathbb{E}_{\mathbf{Z}}\left[|\mathbf{S}_j|^{.2}\right] \circ \mathbf{C}_j^{.-2} \right) \left(\mathbb{E}_{\mathbf{Z}}\left[\frac{1}{\mathbf{H}_j}\right]^{.-1} \right)^{\mathsf{T}} \right], \qquad (5.49)$$

$$\boldsymbol{\rho}_{\mathbf{W}_j} = \mathbb{E}_{\mathbf{Z}}\left[\mathbf{V}_j\right]^{.-1} \mathbb{E}_{\mathbf{Z}}\left[\mathbf{H}_j\right]^{\mathsf{T}}, \qquad (5.50)$$

$$\boldsymbol{\gamma}_{\mathbf{W}_j} = 0, \qquad (5.51)$$

où l'opérateur $\circ$ représente le produit de Hadamard et $\mathbf{M}^{.x}$ représente la puissance x terme à terme de la matrice $\mathbf{M}$. De plus, nous avons noté $\mathbf{S}_j = \{s_{j,fn}\}^{f=1..F}_{n=1..N}$, $\mathbf{V}_j = \{v_{j,fn}\}^{f=1..F}_{n=1..N}$ et $\mathbf{C}_j = \{C_{j,fn}\}^{f=1..F}_{n=1..N}$.

Le même calcul peut être mené pour les activations $h_{j,kn}$ ainsi que pour l'*a priori* gamma. Les mises à jour correspondantes ont été reportées dans le tableau 5.1. On constatera que les mises à jour obtenues avec l'*a priori* de Jeffreys correspondent aux mises à jour pour l'*a priori* gamma avec $a = b = 0$.

Enfin, faisant partie de la famille des distributions exponentielles, la distribution GIG est définie de manière équivalente par ses paramètres ρ, τ et γ et par ses statistiques $\mathbb{E}\left[y\right]$, $\mathbb{E}\left[y^{-1}\right]$ et $\mathbb{E}\left[\log y\right]$. L'inférence alternera donc entre estimation des statistiques et des paramètres de la distribution à la manière de l'algorithme EM. Il est donc important de noter comment calculer les statistiques de la distribution, notamment

$$\mathbb{E}\left[y\right] = \frac{\mathcal{K}_{\gamma+1}\left(2\sqrt{\rho\tau}\right)\sqrt{\tau}}{\mathcal{K}_\gamma\left(2\sqrt{\rho\tau}\right)\sqrt{\rho}}, \qquad (5.52)$$

	A priori de Jeffreys	*A priori* gamma
$\mathbf{W}_j$	$\boldsymbol{\tau}_{\mathbf{W}_j} = \mathbb{E}_{\mathbf{Z}}\left[\frac{1}{\mathbf{W}_j}\right]^{.2} \circ \left[\left(\mathbb{E}_{\mathbf{Z}}\left[\lvert\mathbf{S}_j\rvert^{.2}\right] \circ \mathbf{C}_j^{-2}\right)\left(\mathbb{E}_{\mathbf{Z}}\left[\frac{1}{\mathbf{H}_j}\right]^{.-1}\right)^{\mathsf{T}}\right]$	
	$\boldsymbol{\rho}_{\mathbf{W}_j} = \mathbb{E}_{\mathbf{Z}}\left[\mathbf{V}_j\right]^{.-1}\mathbb{E}_{\mathbf{Z}}\left[\mathbf{H}_j\right]^{\mathsf{T}}$	$\boldsymbol{\rho}_{\mathbf{W}_j} = a + \mathbb{E}_{\mathbf{Z}}\left[\mathbf{V}_j\right]^{.-1}\mathbb{E}_{\mathbf{Z}}\left[\mathbf{H}_j\right]^{\mathsf{T}}$
	$\boldsymbol{\gamma}_{\mathbf{W}_j} = 0$	$\boldsymbol{\gamma}_{\mathbf{W}_j} = a$
$\mathbf{H}_j$	$\boldsymbol{\tau}_{\mathbf{H}_j} = \mathbb{E}_{\mathbf{Z}}\left[\frac{1}{\mathbf{H}_j}\right]^{.2} \circ \left[\left(\mathbb{E}_{\mathbf{Z}}\left[\frac{1}{\mathbf{W}_j}\right]^{.-1}\right)^{\mathsf{T}}\left(\mathbb{E}_{\mathbf{Z}}\left[\lvert\mathbf{S}_j\rvert^{.2}\right] \circ \mathbf{C}_j^{-2}\right)\right]$	
	$\boldsymbol{\rho}_{\mathbf{H}_j} = \mathbb{E}_{\mathbf{Z}}\left[\mathbf{W}_j\right]^{\mathsf{T}}\mathbb{E}_{\mathbf{Z}}\left[\mathbf{V}_j\right]^{.-1}$	$\boldsymbol{\rho}_{\mathbf{H}_j} = b + \mathbb{E}_{\mathbf{Z}}\left[\mathbf{W}_j\right]^{\mathsf{T}}\mathbb{E}_{\mathbf{Z}}\left[\mathbf{V}_j\right]^{.-1}$
	$\boldsymbol{\gamma}_{\mathbf{H}_j} = 0$	$\boldsymbol{\gamma}_{\mathbf{H}_j} = b$

TABLEAU 5.1 – Mises à jour des distributions des paramètres de NMF en fonction de l'*a priori* choisi.

$$\mathbb{E}\left[y^{-1}\right] = \frac{\mathcal{K}_{\gamma-1}\left(2\sqrt{\rho\tau}\right)\sqrt{\rho}}{\mathcal{K}_{\gamma}\left(2\sqrt{\rho\tau}\right)\sqrt{\tau}}. \tag{5.53}$$

On remarque en effet que, dans les mises à jour présentées dans le tableau 5.1, seules les grandeurs de la forme $\mathbb{E}\left[y\right]$ et $\mathbb{E}\left[y^{-1}\right]$ interviennent. Ainsi, il ne nous sera pas nécessaire de calculer les statistiques de la forme $\mathbb{E}\left[\log y\right]$.

5.1.4 Estimation des sources séparées

Rappelant que l'objectif est ici d'estimer la TFCT des J sources composant le mélange et correspondant aux variables $s_{j,fn}$, nous proposons d'identifier l'estimée de la TFCT de la $j^{\text{ième}}$ source $\tilde{s}_{j,fn}$ à l'espérance de la probabilité *a posteriori* de la variable des sources $q(\mathbf{s}_{fn})$, soit la moyenne $\boldsymbol{\mu}_{\mathbf{s}_{fn}}$ définie à l'équation (5.44). En développant son expression, nous obtenons la formulation suivante pour l'estimée de la source j :

$$\tilde{s}_{j,fn} = \frac{C_{j,fn}}{\sum_{j'=1}^{J} C_{j',fn} + \sigma^2}\, x_{fn}. \tag{5.54}$$

Le terme $C_{j,fn}$, défini en (5.37), étant homogène à une densité spectrale de puissance, nous retrouvons ici une formulation très similaire au filtrage de Wiener dans le cas de la NMF non-bayésienne, comme exprimé précédemment à l'équation (2.24), la variance $v_{j,fn} = \sum_k w_{j,fk}h_{j,kn}$ de chaque source étant simplement remplacée par le terme $C_{j,fn}$.

5.1.5 Calcul de l'énergie libre

Comme nous l'avons évoqué en introduction, la mise en œuvre d'un modèle bayésien complet nous permet d'estimer sa vraisemblance marginale. En recourant à une méthode d'inférence approchée, nous ne pouvons en pratique estimer qu'une approximation de cette vraisemblance marginale. Pour le cas de l'inférence variationnelle bayésienne, cette approximation est donnée par l'énergie

libre définie à l'équation (5.12). Cette énergie libre peut aussi s'écrire

$$\mathcal{L}[q] = \int q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} d\mathbf{Z} \tag{5.55}$$

$$= \int q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}) d\mathbf{Z} - \int q(\mathbf{Z}) \log q(\mathbf{Z}) d\mathbf{Z} \tag{5.56}$$

$$= \mathbb{E}\left[\log p(\mathbf{X}, \mathbf{Z})\right] - \mathbb{E}\left[\log q(\mathbf{Z})\right]. \tag{5.57}$$

En injectant dans l'expression de l'énergie libre l'expression de la probabilité jointe $p(\mathbf{X}, \mathbf{Z})$ définie en (5.10) et l'expression de la distribution variationnelle $q(\mathbf{Z})$ définie en (5.14), nous obtenons :

$$\begin{aligned} \mathcal{L}[q] = \quad & \textstyle\sum_{fn} \mathbb{E}\left[\log p(x_{fn}|\mathbf{A}\mathbf{s}_{fn})\right] \\ + \quad & \textstyle\sum_{fn} \left(\sum_j \mathbb{E}\left[\log p(s_{j,fn}|v_{j,fn})\right] - \mathbb{E}\left[\log q(\mathbf{s}_{fn})\right] \right) \\ + \quad & \textstyle\sum_{j,fk} \left(\mathbb{E}\left[\log p(w_{j,fk})\right] - \mathbb{E}\left[\log q(w_{j,fk})\right] \right) \\ + \quad & \textstyle\sum_{j,kn} \left(\mathbb{E}\left[\log p(h_{j,kn})\right] - \mathbb{E}\left[\log q(h_{j,kn})\right] \right). \end{aligned} \tag{5.58}$$

5.2 Moyennage bayésien de modèles

Comme nous l'avons indiqué en introduction de ce chapitre, la mise en place d'un modèle bayésien complet nous permet le calcul de la vraisemblance marginale qui peut être utilisée comme un critère de sélection de modèle. Pour aller plus loin, nous proposons dans la partie 5.2.1 suivante de rappeler le principe du moyennage bayésien de modèles déjà évoqué dans la partie 2.3.2 puis de l'appliquer dans la partie 5.2.2 à la NMF bayésienne afin d'en déduire une interprétation probabiliste de nos coefficients de fusion.

5.2.1 Principe

Supposons ici que nous cherchons à estimer la probabilité *a posteriori* d'un paramètre $\mathbf{Z}$, ou d'un ensemble de paramètres $\mathbf{Z}$, étant donnée l'observation $\mathbf{X}$, soit la quantité $p(\mathbf{Z}|\mathbf{X})$. En pratique, pour estimer cette probabilité *a posteriori*, il nous faut disposer d'un modèle $\mathcal{M}_m$ et la valeur estimée de la probabilité *a posteriori* dépend en réalité de ce modèle $\mathcal{M}_m$. Nous la noterons donc $p(\mathbf{Z}|\mathbf{X}, \mathcal{M}_m)$.

Supposons à présent que nous disposons de M estimées différentes de la probabilité $p(\mathbf{Z}|\mathbf{X}, \mathcal{M}_m)$, obtenues avec M modèles distincts ($m \in [1, M]$). Si les M modèles sont bayésiens, nous pouvons utiliser la *vraisemblance marginale* comme critère de sélection du meilleur modèle. En notant $\mathbf{Z}_m$ l'ensemble des paramètres du modèle $\mathcal{M}_m$, $\mathbf{Z}_m$ comprenant nécessairement le paramètre d'intérêt $\mathbf{Z}$, la vraisemblance marginale pour le modèle $\mathcal{M}_m$ est obtenue en *marginalisant* la probabilité jointe $p(\mathbf{X}, \mathbf{Z}_m|\mathcal{M}_m)$ du modèle $\mathcal{M}_m$ par rapport à l'ensemble de ses paramètres $\mathbf{Z}_m$, selon :

$$p(\mathbf{X}|\mathcal{M}_m) = \int p(\mathbf{X}, \mathbf{Z}_m|\mathcal{M}_m)\, d\mathbf{Z}_m. \tag{5.59}$$

Le meilleur modèle $\mathcal{M}_{m^*}$ est réputé être celui dont la vraisemblance marginale est maximale, soit :

$$m^* = \underset{m}{\operatorname{argmax}}\ p(\mathbf{X}|\mathcal{M}_m). \tag{5.60}$$

L'estimation de la probabilité *a posteriori* que nous retenons est donc $p(\mathbf{Z}|\mathbf{X}, \mathcal{M}_{m^*})$.

À l'instar de la fusion, le moyennage bayésien de modèles propose, plutôt que de ne retenir qu'un modèle parmi les M envisagés, de moyenner les estimées des probabilités *a posteriori* $p(\mathbf{Z}|\mathbf{X}, \mathcal{M}_m)$

sur l'ensemble des modèles, chaque estimée étant pondérée par la probabilité *a posteriori* du modèle associé $p(\mathcal{M}_m|\mathbf{X})$. Ainsi, une nouvelle probabilité *a posteriori* peut être formulée selon

$$p(\mathbf{Z}|\mathbf{X}) = \sum_{m=1}^{M} p(\mathcal{M}_m|\mathbf{X})\, p(\mathbf{Z}|\mathbf{X}, \mathcal{M}_m). \tag{5.61}$$

La probabilité *a posteriori* du modèle $\mathcal{M}_m$, $p(\mathcal{M}_m|\mathbf{X})$, peut être simplement obtenue par application de la règle de Bayes :

$$p(\mathcal{M}_m|\mathbf{X}) = \frac{p(\mathbf{X}|\mathcal{M}_m)\, p(\mathcal{M}_m)}{\sum_{m'=1}^{M} p(\mathbf{X}|\mathcal{M}_{m'})\, p(\mathcal{M}_{m'})}. \tag{5.62}$$

On retrouve alors la vraisemblance marginale $p(\mathbf{X}|\mathcal{M}_m)$ déjà définie en (5.59). La quantité $p(\mathcal{M}_m)$ fait elle référence à la probabilité *a priori* du modèle $\mathcal{M}_m$. En pratique, elle sera à choisir au même titre que les probabilités *a priori* des paramètres du modèle $p(\mathbf{Z}_m)$.

5.2.2 Application à la NMF bayésienne

Le principe du moyennage bayésien de modèles peut être très simplement appliqué à la NMF bayésienne introduite dans la partie 5.1. Pour cela, nous devons dans un premier temps définir un ensemble de M modèles distincts. En référence aux expériences déjà menées dans les chapitres 3 et 4 sur le corpus de parole *CHiME*, nous proposons ici de différencier les modèles par les nombres de composantes K_{jm} choisis pour chacune des sources. Ainsi, le modèle $\mathcal{M}_m$ est entièrement défini par l'ensemble de ses nombres de composantes, soit $\mathbf{K}_m = \{K_{1m}, ..., K_{jm}, ..., K_{Jm}\}$.

Afin d'opérer le moyennage bayésien, il nous faut définir la probabilité *a posteriori* du modèle $p(\mathbf{K}_m|\mathbf{X})$. Selon la règle de Bayes (5.62), celle ci peut s'exprimer comme le produit de la vraisemblance marginale $p(\mathbf{X}|\mathbf{K}_m)$ et de la probabilité *a priori* du modèle $p(\mathbf{K}_m)$ que nous noterons π_m ci-après. Comme nous l'avons indiqué plus tôt, la vraisemblance marginale du modèle NMF bayésien n'a pas de forme analytique. Ceci étant dit, nous avons donné dans la partie 5.1.5 l'expression de l'énergie libre, qui est une borne inférieure de la log-vraisemblance marginale. Ainsi, nous pouvons remplacer dans le calcul de $p(\mathbf{K}_m|\mathbf{X})$ la vraisemblance marginale par l'exponentielle de l'énergie libre définie à l'équation (5.58), de sorte que

$$p(\mathbf{K}_m|\mathbf{X}) = \frac{1}{\delta}\pi_m e^{\mathcal{L}_m}, \tag{5.63}$$

où $\mathcal{L}_m$ désigne l'énergie libre du modèle $\mathcal{M}_m$ et $\delta = \sum_{m=1}^{M} \pi_m e^{\mathcal{L}_m}$ permet de normaliser la probabilité afin qu'elle se somme à 1.

Nous proposons d'appliquer la règle de moyennage bayésien de modèles (5.61) au moyennage des estimées de la probabilité *a posteriori* des sources $q_m(\mathbf{s}_{fn})$ définie en (5.44). Ceci nous permet de formuler une nouvelle probabilité *a posteriori* des sources selon

$$q(\mathbf{s}_{fn}) = \frac{1}{\delta} \sum_{m=1}^{M} \pi_m e^{\mathcal{L}_m} q_m(\mathbf{s}_{fn}). \tag{5.64}$$

En en prenant l'espérance, nous pouvons alors définir une nouvelle estimée de chaque source $\tilde{s}_{j,fn}$ telle que

$$\tilde{s}_{j,fn} = \frac{1}{\delta} \sum_{m=1}^{M} \pi_m e^{\mathcal{L}_m} \tilde{s}_{jm,fn} \tag{5.65}$$

où $\tilde{s}_{jm,fn}$ désigne l'estimée de la source j donnée par le modèle $\mathcal{M}_m$ selon l'équation (5.54). On remarquera que cette règle de fusion s'inscrit parfaitement dans le cadre général décrit au

chapitre 3. En effet, les poids $\pi_m e^{\mathcal{L}_m}$ peuvent être identifiés aux coefficients de fusion invariante α_m introduits à l'équation (3.10) et l'équation (5.65) peut se réécrire dans le domaine temporel :

$$\forall j,t, \quad \tilde{s}_j(t) = \sum_{m=1}^{M} \alpha_m \tilde{s}_{jm}(t) \quad \text{avec} \quad \alpha_m = \frac{1}{\delta}\pi_m e^{\mathcal{L}_m}. \tag{5.66}$$

Contrairement aux approches présentées au chapitre 4, les coefficients de fusion invariante dépendent ici du signal à séparer par le biais de l'énergie libre $\mathcal{L}_m$. Nous parlerons donc de *fusion adaptative invariante VB*, ou plus concisément, de *fusion VB invariante*.

5.2.3 Extension aux fusions variant en temps et variant en fréquence

L'expression de l'énergie libre $\mathcal{L}_m$ pour le modèle $\mathcal{M}_m$ peut être facilement décomposée en termes dépendant de n (ou respectivement, dépendant de f). De ces expressions, il est possible de déduire une interprétation probabiliste des coefficients de fusion variant en temps et variant en fréquence.

Coefficients de fusion VB variant en temps

L'énergie libre, dont l'expression a été détaillée à l'équation (5.58), peut être décomposée en N termes $\mathcal{L}_{m,n}$ dépendant de l'indice de trame n, exprimant ainsi l'énergie libre de la trame n uniquement. Cette énergie libre par trame est exprimée selon

$$\begin{aligned}
\mathcal{L}_n[q] = \; & \textstyle\sum_f \mathbb{E}\left[\log p(x_{fn}|\mathbf{A}\mathbf{s}_{fn})\right] \\
& + \textstyle\sum_f \left(\sum_j \mathbb{E}\left[\log p(s_{j,fn}|v_{j,fn})\right] - \mathbb{E}\left[\log q(\mathbf{s}_{fn})\right] \right) \\
& + \textstyle\sum_{j,fk} \left(\mathbb{E}\left[\log p(w_{j,fk})\right] - \mathbb{E}\left[\log q(w_{j,fk})\right] \right) \\
& + \textstyle\sum_{j,k} \left(\mathbb{E}\left[\log p(h_{j,kn})\right] - \mathbb{E}\left[\log q(h_{j,kn})\right] \right).
\end{aligned} \tag{5.67}$$

Par application du moyennage bayésien aux sources estimées, de façon similaire au cas invariant, nous pouvons formuler les coefficients de fusion adaptative VB variant en temps selon

$$\forall j,t, \quad \tilde{s}^n_j(t) = \sum_{m=1}^{M} \alpha_{m,n} \tilde{s}^n_{jm}(t) \quad \text{avec} \quad \alpha_{m,n} = \frac{1}{\delta_n}\pi_m e^{\mathcal{L}_{m,n}}, \tag{5.68}$$

où cette fois, $\forall n, \delta_n = \sum_m \pi_m e^{\mathcal{L}_{m,n}}$. Notons que la probabilité *a priori* du nombre de composantes π_m est considérée indépendante de la trame.

Coefficients de fusion VB variant en fréquence

De la même manière, l'énergie libre peut être exprimée pour la bande de fréquence f selon

$$\begin{aligned}
\mathcal{L}_f[q] = \; & \textstyle\sum_n \mathbb{E}\left[\log p(x_{fn}|\mathbf{A}\mathbf{s}_{fn})\right] \\
& + \textstyle\sum_n \left(\sum_j \mathbb{E}\left[\log p(s_{j,fn}|v_{j,fn})\right] - \mathbb{E}\left[\log q(\mathbf{s}_{fn})\right] \right) \\
& + \textstyle\sum_{j,k} \left(\mathbb{E}\left[\log p(w_{j,fk})\right] - \mathbb{E}\left[\log q(w_{j,fk})\right] \right) \\
& + \textstyle\sum_{j,kn} \left(\mathbb{E}\left[\log p(h_{j,kn})\right] - \mathbb{E}\left[\log q(h_{j,kn})\right] \right).
\end{aligned} \tag{5.69}$$

Les coefficients de fusion adaptative VB variant en fréquence prennent alors la forme suivante :

$$\forall j,t, \quad \tilde{s}^f_j(t) = \sum_{m=1}^{M} \alpha_{m,f} \tilde{s}^f_{jm}(t) \quad \text{avec} \quad \alpha_{m,f} = \frac{1}{\delta_f}\pi_{m,f} e^{\mathcal{L}_{m,f}} \tag{5.70}$$

et $\forall f, \delta_f = \sum_m \pi_{m,f} e^{\mathcal{L}_{m,f}}$. Contrairement au cas variant en temps, nous noterons qu'ici, nous envisageons de donner une distribution *a priori* du nombre de composantes pour chaque bande de fréquence f. En effet, il nous semble plus naturel de formuler un *a priori* par bande de fréquence que par trame temporelle. De plus, nous verrons plus tard que ces probabilités *a priori* seront déterminées par apprentissage. Par conséquent, nous ne pourrons pas apprendre des probabilités *a priori* par trame, au même titre que nous n'avons pas pu apprendre des coefficients de fusion statique variant en temps dans le chapitre 4.

5.3 NMF à ordre multiple

Dans la partie précédente, nous avons proposé de déduire du principe de moyennage bayésien l'expression de coefficients de fusion adaptative invariante. Comme pour le cas de la fusion statique, afin d'opérer la fusion, il faut avant tout estimer indépendamment les M sources $\tilde{s}_{jm}(t)$ et donc multiplier d'autant le temps de calcul nécessaire, par rapport aux approches classiques de sélection ne requérant qu'un seul séparateur.

Comme nous allons le montrer ici, la modélisation bayésienne de la NMF nous permet de formuler différemment la règle de fusion afin que les M modèles NMF soient estimés et moyennés de façon conjointe, permettant ainsi un gain de calcul non négligeable. Il convient pour cela d'intégrer au modèle bayésien le nombre de composantes K_j comme nous le présenterons dans la partie 5.3.1 ci-dessous. Nous montrerons ensuite dans la partie 5.3.2 comment modifier le schéma d'inférence variationnelle bayésienne afin de tenir compte de cette nouvelle variable et formuler les règles de mises à jour qui seront présentées dans la partie 5.3.3.

5.3.1 Formulation

Comparativement au modèle bayésien de la NMF introduit dans la partie 5.1.1 précédente, le nombre de composantes est cette fois-ci vu comme une variable aléatoire au même titre que les paramètres de NMF et que les sources. Il convient donc en premier lieu de lui donner une distribution *a priori*. Nous supposerons dans la suite que le nombre de composantes pour la source j est distribué selon une distribution catégorielle (ou *multi-Bernoulli*), que nous noterons

$$K_j \sim \text{Cat}\left(\pi_{j1}, ..., \pi_{jm}, ..., \pi_{jM_j}\right) \tag{5.71}$$

où π_{jm} désigne la probabilité *a priori* du nombre de composantes K_{jm} pour la source j. Nous remarquerons que, pour l'instant, les probabilités *a priori* des nombres de composantes sont exprimées pour chacune des sources indépendamment (*c.-à-d.*, $\forall m \in [1, M_j], \pi_{jm} = p(K_{jm})$) et non pour toutes les sources à la fois (*c.-à-d.*, $\forall m \in [1, M], \pi_m = p(\mathbf{K}_m) = p(K_1, ..., K_j, ..., K_J)$) comme ce fut le cas pour le moyennage bayésien exposé dans la partie 5.2.2.

Ce modèle génératif bayésien complet peut être représenté sous la forme du modèle graphique de la figure 5.2. Comme nous le verrons plus tard, cette formulation va nous permettre d'exprimer chaque source comme le résultat de plusieurs NMFs d'ordres différents, à la manière du moyennage bayésien de modèles exprimé à l'équation (5.65). C'est pourquoi nous dénommerons par la suite ce modèle par le terme *NMF à ordre multiple*, par opposition à la *NMF à ordre unique* exposée dans la partie 5.1 et représentée par le modèle graphique de la figure 5.1.

Il résulte de cette modélisation que chaque source s_j peut être décrite comme le résultat d'une NMF indexée par m parmi M_j NMFs d'ordres différents. De ce fait, les paramètres de NMF doivent eux aussi être indexés par m de sorte que $\mathbf{W}_{jm} = \{w_{jm,fk}\}$ et $\mathbf{H}_{jm} = \{h_{jm,kn}\}$ désignent le dictionnaire et la matrice d'activation de la NMF d'ordre K_{jm} pour la source j. Comme pour la NMF bayésienne à ordre unique, les paramètres de NMF pourront suivre soit un *a priori* de Jeffreys

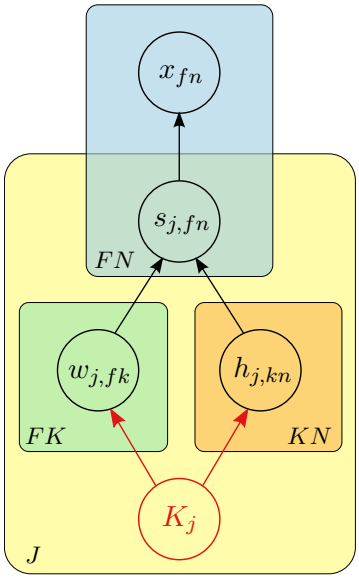


FIGURE 5.2 – Modèle graphique de la NMF bayésienne à ordre multiple.

(5.8), soit un *a priori* gamma (5.6). L'ensemble des paramètres s'écrit maintenant

$$\mathbf{Z} = \{\mathbf{S}, \mathbf{W}, \mathbf{H}, \mathbf{K}\} \tag{5.72}$$

avec $\mathbf{K} = \{K_j\}_{j=1..J}$, $\mathbf{W} = \{\mathbf{W}_{j1}, ..., \mathbf{W}_{jm}, ..., \mathbf{W}_{jM_j}\}_{j=1..J}$ et $\mathbf{H} = \{\mathbf{H}_{j1}, ..., \mathbf{H}_{jm}, ..., \mathbf{H}_{jM_j}\}_{j=1..J}$. Leur probabilité conjointe est donnée par :

$$p(\mathbf{X}, \mathbf{Z}) \;=\; p(\mathbf{X}|\mathbf{S})\, p(\mathbf{S}|\mathbf{W}, \mathbf{H})\, p(\mathbf{W}|\mathbf{K})\, p(\mathbf{H}|\mathbf{K})\, p(\mathbf{K}). \tag{5.73}$$

Les termes mis en rouge sont ceux qui diffèrent de la formulation de la NMF à ordre unique.

5.3.2 Inférence variationnelle bayésienne

Comme nous l'avons fait pour la NMF à ordre unique, nous cherchons à présent à calculer la probabilité *a posteriori* des paramètres du modèle $p(\mathbf{Z}|\mathbf{X})$ ainsi que la vraisemblance marginale $p(\mathbf{X})$. Pour ce faire, nous allons ici aussi suivre le schéma d'inférence variationnelle bayésienne. En premier lieu, il nous faut donc choisir la forme de la distribution variationnelle $q(\mathbf{Z})$. Nous proposons la factorisation suivante :

$$q(\mathbf{Z}) = q(\mathbf{S})\, q(\mathbf{W}|\mathbf{K})\, q(\mathbf{H}|\mathbf{K})\, q(\mathbf{K}) \tag{5.74}$$

$$= \prod_{fn} q(\mathbf{s}_{fn}) \prod_{jm,fk} q(w_{jm,fk}) \prod_{jm,kn} q(h_{jm,kn}) \prod_{jm} q(K_j = K_{jm}). \tag{5.75}$$

$$\tag{5.76}$$

Nous noterons que contrairement à la distribution variationnelle (5.19) pour la NMF à ordre unique, nous proposons ici de garder le conditionnement des distributions variationnelles des paramètres NMF $w_{jm,fk}$ et $h_{jm,kn}$ au nombre de composantes K_{jm} associé. En effet, les paramètres NMF sont définis relativement à l'une des valeurs possibles de K_j. Par conséquent, sans ce conditionnement, les distributions variationnelles associées ne seraient pas définies.

Cette forme particulière de factorisation de la distribution variationnelle $q(\mathbf{Z})$ induit un changement dans le schéma d'inférence traditionnel. Dans la littérature [BISHOP et WINN, 2003; SAUL et JORDAN, 1996], l'inférence porte alors le nom d'*inférence variationnelle bayésienne structurée*.

Concomitamment à nos travaux, cette forme d'inférence a également été appliquée à la NMF dans [Hoffman, 2014a] pour la sélection du nombre de composantes à l'aide d'un *a priori* de parcimonie, similairement aux méthodes présentées dans la partie 2.3.1. Pour notre application, les distributions variationnelles optimales sont alors données par les équations suivantes :

$$\log q^*(\mathbf{s}_{fn}) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log p(\mathbf{X},\mathbf{Z})\right] + \text{cte} \tag{5.77}$$

$$\log q^*(w_{jm,fk}) = \mathbb{E}_{\mathbf{Z}\setminus\{w_{j,fk},K_j\}}\left[\log p(\mathbf{X},\mathbf{Z})|K_j = K_{jm}\right] + \text{cte} \tag{5.78}$$

$$\log q^*(h_{jm,kn}) = \mathbb{E}_{\mathbf{Z}\setminus\{h_{j,kn},K_j\}}\left[\log p(\mathbf{X},\mathbf{Z})|K_j = K_{jm}\right] + \text{cte} \tag{5.79}$$

$$\begin{aligned}\log q^*(K_j) = &\mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{X},\mathbf{Z})\right] - \mathbb{E}_{\mathbf{W}_j}\left[\log q(\mathbf{W}_j|K_j)\right] \\ &- \mathbb{E}_{\mathbf{H}_j}\left[\log q(\mathbf{H}_j|K_j)\right] + \text{cte}\end{aligned} \tag{5.80}$$

On notera en particulier que l'expression de la distribution variationnelle des sources reste inchangée alors que les distributions des paramètres de NMF sont cette fois obtenues par calcul de l'espérance de la log-probabilité conjointe $\log p(\mathbf{X},\mathbf{Z})$ par rapport à tous les paramètres sauf $w_{j,fk}$ et K_j pour la distribution $q(w_{jm,fk}|K_j)$ et $h_{j,kn}$ et K_j pour la distribution $q(h_{jm,kn}|K_j)$. Enfin, la distribution variationnelle du nombre de composantes est elle exprimée comme l'espérance de la log-probabilité conjointe par rapport à tous les paramètres sauf K_j à laquelle sont ôtées les espérances des distributions variationnelles des paramètres de NMF conditionnées à K_j. Le détail des calculs peut être trouvé en annexe E.2.

De même que pour la NMF à ordre unique, les termes de la forme $\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{X},\mathbf{Z})\right]$ n'ont pas d'expression analytique. En effet, ces dernier comportent un terme de la forme

$$\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(\mathbf{S}|\mathbf{W},\mathbf{H})\right] = \sum_{j,fn}\mathbb{E}_{\mathbf{Z}\setminus\mathbf{Z}_\delta}\left[\log p(s_{j,fn}|v_{j,fn})\right] \tag{5.81}$$

$$= \sum_{j,fn}\sum_{m=1}^{M_j} q(K_{jm})\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}\left[\log p(s_{j,fn}|v_{j,fn})|K_j = K_{jm}\right] \tag{5.82}$$

$$= -JFN\log\pi$$

$$-\sum_{j,fn}\sum_{m=1}^{M_j} q(K_{jm})\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}\left[\log v_{j,fn}|K_j = K_{jm}\right] \tag{5.83}$$

$$-\sum_{j,fn}\sum_{m=1}^{M_j} q(K_{jm})\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}\left[|s_{j,fn}|^2\right]\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}[v_{j,fn}^{-1}|K_j = K_{jm}].$$

Nous retrouvons donc des termes n'ayant pas d'expression analytique semblables à ceux de l'équation (5.26) pour le cas de la NMF à ordre unique. Cette fois, les termes problématiques sont de la forme $\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}\left[\log v_{j,fn}|K_j = K_{jm}\right]$ et $\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}[v_{j,fn}^{-1}|K_j = K_{jm}]$. Rappelant que, par définition, $v_{j,fn} = \sum_{k=1}^{K_j} w_{j,fk}h_{j,kn}$ et que K_j est à présent une variable aléatoire, nous proposons de réécrire ces termes tels que

$$\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}\left[\log v_{j,fn}|K_j = K_{jm}\right] = \mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}\left[\log v_{jm,fn}\right], \tag{5.84}$$

$$\mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}[v_{j,fn}^{-1}|K_j = K_{jm}] = \mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{Z}_\delta,K_j\}}[v_{jm,fn}^{-1}], \tag{5.85}$$

avec $v_{jm,fn} = \sum_{k=1}^{K_{jm}} w_{jm,fk}h_{jm,kn}$. Ces termes peuvent donc être approximés de manière similaire au cas de la NMF à ordre unique selon les approximations (5.27) et (5.28), et ce indépendamment pour tout $m \in [1, M_j]$. Selon le principe introduit dans la partie 5.1.2, les valeurs des variables auxiliaires $\boldsymbol{\omega} = \{\omega_{jm,fn}\}$ et $\boldsymbol{\phi} = \{\phi_{jm,fn,k}\}$ devront donc être déterminées par maximisation de la borne inférieure de l'énergie libre ainsi définie avant de résoudre les équations (5.77), (5.78), (5.79) et (5.80) pour en déduire les distributions variationnelles des paramètres.

5.3.3 Mises à jour

Comme pour la NMF à ordre unique, le calcul des distributions variationnelles optimales est en pratique mené en deux temps. Dans un premier temps, il convient de maximiser par rapport aux paramètres auxiliaires $\mathbf{\Omega} = \{\boldsymbol{\omega}, \boldsymbol{\phi}\}$ la borne inférieure de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega})$ définie à l'équation (5.17) avec

$$f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega}) \;=\; p(\mathbf{X}|\mathbf{S})\, g(\mathbf{S}|\mathbf{W}, \mathbf{H}, \boldsymbol{\omega}, \boldsymbol{\phi})\, p(\mathbf{W}|\mathbf{K})\, p(\mathbf{H}|\mathbf{K})\, p(\mathbf{K}). \tag{5.86}$$

Dans un second temps, la borne inférieure de l'énergie libre sera maximisée par rapport aux distributions variationnelles.

Maximisation par rapport aux variables auxiliaires

Le calcul des paramètres $\omega^*_{jm,fn}$ et $\phi^*_{jm,fn,k}$ qui maximisent la borne inférieure de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega})$ est mené de manière parfaitement identique au cas de la NMF à ordre unique. Ainsi, les solutions sont

$$\omega^*_{jm,fn} = \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[v_{jm,fn}\right] = \sum_{k=1}^{K_{jm}} \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[w_{jm,fk}\right] \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[h_{jm,kn}\right]. \tag{5.87}$$

et

$$\phi^*_{jm,fn,k} = \frac{1}{C_{jm,fn}} \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[w^{-1}_{jm,fk}\right]^{-1} \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[h^{-1}_{jm,kn}\right]^{-1} \tag{5.88}$$

où $C_{jm,fn}$ désigne la constante de normalisation

$$C_{jm,fn} = \sum_{k=1}^{K_{jm}} \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[w^{-1}_{jm,fk}\right]^{-1} \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[h^{-1}_{jm,kn}\right]^{-1}. \tag{5.89}$$

On notera que $\omega^*_{jm,fn}$ et $C_{jm,fn}$ sont très similaires aux termes $\omega^*_{j,fn}$ et $C_{j,fn}$ précédemment définis aux équations (5.88) et (5.89) pour le modèle NMF à ordre unique. Ils expriment la densité spectrale de la source j exprimée par la NMF bayésienne d'ordre K_{jm}.

Distribution variationnelle des sources

Le développement de (5.77) nous permet d'obtenir l'expression de la distribution variationnelle des sources maximisant la borne inférieure de l'énergie libre selon :

$$\log q^*(\mathbf{s}_{fn}) \;=\; \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log p(\mathbf{X}|\mathbf{S})\right] + \mathbb{E}_{\mathbf{Z}\setminus\mathbf{s}_{fn}}\left[\log g(\mathbf{S}|\mathbf{W}, \mathbf{H}, \boldsymbol{\omega}^*, \boldsymbol{\phi}^*)\right] + \text{cte} \tag{5.90}$$

$$\;=\; -\frac{1}{\sigma^2}|x_{fn} - \mathbf{A}\mathbf{s}_{fn}|^2 - \sum_j |s_{j,fn}|^2 \sum_{m=1}^{M_j} q(K_{jm})\, C^{-1}_{jm,fn} + \text{cte}. \tag{5.91}$$

En posant

$$C_{j,fn} = \left(\sum_{m=1}^{M_j} q(K_{jm}) C^{-1}_{jm,fn}\right)^{-1} \quad \text{et} \quad \mathbf{C}_{fn} = \text{diag}\left(C_{j,fn}\right)_{j=1..J}, \tag{5.92}$$

la distribution $q^*(\mathbf{s}_{fn})$ peut être identifiée à une distribution normale complexe multivariée, tout comme pour la NMF à ordre unique :

$$q^*(\mathbf{s}_{fn}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{s}_{fn}}, \mathbf{\Sigma}_{\mathbf{s}_{fn}}) \tag{5.93}$$

avec

$$\boldsymbol{\mu}_{\mathbf{s}_{fn}} = \boldsymbol{\Sigma}_{\mathbf{s}_{fn}} \mathbf{A} \frac{1}{\sigma^2} x_{fn} \quad \text{et} \quad \boldsymbol{\Sigma}_{\mathbf{s}_{fn}} = \left(\mathbf{C}_{fn}^{-1} + \frac{1}{\sigma^2} \mathbf{A}^\intercal \mathbf{A} \right)^{-1}. \tag{5.94}$$

La seule différence avec le cas à ordre unique réside donc dans le terme $\mathbf{C}_{fn}^{-1}$ qui cette fois n'est plus le résultat d'une seule NMF mais la combinaison linéaire des M NMFs pondérées chacune par la probabilité *a posteriori* du nombre de composantes K_{jm} correspondante selon (5.89), à la manière du moyennage bayésien exposé dans la partie 5.2.

Distributions variationnelles des paramètres de NMF

Les distributions variationnelles des paramètres de NMF sont aussi obtenues de façon similaire à la NMF à ordre unique de sorte que

$$\forall j, \forall m \in [1, M_j], \forall k \in [1, K_{jm}], \quad \forall f, \quad w_{jm,fk} \sim \mathsf{GIG}\left(\gamma_{jm,fk}, \rho_{jm,fk}, \tau_{jm,fk}\right) \tag{5.95}$$

$$\text{et} \quad \forall n, \quad h_{jm,kn} \sim \mathsf{GIG}\left(\gamma_{jm,kn}, \rho_{jm,kn}, \tau_{jm,kn}\right). \tag{5.96}$$

Les paramètres des distributions GIG peuvent être comme précédemment mis à jour sous forme matricielle selon les formules données dans le tableau 5.2. On notera que seul le terme $\mathbb{E}\left[|\mathbf{S}_j|^{\cdot 2}\right]$ est parfaitement inchangé par rapport à la NMF à ordre unique. Les autres termes, et notamment $\mathbb{E}\left[\mathbf{V}_{jm}\right]$ et $\mathbf{C}_{jm}^{\cdot -2}$, sont à calculer pour le nombre de composantes K_{jm} courant, selon (5.87) et (5.89).

	A priori de Jeffreys	*A priori* gamma
$\mathbf{W}_{jm}$	$\boldsymbol{\tau}_{\mathbf{W}_{jm}} = \mathbb{E}\left[\frac{1}{\mathbf{W}_{jm}}\right]^{\cdot 2} \circ \left[\left(\mathbb{E}\left[|\mathbf{S}_j|^{\cdot 2}\right] \circ \mathbf{C}_{jm}^{\cdot -2}\right)\left(\mathbb{E}\left[\frac{1}{\mathbf{H}_{jm}}\right]^{\cdot -1}\right)^\intercal\right]$	
	$\boldsymbol{\rho}_{\mathbf{W}_m} = \mathbb{E}\left[\mathbf{V}_{jm}\right]^{\cdot -1} \mathbb{E}\left[\mathbf{H}_{jm}\right]^\intercal$	$\boldsymbol{\rho}_{\mathbf{W}_{jm}} = a + \mathbb{E}\left[\mathbf{V}_{jm}\right]^{\cdot -1} \mathbb{E}\left[\mathbf{H}_{jm}\right]^\intercal$
	$\boldsymbol{\gamma}_{\mathbf{W}_{jm}} = 0$	$\boldsymbol{\gamma}_{\mathbf{W}_{jm}} = a$
$\mathbf{H}_{jm}$	$\boldsymbol{\tau}_{\mathbf{H}_{jm}} = \mathbb{E}\left[\frac{1}{\mathbf{H}_{jm}}\right]^{\cdot 2} \circ \left[\left(\mathbb{E}\left[\frac{1}{\mathbf{W}_{jm}}\right]^{\cdot -1}\right)^\intercal \left(\mathbb{E}\left[|\mathbf{S}_j|^{\cdot 2}\right] \circ \mathbf{C}_{jm}^{\cdot -2}\right)\right]$	
	$\boldsymbol{\rho}_{\mathbf{H}_{jm}} = \mathbb{E}\left[\mathbf{W}_{jm}\right]^\intercal \mathbb{E}\left[\mathbf{V}_{jm}\right]^{\cdot -1}$	$\boldsymbol{\rho}_{\mathbf{H}_{jm}} = b + \mathbb{E}\left[\mathbf{W}_{jm}\right]^\intercal \mathbb{E}\left[\mathbf{V}_{jm}\right]^{\cdot -1}$
	$\boldsymbol{\gamma}_{\mathbf{H}_{jm}} = 0$	$\boldsymbol{\gamma}_{\mathbf{H}_{jm}} = b$

TABLEAU 5.2 – Mises à jour des paramètres NMF en fonction de l'*a priori* choisi, pour la NMF à ordre multiple.

Distributions variationnelles des nombres de composantes

Les distributions variationnelles des nombres de composantes sont obtenues en résolvant l'équation (5.80). En développant les termes de la partie droite et en tenant compte de l'approximation

de l'énergie libre définie par (5.86), nous obtenons :

$$\begin{aligned}\log q^*(K_j) = &\mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{X}|\mathbf{S})\right] + \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log f(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega}^*,\boldsymbol{\phi}^*)\right] \\ &+ \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{W}|\mathbf{K})\right] + \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{H}|\mathbf{K})\right] + \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{K})\right] \\ &- \mathbb{E}_{\mathbf{W}_j}\left[\log q(\mathbf{W}_j|K_j)\right] - \mathbb{E}_{\mathbf{H}_j}\left[\log q(\mathbf{H}_j|K_j)\right] + \text{cte.}\end{aligned} \tag{5.97}$$

En incluant dans la constante les termes indépendants du nombre de composantes K_j, la distribution variationnelle $q^*(K_j)$ peut alors s'écrire

$$\begin{aligned}\log q^*(K_j) = &\mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{W}_j|K_j)\right] - \mathbb{E}_{\mathbf{W}_j}\left[\log q(\mathbf{W}_j|K_j)\right] \\ &+ \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log p(\mathbf{H}_j|K_j)\right] - \mathbb{E}_{\mathbf{H}_j}\left[\log q(\mathbf{H}_j|K_j)\right] \\ &+ \log p(K_j) + \text{cte.}\end{aligned} \tag{5.98}$$

Cette distribution peut finalement être identifiée à une distribution catégorielle

$$\begin{aligned}\log q^*(K_j = K_{jm}) = \quad &\textstyle\sum_{fk}\Big(\mathbb{E}_{w_{jm,fk}}\big[\log p(w_{jm,fk})\big] \\ &\qquad -\mathbb{E}_{w_{jm,fk}}\big[\log q(w_{jm,fk})\big]\Big) \\ &+\textstyle\sum_{kn}\Big(\mathbb{E}_{h_{jm,kn}}\big[\log p(h_{jm,kn})\big] \\ &\qquad -\mathbb{E}_{h_{jm,kn}}\big[\log q(h_{jm,kn})\big]\Big) \\ &+\log \pi_{jm} + \text{cte.}\end{aligned} \tag{5.99}$$

La constante est obtenue en normalisant $q^*(K_j)$ de sorte que $\sum_{m=1}^{M_j} q^*(K_j = K_{jm}) = 1$.

Par ailleurs, il est possible d'interpréter cette expression de la probabilité *a posteriori* des nombres de composantes. En effet, on pourra remarquer que les termes de (5.97) sont semblables aux termes de l'énergie libre de la NMF à ordre unique exprimée aux équations (5.12) et (5.58). Les termes manquants, notamment $\mathbb{E}\left[\log q(\mathbf{s}_{fn})\right]$ et les termes $\mathbb{E}_{\mathbf{W}_{j'}}\left[\log q(\mathbf{W}_{j'}|K_{j'})\right]$ et $\mathbb{E}_{\mathbf{H}_{j'}}\left[\log q(\mathbf{H}_{j'}|K_{j'})\right]$ relatifs aux autres sources $j' \neq j$, sont communs aux M probabilités $q^*(K_j = K_{jm})$ et sont donc inclus dans la constante. Par conséquent, l'expression (5.99) peut être réécrite plus simplement selon

$$q^*(K_j = K_{jm}) = \frac{1}{\delta}\pi_{jm} e^{\mathcal{L}_{jm}} \tag{5.100}$$

avec $\delta = \sum_{m=1}^{M_j} \pi_{jm} e^{\mathcal{L}_{jm}}$ et $\mathcal{L}_{jm} = \mathbb{E}_{\mathbf{Z}\setminus K_j}\left[\log \frac{p(\mathbf{X},\mathbf{Z})}{q(\mathbf{Z})}\middle| K_j = K_{jm}\right]$. On retrouve alors une expression semblable à la probabilité *a posteriori* d'un modèle NMF à ordre unique comme exprimée à l'équation (5.63).

5.3.4 Estimation des sources séparées

Comme pour la NMF bayésienne à ordre unique, l'estimée de la $j^{\text{ième}}$ source est obtenue en prenant l'espérance de la probabilité *a posteriori* optimale de la variable des sources $q^*(\mathbf{s}_{fn})$ définie en (5.93), soit la moyenne $\boldsymbol{\mu}_{\mathbf{s}_{fn}}$ définie à l'équation (5.94). En développant son expression, nous obtenons la formulation suivante pour l'estimée de la source j :

$$\tilde{s}_{j,fn} = \frac{C_{j,fn}}{\sum_{j'=1}^{J} C_{j',fn} + \sigma^2}\, x_{fn}, \tag{5.101}$$

où $C_{j,fn}$ est cette fois défini selon (5.92).

5.3.5 Relation avec la fusion adaptative VB

Nous avons ci-dessus souligné que la probabilité *a posteriori* du nombre de composantes $q(K_j)$ (5.100) estimée pour le modèle NMF à ordre multiple prenait une expression très similaire à la probabilité *a posteriori* du modèle à NMF à ordre unique $p(\mathbf{K}_m|X)$ définie à l'équation (5.63).

Il est un cas en particulier où ces deux probabilités *a posteriori* expriment la probabilité de modèles équivalents. Notons dans un premier temps que le modèle NMF à ordre multiple est strictement équivalent au modèle à ordre unique lorsque $\forall j, M_j = 1$. Dans ce cas en effet, chaque source n'est décrite que par une seule NMF d'ordre K_j dont les probabilités *a priori* et *a posteriori* sont toutes deux égales à 1 ($\forall j, q(K_j) = p(K_j) = 1$).

Supposons à présent que nous ne souhaitons modéliser par M NMFs différentes que la source j. Les autres sources telles que $j' \neq j$ seront chacune modélisées par une NMF d'ordre $K_{j'}$. Par conséquent, pour la NMF à ordre unique, un modèle est parfaitement défini par le nombre de composantes K_{jm} de la $j^{\text{ième}}$ source. La probabilité *a posteriori* du modèle $\mathcal{M}_m$ peut donc s'écrire, selon (5.63) :

$$p(K_j = K_{jm}|\mathbf{X}) = \frac{1}{\delta}\pi_m e^{\mathcal{L}_m} \tag{5.102}$$

où nous avons simplement remplacé le terme $\mathbf{K}_m$ désignant les J nombres de composantes considérés par $K_j = K_{jm}$, indiquant ainsi que seul le nombre de composantes K_j varie entre nos M séparateurs à fusionner.

Pour la NMF à ordre multiple, les probabilités *a priori* et *a posteriori* des modèles des sources $j' \neq j$ seront toutes égales à 1. Seule la source j est définie à l'aide de plusieurs NMFs, chacune ayant une probabilité π_{jm} qui peut être identifiée à la probabilité du modèle à ordre unique K_{jm}, de sorte que $\pi_m = \pi_{jm}$. De la même manière, le terme $\mathcal{L}_{jm}$ se trouve être équivalent à l'énergie libre du modèle à ordre unique K_{jm}, donnant $\mathcal{L}_m = \mathcal{L}_{jm}$. La probabilité *a posteriori* (5.100) peut alors s'écrire :

$$q(K_j = K_{jm}) = \frac{1}{\delta}\pi_m e^{\mathcal{L}_m}. \tag{5.103}$$

On constate donc que les probabilités *a posteriori* des nombres de composantes sont dans ce cas strictement équivalentes, telles que $p(K_j = K_{jm}|\mathbf{X}) = q(K_j = K_{jm})$. Bien entendu, même si les probabilités expriment des probabilités de modèles équivalents, elles ne seront pas pour autant égales en pratique. En effet, dans le cas de la NMF à ordre unique, l'énergie libre $\mathcal{L}_m$, utilisée dans le calcul de la probabilité $p(K_j = K_{jm}|\mathbf{X})$ et dont le calcul a été détaillé à l'équation (5.58), dépend de termes relatifs aux sources $j' \neq j$. Ces termes varieront donc en fonction de m, ce qui n'est pas le cas pour la NMF à ordre multiple. En effet, dans le cas de la NMF à ordre multiple, l'énergie libre $\mathcal{L}_m$ dépend aussi de termes relatifs aux sources $j' \neq j$ mais ces termes seront identiques quel que soit m, puisque les M NMFs sont estimées conjointement et partagent les mêmes modèles NMF pour les sources $j' \neq j$.

Cette comparaison nous permettra cependant de mettre en place des méthodes d'apprentissage que nous introduirons dans la partie 5.5. Dans la suite, nous ne considérerons donc que ce cas particulier afin de pouvoir comparer la fusion VB au modèle NMF à ordre multiple ici présenté.

5.3.6 Extension aux fusions variant en temps et variant en fréquence

De même que pour le moyennage de NMFs à ordre unique, nous pouvons étendre le modèle NMF à ordre multiple afin que le moyennage des paramètres de NMF exprimé au travers du terme $C_{j,fn}$ (5.92) soit réalisé par trames ou par bandes de fréquence, opérant donc une fusion variant en temps ou variant en fréquence.

Fusion variant en temps

Afin de formuler le modèle NMF à ordre multiple variant en temps, il nous faut modifier notre modèle génératif. Nous supposerons ici que nous avons une variable aléatoire $K_{j,n}$ par trame temporelle. Quelle que soit la trame, nous supposerons que chaque nombre de composantes a une probabilité *a priori* $p(K_{j,n} = K_{jm,n}) = \pi_{jm}$ indépendante de la trame n. De ce fait, la probabilité *a posteriori* du nombre de composantes devient :

$$\begin{aligned}\log q^*(K_{j,n} = K_{jm}) = \quad & \sum_{fk} \Big(\mathbb{E}_{w_{jm,fk}}\big[\log p(w_{jm,fk})\big] \\ & \qquad - \mathbb{E}_{w_{jm,fk}}\big[\log q(w_{jm,fk})\big] \Big) \\ & + \sum_{k} \Big(\mathbb{E}_{h_{jm,kn}}\big[\log p(h_{jm,kn})\big] \\ & \qquad - \mathbb{E}_{h_{jm,kn}}\big[\log q(h_{jm,kn})\big] \Big) \\ & + \log \pi_{jm} + \text{cte.}\end{aligned} \tag{5.104}$$

La probabilité *a posteriori* peut alors se mettre sous une forme identique à la fusion VB variant en temps exprimée à l'équation (5.68) telle que :

$$q^*(K_{j,n} = K_{jm}) = \frac{1}{\delta_n} \pi_{jm} e^{\mathcal{L}_{jm,n}} \tag{5.105}$$

où $\mathcal{L}_{jm,n}$ désigne l'énergie libre du modèle avec $K_{j,n} = K_{jm}$ exprimée pour la trame n, soit l'exponentielle du terme de droite de l'équation (5.104).

Fusion variant en fréquence

De la même manière, nous pouvons supposer qu'une variable aléatoire $K_{j,f}$ est définie pour chaque bande de fréquence. Contrairement au cas variant en temps, nous supposons ici que chaque nombre de composantes a une probabilité *a priori* $p(K_{j,f} = K_{jm}) = \pi_{jm,f}$ définie indépendamment sur chaque bande de fréquence f. La probabilité *a posteriori* du nombre de composantes est alors donnée par

$$\begin{aligned}\log q^*(K_{j,f} = K_{jm}) = \quad & \sum_{k} \Big(\mathbb{E}_{w_{jm,fk}}\big[\log p(w_{jm,fk})\big] \\ & \qquad - \mathbb{E}_{w_{jm,fk}}\big[\log q(w_{jm,fk})\big] \Big) \\ & + \sum_{kn} \Big(\mathbb{E}_{h_{jm,kn}}\big[\log p(h_{jm,kn})\big] \\ & \qquad - \mathbb{E}_{h_{jm,kn}}\big[\log q(h_{jm,kn})\big] \Big) \\ & + \log \pi_{jm,f} + \text{cte.}\end{aligned} \tag{5.106}$$

La probabilité *a posteriori* prend alors une forme identique à la fusion VB variant en fréquence exprimée à l'équation (5.70) telle que :

$$q^*(K_{j,f} = K_{jm}) = \frac{1}{\delta_f} \pi_{jm,f} e^{\mathcal{L}_{jm,f}} \tag{5.107}$$

où $\mathcal{L}_{jm,f}$ désigne l'énergie libre du modèle avec $K_{j,f} = K_{jm}$ exprimée pour la bande de fréquence f, soit l'exponentielle du terme de droite de l'équation (5.106).

5.4 Distribution a posteriori du nombre de composantes

Quel que soit le modèle de NMF choisi, à ordre unique ou à ordre multiple, nous avons montré dans la partie précédente que l'expression de la probabilité *a posteriori* du nombre de composantes

K_j prenait la forme générale suivante lorsque les M modèles à fusionner ne se distinguaient que par le nombre de composantes choisi pour la $j^{\text{ième}}$ source :

$$p(K_j = K_{jm}|\mathbf{X}) \approx \frac{1}{\delta}\pi_m e^{\mathcal{L}_m} \tag{5.108}$$

où π_m désigne la probabilité *a priori* du nombre de composantes K_{jm} et $\mathcal{L}_m$ désigne l'énergie libre du modèle correspondant. L'énergie libre étant une borne inférieure de la vraisemblance marginale $p(\mathbf{X})$, elle forme un critère de sélection de modèle. Le modèle expliquant au mieux les données est celui dont l'énergie libre est la plus grande. Le moyennage bayésien nous permet d'aller plus loin que ce simple critère de sélection et de tenir compte de l'incertitude induite par chaque modèle $\mathcal{M}_m$ en les pondérant par leurs probabilités *a posteriori*, ici $p(K_j) = (K_{jm}|\mathbf{X})$.

Comme nous l'avons souligné dans la partie 5.2.2, la probabilité *a posteriori* peut être identifiée aux coefficients de fusion invariante introduits dans la section 3.2.1, tels que $\alpha_m = p(K_j = K_{jm}|\mathbf{X})$. Toutefois, nous allons montrer ici que la formulation bayésienne de la fusion ne nous permet pas, en l'état, d'exploiter cette probabilité comme poids effectif de fusion. Dans la partie 5.4.1 suivante, nous démontrerons à l'aide de tests synthétiques construits à partir du corpus *CHiME*, que la probabilité *a posteriori* $p(K_{jm}|\mathbf{X})$ s'avère en pratique toujours égale à 1 pour l'un des modèles et nulle pour les autres, opérant ainsi pratiquement une sélection, et non une fusion des modèles. Pour y remédier, nous proposerons dans la partie 5.4.2 l'introduction d'un paramètre de contrôle de l'entropie de la probabilité *a posteriori* qui permet, comme nous le vérifierons dans la partie 5.4.3, de rendre effective la fusion.

5.4.1 Tests synthétiques préliminaires

Dans cette partie, nous allons analyser le comportement des modèles bayésiens génératifs introduits dans ce chapitre. Pour ce faire, nous proposons de générer des exemples synthétiques respectant les modèles génératifs représentés sur les figures 5.1 et 5.2 et les distributions *a priori* choisies pour leurs paramètres. Les données synthétiques ont été générées à partir du corpus *CHiME* introduit dans la partie 3.3.2.

Génération des exemples synthétiques

Pour commencer, nous avons sélectionné aléatoirement l'un des locuteurs du corpus (en l'occurence, le $8^{\text{ième}}$ locuteur) ainsi que 10 secondes de bruit. Nous avons scindé l'ensemble d'apprentissage de ce locuteur en deux parts égales composées chacune de 250 séquences de mots. Sur la première partie de cet ensemble, nous avons appris $M = 7$ modèles du locuteur pour sept nombres de composantes $K_{1m} = 2^m$ avec $m = 1..7$, suivant la méthode détaillée dans la partie 3.3.4. Les M dictionnaires $\mathbf{W}_{1m}^{(1)}$ ainsi appris décrivent donc le même contenu spectral mais à des niveaux de précision différents. Ainsi, le dictionnaire $\mathbf{W}_{11}^{(1)}$ donne une description grossière des caractéristiques spectrales du locuteur avec seulement deux composantes alors que le dictionnaire $\mathbf{W}_{17}^{(1)}$ donne une description bien plus détaillée avec 128 composantes, au risque de présenter certaines redondances. Un modèle de bruit à 16 composantes $\mathbf{W}_2$ a été par ailleurs appris suivant la méthode exposée dans la partie 3.3.3. A chaque fois, nous avons utilisé la TFCT avec une fenêtre sinusoïdale de 2048 échantillons et un recouvrement de moitié.

Pour chaque ordre K_{1m}, nous avons généré plusieurs mélanges $\mathbf{X}$ de 300 trames temporelles chacun, selon le modèle génératif de la NMF à ordre unique représenté sur la figure 5.1. Les dictionnaires du locuteur $\mathbf{W}_{1m}$ et de bruit $\mathbf{W}_2$ ont été fixés aux valeurs préalablement apprises. Les matrices d'activation associées $\mathbf{H}_{1m}$ et $\mathbf{H}_2$ ont été tirées aléatoirement selon leurs distributions *a priori* de type gamma définies à l'équation (5.6) avec $b = 0.2$. Les sources de parole s_1 et de bruit s_2 ont été générées aléatoirement selon la distribution *a priori* définie en (5.2). Le bruit de

capteur ϵ_{fn} a été choisi de variance constante $\sigma^2 = 10^{-6}$. Enfin, l'observation $\mathbf{X}$ a été générée selon l'équation de mélange (5.1) à six RSBs différents, variant de -6 dB à 9 dB par pas de 3 dB. Au total, 42 mélanges synthétiques ont été générés, pour six RSBs différents et les sept nombres de composantes définis plus tôt, soit un mélange par couple de RSB et de nombre de composantes.

Apprentissage d'un modèle de locuteur

La seconde partie de l'ensemble d'apprentissage relative à ce locuteur a été utilisée afin d'apprendre $M = 7$ autres modèles du locuteur pour les sept mêmes nombres de composantes $K_{1m} = 2^m$ avec $m = 1..7$. Les M dictionnaires $\mathbf{W}_{1m}^{(2)}$ ainsi appris sont donc différents des M dictionnaires $\mathbf{W}_{1m}^{(1)}$ appris pour générer les mélanges. Les dictionnaires $\{\mathbf{W}_{1m}^{(1)}\}_{m=1..7}$ et $\{\mathbf{W}_{1m}^{(2)}\}_{m=1..7}$ composent donc un total de quatorze modèles différents d'un même locuteur. On notera bien pour la suite que seul les dictionnaires $\{\mathbf{W}_{1m}^{(1)}\}_{m=1..7}$ ont été utilisés pour la génération des exemples.

Séparation

Nous proposons quatre cas d'étude dont les caractéristiques sont définies dans le tableau 5.3 selon l'observation considérée et le modèle de parole choisi. Quel que soit le cas considéré, la séparation a été menée pour chacun des nombres de composantes envisagés K_{1m}. Pour les cas 1 et 2, l'observation considérée n'est composée que de la source de parole synthétisée s_1 et du bruit de capteur ϵ_{fn}. Pour les cas 3 et 4, le mélange des sources de parole et de bruit a été considéré. Nous avons aussi fait varier les modèles de parole utilisés pour la source s_1. Dans les cas 1 et 3, ce sont les dictionnaires $\mathbf{W}_{1m}^{(1)}$ utilisés pour générer les exemples qui ont été également utilisés pour l'étape de séparation alors que dans les cas 2 et 4, nous avons utilisé les dictionnaires $\mathbf{W}_{1m}^{(2)}$ pour la séparation, ces dictionnaires étant totalement indépendants de l'étape de génération des exemples. Quel que soit l'exemple, le dictionnaire utilisé à la génération sera donc toujours présent parmi les M modèles de parole dans les cas 1 et 3. Dans les cas 2 et 4, nous aurons toujours un dictionnaire de taille identique à celui utilisé pour la génération mais ce dernier n'aura pas été appris sur les mêmes données. Enfin, nous noterons que le cas 4 est celui qui se rapproche le plus du cas réel déjà étudié dans les chapitres précédents.

Chaque exemple synthétique a été séparé dans les quatre cas sus-nommés à l'aide du formalisme bayésien introduit dans la partie 5.1. Les distributions relatives aux dictionnaires de la source s_1 ont été initialisées comme des distributions de Dirac, avec pour valeur les dictionnaires appris $\mathbf{W}_{1m}^{(1)}$ ou $\mathbf{W}_{1m}^{(2)}$ selon le cas. En pratique, cela revient simplement à poser

$$\mathbb{E}\left[\mathbf{W}_{1m}\right] = \mathbb{E}\left[1/\mathbf{W}_{1m}\right]^{-1} = \mathbf{W}_{1m}^{(i)} \tag{5.109}$$

avec $i = 1$ ou $i = 2$.

Pour les cas 3 et 4, les distributions relatives aux dictionnaires de la source s_2 ont été initialisées par leurs statistiques $\mathbb{E}\left[\mathbf{W}_2\right]$ et $\mathbb{E}\left[1/\mathbf{W}_2\right]$, la statistique $\mathbb{E}\left[\log \mathbf{W}_2\right]$ n'étant pas utilisée dans les règles de mise à jour présentées dans le tableau 5.1. Le terme $\mathbb{E}\left[\mathbf{W}_2\right]$ est initialisé selon l'*a priori* gamma défini en (5.6) avec $a = 0.2$ et $\mathbb{E}\left[\mathbf{W}_2^{-1}\right]$ est posé égal à $\mathbb{E}\left[\mathbf{W}_2\right]^{\cdot -1}$. De la même manière ont été initialisés $\mathbb{E}\left[\mathbf{H}_{1m}\right]$, $\mathbb{E}\left[1/\mathbf{H}_{1m}\right]$, $\mathbb{E}\left[\mathbf{H}_2\right]$ et $\mathbb{E}\left[1/\mathbf{H}_2\right]$. Enfin, pour tous les cas, la distribution $q(\mathbf{W}_{1m})$ est supposée fixée alors que les distributions $q(\mathbf{W}_2)$, $q(\mathbf{H}_{1m})$ et $q(\mathbf{H}_2)$ (pour les cas 2 et 4) seront estimées à l'aide des mises à jour du tableau 5.1. Le nombre d'itérations a été fixé à 50 et au final, les sources estimées dans le domaine temporel ont été obtenues par filtrage de Wiener selon (2.24).

	Observation	Modèle de parole
Cas 1	parole seule	$\mathbf{W}_{1m}^{(1)}$
Cas 2		$\mathbf{W}_{1m}^{(2)}$
Cas 3	parole et bruit	$\mathbf{W}_{1m}^{(1)}$
Cas 4		$\mathbf{W}_{1m}^{(2)}$

TABLEAU 5.3 – Cas d'étude pour la validation sur tests synthétiques du principe de sélection VB

Sélection variationnelle bayésienne

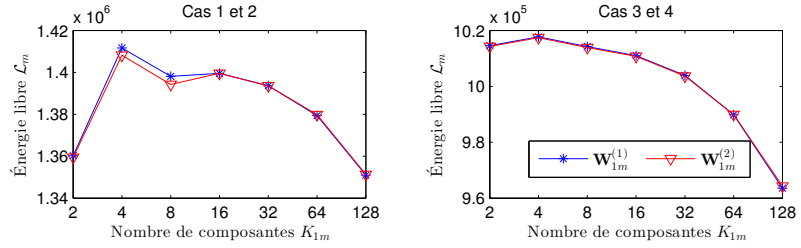
Dans un premier temps, nous proposons d'analyser les performances de la sélection variationnelle bayésienne (VB). Selon le principe déjà invoqué plus tôt, la sélection VB consiste à sélectionner le modèle NMF ayant la plus grande énergie libre. Grâce à nos mélanges synthétiques, nous pouvons analyser la fiabilité de ce critère pour retrouver le nombre de composantes qui a effectivement permis de générer chaque mélange.

Pour six des 42 exemples synthétiques, la figure 5.3 représente les valeurs d'énergies libres calculées pour chacun des $M = 7$ nombres de composantes et pour chacun des quatre cas d'étude. Les exemples considérés ont été générés pour $K_1 = 4$ composantes et des RSBs de -6 dB (sous-figure 5.3a), 0 dB (sous-figure 5.3b) et 6 dB (sous-figure 5.3c). De la même manière, les exemples considérés ont été générés pour $K_1 = 64$ composantes et des RSBs de -6 dB (sous-figure 5.3d), 0 dB (sous-figure 5.3e) et 6 dB (sous-figure 5.3f). Sur chacune de ces sous-figures sont représentées à gauche les énergies libres pour les cas 1 et 2, et à droite pour les cas 3 et 4.

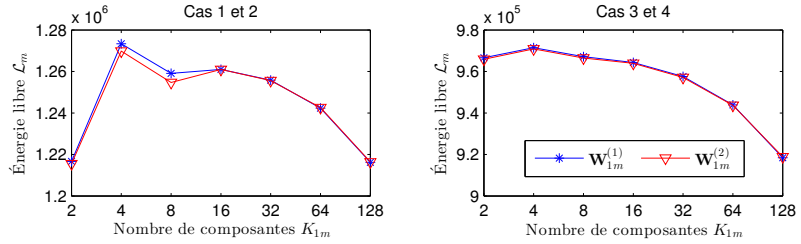
Parole seule (cas 1 et 2) : Généralement, on constate que dans le cas 1, lorsque le mélange n'est composé que de la parole seule et que le dictionnaire $\mathbf{W}_{1m}$ est fixé aux valeurs $\mathbf{W}_{1m}^{(1)}$ utilisées pour générer les exemples, l'énergie libre permet toujours de retrouver le nombre de composantes qui a effectivement permis de générer l'exemple considéré. Ainsi à gauche des figures 5.3a, 5.3b et 5.3c, le maximum de la courbe bleue se trouve toujours en $K_{1m} = 4$. De même, le maximum des courbes bleues à gauche des figures 5.3d, 5.3e et 5.3f est toujours situé en $K_{1m} = 64$. Il en est de même lorsque le dictionnaire $\mathbf{W}_{1m}$ est fixé aux valeurs $\mathbf{W}_{1m}^{(2)}$ distinctes de celles utilisées pour générer les exemples (courbe rouge, sur les sous-figures de gauche).

Parole et bruit (cas 3 et 4) : Lorsque l'observation est composée de parole et de bruit (partie droite des sous-figures), nous constatons que pour le nombre de composantes $K_1 = 4$, le maximum des courbes rouge et bleue se trouve toujours au bon endroit, en $K_{1m} = 4$, quel que soit le RSB. Toutefois, quand $K_1 = 64$, l'énergie libre ne permet plus de déterminer à coup sûr le nombre de composantes utilisé pour générer les exemples. C'est en effet le cas notamment pour un RSB fabile de -6 dB, même lorsque le dictionnaire $\mathbf{W}_{1m}$ est fixé aux valeurs $\mathbf{W}(1)_{1m}$ utilisées pour générer les exemples (cas 3, courbe bleue). De plus, lorsque ce sont les dictionnaires appris sur l'ensemble de test qui sont choisis pour l'estimation, alors quel que soit le RSB, le maximum de la courbe rouge ne se trouve pas au bon endroit. Ceci montre bien les limites de la sélection VB par l'énergie libre.

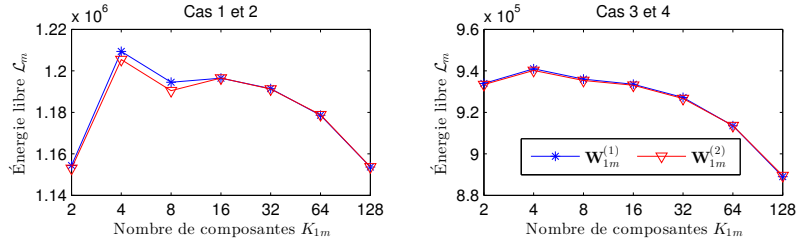
Explication : Ce comportement peut être plus largement observé sur l'ensemble des exemples grâce à la figure 5.4. Sur cette figure sont représentés, en noir, les modèles sélectionnés par le maximum de l'énergie libre pour chacun des nombres de composantes K_1 utilisés à la génération et chacun des RSBs. Nous constatons ainsi que la sélection VB ne fonctionne que dans le cas 1, c'est-à-dire lorsque ce sont les dictionnaires utilisés pour générer les exemples qui sont également



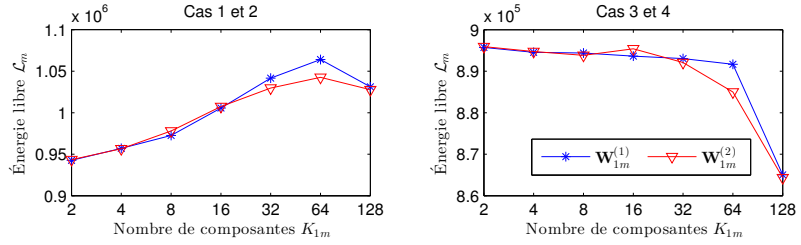
(a) $K_1 = 4$, RSB $= -6$ dB



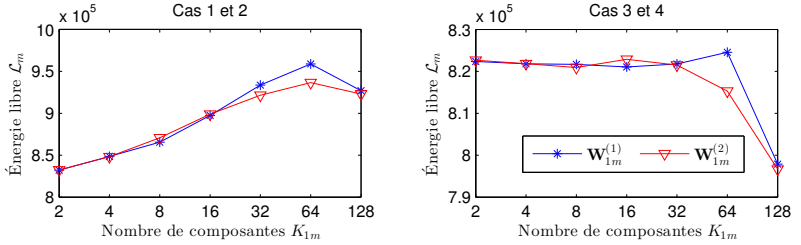
(b) $K_1 = 4$, RSB $= 0$ dB



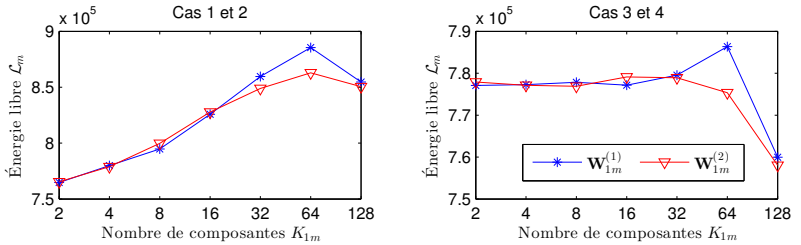
(c) $K_1 = 4$, RSB $= 6$ dB



(d) $K_1 = 64$, RSB $= -6$ dB



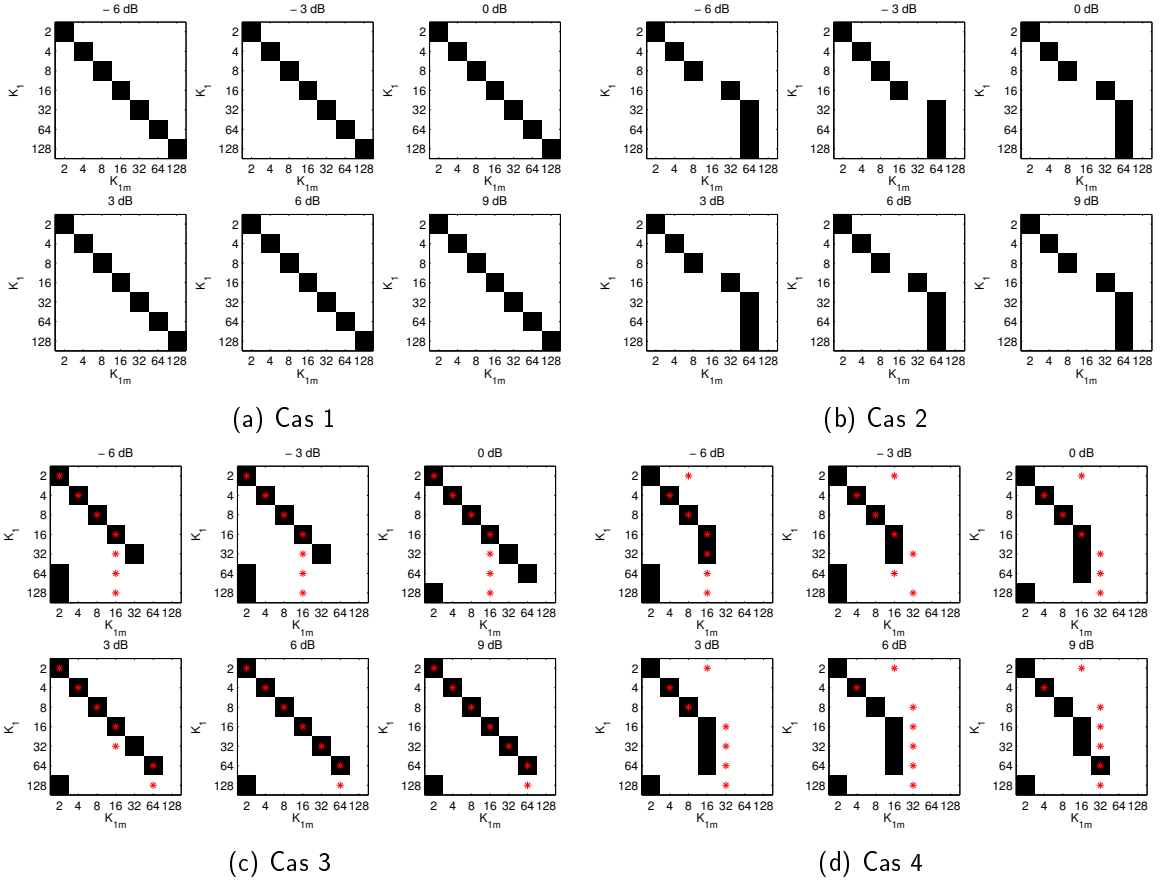
(e) $K_1 = 64$, RSB $= 0$ dB



(f) $K_1 = 64$, RSB $= 6$ dB

FIGURE 5.3 – Valeurs des énergies libres estimées pour les quatre cas d'étude envisagés, pour les $M = 7$ modèles considérés et pour six exemples synthétiques.

utilisés comme modèle de parole et que l'observation n'est composée que du signal de parole. Dans ce cas en effet, quel que soit le RSB, la matrice représentée est diagonale. Dans tous les autres cas, nous observons que la sélection VB fonctionne pour les nombres de composantes faibles et moins bien pour les nombres de composantes plus élevés. Ce comportement peut être expliqué simplement. En effet, pour des nombres de composantes K_1 élevés, la source de parole générée a nécessairement un contenu spectral plus riche que pour des nombres de composantes plus faibles. Il devient donc plus compliqué de modéliser cette source, lorsque nous ne connaissons pas le dictionnaire utilisé pour la générer, et d'autant plus lorsque cette source est mélangée au bruit. Dans ce cas, le critère du maximum de vraisemblance marginale sélectionne plutôt un modèle d'ordre plus faible que celui choisi à la génération. Par ailleurs, nous rappellerons que l'inférence VB ici employée fait appel à deux étapes d'approximation, la première étant intrinsèque au principe de ce type d'inférence, et la deuxième étant due à la nécessité de trouver une forme analytique à une partie de l'énergie libre. De ces deux faits, il vient que la borne inférieure de l'énergie libre que nous exploitons ici comme critère de sélection ne permet pas de sélectionner le meilleur modèle au sens bayésien, comme le permet en théorie la vraisemblance marginale.



(a) Cas 1 (b) Cas 2

(c) Cas 3 (d) Cas 4

FIGURE 5.4 – Maximum des énergies libres en fonction du nombre de composantes à la génération K_1, pour chacun des 6 RSBs. Pour les cas 3 et 4, est aussi indiqué par une étoile rouge le modèle qui donne le meilleur SDR pour chaque K_1.

Relation avec la qualité de séparation : Pour les cas 3 et 4 qui simulent des cas réalistes de séparation de sources, nous avons également tracé à l'aide d'étoiles rouges sur la figure 5.4 les modèles donnant le meilleur SDR pour la source de parole, pour chacun des nombres de composantes K_1. On remarquera qu'il n'y a pas de corrélation immédiate entre maximum de

l'énergie libre et meilleur modèle en terme de SDR. Bien souvent, le modèle K_{1m} donnant le meilleur SDR n'est pas celui dont l'énergie libre est maximale. Ceci est d'autant plus vérifié pour le cas 4 où ne sont pas connus les dictionnaires utilisés à la génération. Ainsi, même si l'énergie libre permet parfois de sélectionner le bon modèle au sens bayésien, elle ne semble pas être un bon critère de sélection au sens du SDR, qui lui est corrélé à notre objectif de séparation de sources.

Fusion variationnelle bayésienne

Si l'énergie libre ne semble pas constituer un critère de sélection pertinent pour la séparation de sources, nous proposons tout de même d'analyser ici son utilisation dans un cadre de fusion, comme proposé dans les parties 5.2 et 5.3. En particulier, l'application du principe de moyennage bayésien au modèle NMF à ordre unique nous a permis de donner une interprétation probabiliste aux coefficients de fusion invariante α_m introduits au chapitre 3. Nous avons également étendu ce principe aux cas de fusions variant en temps et variant en fréquence dans la partie 5.3.6. Toutefois, nous ne nous intéresserons ici qu'au cas invariant.

Les coefficients de fusion invariante VB mettent en jeu l'énergie libre $\mathcal{L}_m$ et la probabilité *a priori* π_m de chaque nombre de composantes. La figure 5.5 présente les coefficients de fusion obtenus par application de l'équation (5.66) sur six de nos exemples synthétiques et pour les cas d'étude 3 et 4 précédemment définis. Notons que nous avons supposé que chaque modèle était équiprobable *a priori*, c'est-à-dire que $\forall m \in [1, M], \pi_m = 1/M$. Nous constatons sur les sous-figures de droite que quel que soit l'exemple, les coefficients de fusion obtenus n'opèrent pas en pratique une fusion comme attendu mais plutôt une sélection. En effet, à chaque fois, un seul des coefficients de fusion calculés est non nul et égal à 1. Ce résultat s'explique par les valeurs très élevées des énergies libres $\mathcal{L}_m$ estimées et qui ont pour effet, lorsqu'on en prend l'exponentielle, de donner la totalité du poids au modèle ayant l'énergie libre la plus forte. La fusion VB telle quelle est donc équivalente à la sélection VB.

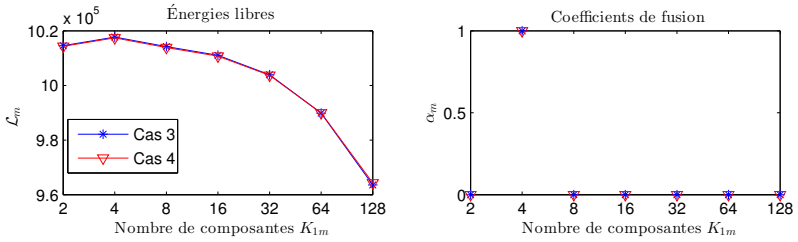
5.4.2 Paramètre de contrôle de l'entropie

Dans la partie précédente, nous avons montré que l'application directe du principe de moyennage bayésien pour déterminer des coefficients de fusion ne rendait pas la fusion effective puisqu'elle menait aux mêmes résultats que la sélection VB. Pourtant, nos expériences oracles menées sur le corpus CHiME dans la partie 3.3 ont montré que la fusion de NMFs de différents ordres pouvait apporter des gains substantiels en terme de SDR par rapport à la simple sélection. Pour atteindre cet objectif de fusion, nous proposons ici d'introduire un paramètre de mise à l'échelle, noté β, de sorte que la définition des coefficients de fusion invariante VB (5.66) se trouve modifiée comme suit :

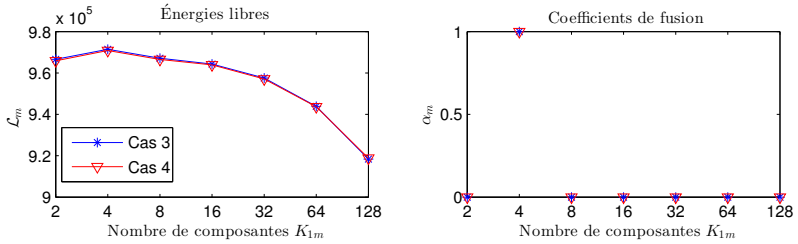
$$\alpha_m = p(K_1 = K_{1m}|\mathbf{X}) = \frac{1}{\delta}\pi_m e^{\mathcal{L}_m/\beta} \quad \text{avec} \quad \beta \geq 1. \tag{5.110}$$

La figure 5.6 illustre l'effet de différentes valeurs de β sur la distribution *a posteriori*, pour les six exemples déjà exploités dans la partie précédente et pour le cas d'étude numéro 4. Nous constatons que l'introduction de ce paramètre permet effectivement d'obtenir des coefficients de fusion tous non nuls. Le cas $\beta = 1$ correspond à la sélection VB alors que le cas limite $\beta \to \infty$ entraîne $\alpha_m = \pi_m = 1/M$. Une valeur de β entre ces deux valeurs extrêmes permet donc de faire un compromis entre la probabilité *a priori* $p(K_1 = K_{1m}) = \pi_m$ et la vraisemblance marginale $p(\mathbf{X}|K_{1m})$.

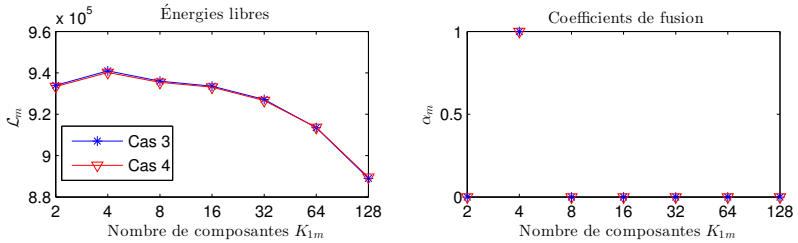
Plus généralement, l'introduction de ce paramètre peut s'expliquer d'un point de vue théorique comme pénalisant l'entropie de la distribution *a posteriori* des nombres de composantes $q(K_{jm})$ dans le cas du modèle NMF à ordre multiple, de façon similaire à l'inférence VB par recuit proposée dans [Katahira et al., 2008]. La justification peut être trouvée en annexe E.3.



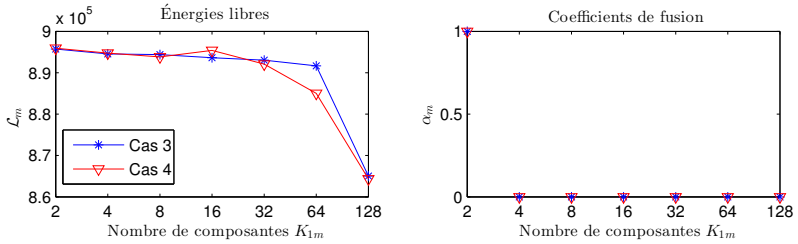
(a) $K_1 = 4$, RSB $= -6$ dB



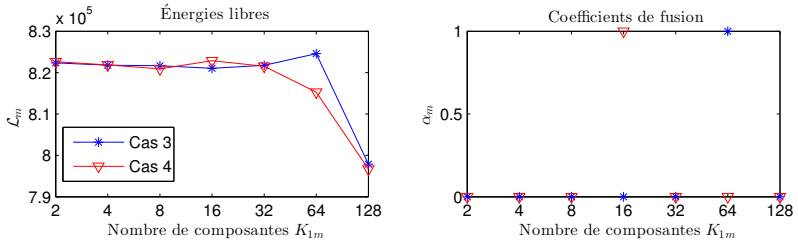
(b) $K_1 = 4$, RSB $= 0$ dB



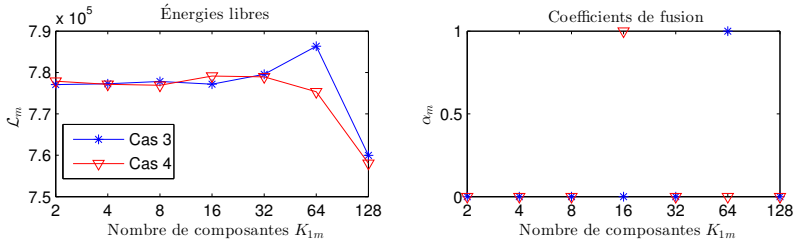
(c) $K_1 = 4$, RSB $= 6$ dB



(d) $K_1 = 64$, RSB $= -6$ dB

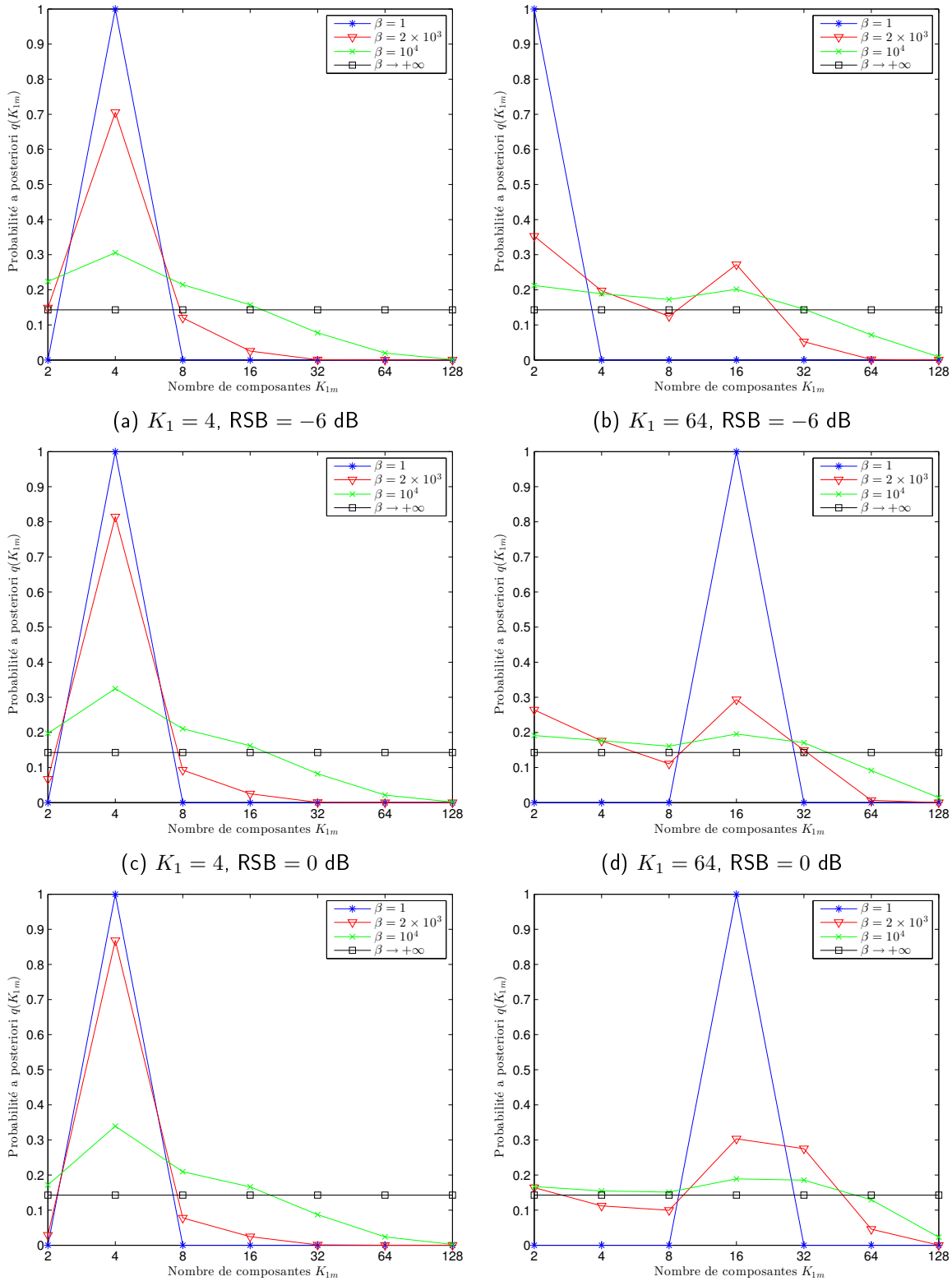


(e) $K_1 = 64$, RSB $= 0$ dB



(f) $K_1 = 64$, RSB $= 6$ dB

FIGURE 5.5 – Valeurs des énergies libres et des coefficients de fusion VB correspondants pour les cas 3 et 4 et pour les $M = 7$ modèles considérés, sur six exemples synthétiques.



(a) $K_1 = 4$, RSB $= -6$ dB

(b) $K_1 = 64$, RSB $= -6$ dB

(c) $K_1 = 4$, RSB $= 0$ dB

(d) $K_1 = 64$, RSB $= 0$ dB

(e) $K_1 = 4$, RSB $= 6$ dB

(f) $K_1 = 64$, RSB $= 6$ dB

FIGURE 5.6 – Effet du paramètre de contrôle de l'entropie β sur la probabilité *a posteriori* du nombre de composantes $q(K_{1m})$ sur six exemples synthétiques, dans le cas 4.

5.4.3 Validation sur tests synthétiques

Nous proposons dans cette partie de valider sur nos données synthétiques l'utilisation du paramètre β pour rendre effective la fusion et améliorer la qualité de séparation par rapport à la simple sélection. Pour chacun des exemples, nous proposons de comparer la sélection VB aux deux approches de fusion adaptative invariante proposées plus tôt : la fusion invariante VB de la partie 5.2.2 et le modèle NMF à ordre multiple de la partie 5.3. Par souci de concision, nous ferons référence à ce dernier par l'acronyme *mo-NMF*, pour *multiple-order NMF* en anglais. Quel que soit le type de modèle NMF envisagé pour la fusion, à ordre unique ou à ordre multiple, les initialisations ont été réalisées selon le processus détaillé dans la partie 5.4.1. Nous n'étudierons ici que le cas 4, afin de simuler au mieux le cas réel qui sera étudié sur le corpus CHiME. Le mélange considéré est donc composé d'une source de parole et d'une source de bruit. Les dictionnaires retenus pour modéliser la parole sont les dictionnaires $\mathbf{W}_{1m}^{(2)}$ précédemment introduits et sont donc distincts des dictionnaires $\mathbf{W}_{1m}^{(1)}$ utilisés pour générer les exemples. La distribution *a priori* du nombre de composantes est supposée uniforme de sorte que $\forall m \in [1, M], p(K_1 = K_{1m}) = 1/M$. Enfin, nous avons choisi $\beta = 10^4$ car cette valeur nous a semblé bien fonctionner lors de tests préliminaires.

Les résultats numériques pour la sélection invariante VB, la fusion invariante VB et le modèle mo-NMF invariant sont donnés dans le tableau 5.4. À titre de comparaison, nous donnons aussi les résultats de sélection invariante oracle et de fusion par moyenne. Les résultats ont été moyennés indépendamment pour chaque RSB ainsi que sur l'ensemble des 42 extraits synthétisés. La dernière colonne du tableau donne le temps moyen de calcul pour un exemple de chacune des méthodes proposées. Nous pouvons ainsi constater que l'ajout du paramètre β nous permet de rendre effective nos approches de fusion VB. En effet, sans ce paramètre (ou avec $\beta = 1$), la fusion VB aurait donné les mêmes résultats que la sélection VB. Avec $\beta = 10^4$, la fusion VB permet de gagner près de 0.4 dB sur nos tests synthétiques. Par ailleurs, le modèle mo-NMF permet un gain de 0.5 dB par rapport à la simple sélection, soit un gain de 0.15 dB par rapport à la fusion VB. De plus, nous constatons que le modèle mo-NMF, qui estime conjointement les M modèles NMFs au lieu de les estimer indépendamment les uns des autres, permet quasiment de diviser le temps de calcul par 3 par rapport à la fusion VB.

	RSB							Temps de calcul
	-6 dB	-3 dB	0 dB	3 dB	6 dB	9 dB	Moyenne	moyen (s)
Sélection VB	3.77	4.36	5.20	5.72	6.19	6.40	5.27	172.7
Fusion VB avec $\beta = 10^4$	**4.32**	4.88	5.36	5.96	6.48	6.82	5.64	172.7
mo-NMF avec $\beta = 10^4$	4.24	**4.90**	**5.47**	**6.14**	**6.76**	**7.21**	**5.79**	**60.8**
Sélection oracle	4.35	5.04	5.69	6.45	7.14	7.77	6.07	172.7
Fusion par moyenne	3.81	4.57	5.22	5.91	6.53	7.04	5.52	172.7

TABLEAU 5.4 – Peformance moyenne (SDR en dB) de séparation par fusion adaptative invariante VB de modèles NMF à ordre unique et par fusion adaptative invariante par modèle à NMF à ordre multiple. Le temps de calcul moyen par exemple est indiqué dans la dernière colonne.

Il résulte donc de ces expériences synthétiques que le paramètre β nous permet de rendre effective la fusion inspirée par le moyennage bayésien. De plus, le modèle NMF à ordre multiple nous permet un gain de temps de calcul certain, pour des résultats comparables à la fusion VB. Il convient donc à présent de confirmer ces résultats prometteurs sur données réelles.

5.5 Expériences et discussion

Dans cette partie, nous proposons d'évaluer sur le corpus *CHiME* les modèles et approches de fusion introduites dans les parties précédentes. Dans un premier temps, nous définirons dans la

partie 5.5.1 les séparateurs choisis pour la fusion et comparerons leurs performances individuelles aux modèles NMF standard déjà exploités et analysés dans les chapitres précédents. Nous mettrons ensuite en œuvre les approches de fusion adaptative basées sur le calcul de l'énergie libre des modèles, à savoir la fusion VB exposée dans la partie 5.2 et le modèle NMF à ordre multiple introduit dans la partie 5.3. Pour ce faire, nous introduirons dans la partie 5.5.2 différentes méthodes permettant d'apprendre les probabilités *a priori* des nombres de composantes ainsi que le paramètre de contrôle de l'entropie β sur l'ensemble d'apprentissage de notre corpus. Nous évaluerons ensuite ces méthodes dans la partie 5.5.3.

5.5.1 Performances individuelles

Comme dans les chapitres précédents, nous nous intéressons ici à l'estimation de la parole dans un mélange bruité à l'aide de $M = 7$ séparateurs. Chaque séparateur est défini par le nombre de composantes $K_{1m} = 2^m$ choisi pour représenter la source de parole s_1. La structure spectrale retenue pour le modèles de parole est de type NMF simple, bien que les structures de type EF, présentées dans la partie 3.3.1, soient parfaitement compatibles avec le modèle NMF bayésien introduit dans ce chapitre. De même, le bruit sera modélisé par un modèle NMF simple. Comme pour les expériences précédentes, nous avons choisi d'utiliser la transformée QERB avec une fenêtre sinusoïdale de 1024 échantillons et $F = 350$ bandes de fréquence.

Nous proposons ici de comparer le modèle NMF standard déjà étudié dans les chapitres précédents (que nous noterons *ML-NMF*, pour *Maximum-Likelihood NMF* en anglais) au modèle bayésien introduit dans ce chapitre (que nous désignerons par *VB-NMF* pour *Variational Bayesian NMF*). Nous comparerons également les performances de la VB-NMF selon le choix de la distribution *a priori* des paramètres NMF, de type gamma (5.6) ou Jeffreys (5.8). Nous nous référerons au premier par le terme *VB-NMF gamma* et au second par le terme *VB-NMF Jeffreys*.

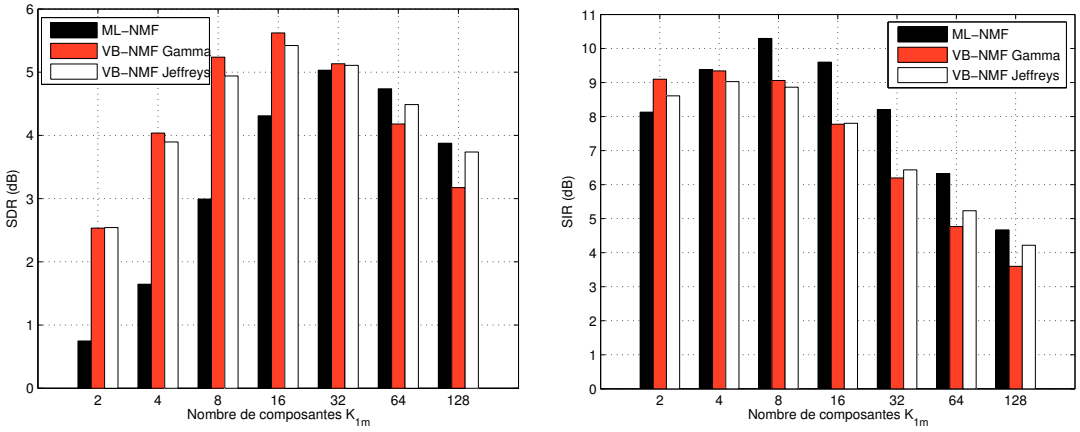
Quel que soit le type d'*a priori* choisi, les dictionnaires de chacune des sources ont été appris selon les mêmes procédés décrits dans la partie 3.3.4 pour l'apprentissage du modèle de parole et dans la partie 3.3.3 pour l'apprentissage du modèle de bruit. Notons que cette étape d'apprentissage a été menée avec la formulation ML-NMF. Par conséquent, pour un locuteur donné, les mêmes dictionnaires à K_{1m} composantes ont donc été utilisés pour tous les modèles NMF envisagés (ML-NMF, VB-NMF gamma et VB-NMF Jeffreys). Similairement, pour un exemple donné, le même dictionnaire d'ordre K_2 a été utilisé pour modéliser le bruit. Comme dans nos expériences précédentes, nous avons fixé $K_2 = 16$.

L'initialisation des paramètres des modèles VB-NMF a été menée de façon identique aux tests synthétiques de la partie 5.4.1. Plus précisément, seules les distributions relatives aux matrices d'activation $q(\mathbf{H}_j)$ seront à estimer lors de la séparation du mélange bruité, les distributions relatives aux dictionnaires $q(\mathbf{W}_j)$ restant fixées à leurs valeurs apprises. Pour le modèle $\mathcal{M}_m$, les distributions $q(\mathbf{W}_j)$ ont donc été initialisées comme des distributions de Dirac, avec pour valeur les dictionnaires appris $\mathbf{W}_{1m}$ pour la source de parole et $\mathbf{W}_2$ pour la source de bruit. Les distributions $q(\mathbf{H}_j)$ ont été initialisées par leurs statistiques $\mathbb{E}[\mathbf{H}_j]$ and $\mathbb{E}[1/\mathbf{H}_j]$. Comme pour la ML-NMF, plutôt que d'initialiser aléatoirement ces distributions, nous avons utilisé les valeurs moyennes des activations obtenues lors de l'apprentissage notées $\widetilde{\mathbf{H}}_j$ (voir parties 3.3.4 et 3.3.3) de sorte que :

$$\forall j, \quad \mathbb{E}[\mathbf{H}_j] = \widetilde{\mathbf{H}}_j \quad \text{et} \qquad \mathbb{E}\left[\frac{1}{\mathbf{H}_j}\right] = \frac{1}{\widetilde{\mathbf{H}}_j}. \tag{5.111}$$

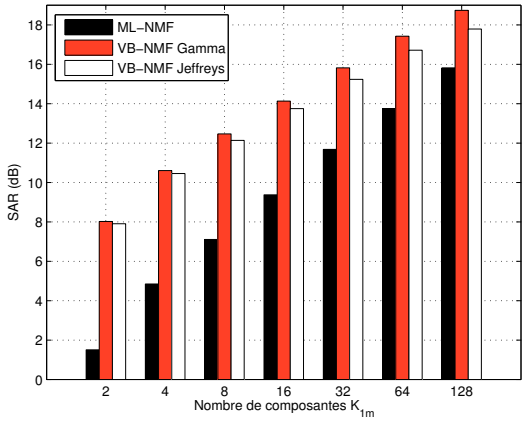
Pour le cas de l'*a priori* gamma, nous avons choisi $b = 0.2$ dans l'équation (5.6). Finalement, le nombre d'itérations a été fixé à 50 et les sources estimées dans le domaine temporel ont été obtenues par filtrage de Wiener selon (5.54).

Les performances mesurées sur la source de parole pour les $M = 7$ nombres de composantes et les trois modèles envisagés sont représentées sur la figure 5.7. Les mesures ont été moyennées



(a) SDR

(b) SIR



(c) SAR

FIGURE 5.7 – Performance (SDR, SIR, SAR) individuelle des séparateurs en fonction du nombre de composantes, pour la NMF standard (*ML-NMF*) et la NMF bayésienne avec *a priori* gamma (*VB-NMF gamma*) et *a priori* Jeffreys (*VB-NMF Jeffreys*).

sur les ensembles de validation et de test. Les valeurs numériques sont reportées dans le tableau 5.5, indépendamment pour l'ensemble de validation et l'ensemble de test. Sur la sous-figure 5.7a représentant le SDR de la source de parole, nous pouvons constater que le modèle VB-NMF gamma à 16 composantes est celui qui donne le meilleur SDR. On constatera aussi que les modèles VB-NMF fonctionnent mieux que le modèle ML-NMF lorsque le nombre de composantes choisi est faible (presque 2 dB de gain pour $K_{1m} = 2$), alors que pour les nombres de composantes $K_{1m} = 64$ et $K_{1m} = 128$, le modèle ML-NMF est celui qui donne le meilleur SDR. Enfin, on notera que l'*a priori* gamma semble donner de meilleurs résultats que l'*a priori* de Jeffreys pour les nombres de composantes faibles, cette tendance s'inversant pour $K_{1m} = 64$ et $K_{1m} = 128$.

Concernant le SIR illustré sur la sous-figure 5.7b, on constatera que les modèles VB-NMF donnent des SIR sensiblement plus faibles que les modèles ML-NMF correspondants (sauf pour $K_{1m} = 2$). Comme nous l'avions déjà remarqué pour le modèle ML-NMF dans le chapitre 3, le SIR a tendance à décroître avec le nombre de composantes. À l'inverse, le SAR a lui tendance à croître avec le nombre de composantes, comme l'illustre la sous-figure 5.7c. Toutefois, les modèles VB-NMF permettent un gain très net de SAR comparativement aux modèles ML-NMF, et ce quel que soit le nombre de composantes. En moyenne, ce gain s'élève à 4.6 dB.

Nous retiendrons donc que les modèles VB-NMF introduits dans ce chapitre permettent d'améliorer la qualité globale de séparation et engendrent moins d'artefacts que le modèle ML-NMF précédemment étudié, au prix toutefois d'une perte en SIR. De plus, nous noterons que l'*a priori* gamma semble donner des performances sensiblement supérieures à l'*a priori* de Jeffreys.

Modèle		SDR		ISR		SIR		SAR	
Type	K_{1m}	valid	test	valid	test	valid	test	valid	test
	2	0.77	0.73	1.00	0.98	8.74	7.52	1.68	1.33
	4	1.55	1.74	2.39	2.45	9.07	9.68	4.72	4.98
	8	2.94	3.04	4.50	4.63	**10.21**	**10.38**	6.88	7.34
ML-NMF	16	4.11	4.51	7.28	7.60	9.35	9.85	9.27	9.47
	32	**4.92**	**5.14**	10.78	10.83	8.04	8.36	11.52	11.83
	64	4.69	4.79	13.95	13.57	6.17	6.48	13.76	13.74
	128	3.80	3.95	**16.31**	**16.43**	4.60	4.75	**15.70**	**15.94**
	2	2.40	2.67	3.99	3.86	8.82	9.38	8.08	7.96
	4	3.73	4.34	6.67	6.78	**8.95**	**9.73**	10.41	10.80
VB-NMF	8	4.95	5.53	9.78	9.65	8.70	9.42	12.23	12.71
	16	**5.39**	**5.85**	12.78	12.74	7.51	8.04	14.01	14.24
gamma	32	5.04	5.23	15.25	15.36	6.11	6.28	15.76	15.87
	64	4.12	4.24	16.94	16.88	4.68	4.85	17.32	17.53
	128	3.04	3.31	**17.27**	**17.60**	3.47	3.73	**18.57**	**18.91**
	2	2.38	2.71	4.17	4.11	8.25	8.96	7.89	7.91
	4	3.63	4.16	6.70	6.74	**8.68**	**9.38**	10.26	10.65
VB-NMF	8	4.66	5.22	9.41	9.29	8.50	9.22	11.93	12.34
	16	**5.20**	**5.64**	12.27	12.23	7.55	8.05	13.63	13.87
Jeffreys	32	4.95	5.27	14.43	14.50	6.30	6.56	15.20	15.27
	64	4.41	4.57	16.30	16.03	5.10	5.37	16.71	16.72
	128	3.59	3.89	**16.82**	**17.11**	4.06	4.38	**17.70**	**17.87**

TABLEAU 5.5 – Performance (SDR, ISR, SIR et SAR) individuelle des séparateurs en fonction du nombre de composantes, pour la NMF standard (*ML-NMF*) et la NMF bayésienne avec *a priori* gamma (*VB-NMF gamma*) et *a priori* Jeffreys (*VB-NMF Jeffreys*).

5.5.2 Apprentissage des paramètres de fusion adaptative

Outre les performances légèrement supérieures aux modèles ML-NMF, les modèles VB-NMF ici introduits nous permettent d'adapter nos coefficients de fusion à chaque mélange à séparer, grâce à l'énergie libre, approchant la vraisemblance marginale de chaque modèle. Nous avons dans ce chapitre proposé deux approches de fusion dédiées aux modèles VB-NMF. La première, nommée *fusion adaptative VB* et exposée dans la partie 5.2, est une application directe du principe de moyennage bayésien de modèles. La seconde est déduite d'un modèle génératif incluant le nombre de composantes comme variable aléatoire. Cette approche, contrairement à la première, ne respecte pas le cadre général de fusion introduit dans le chapitre 3 mais permet, tout comme la fusion adaptative VB, d'estimer une source comme le résultat de plusieurs NMFs à la manière du moyennage bayésien de modèles. À ce titre, nous y ferons référence par le terme *VB-NMF à ordre multiple*, ou comme précédemment par l'acronyme *mo-NMF*.

Afin de pouvoir comparer ces deux approches de fusion, nous nous plaçons ici dans le cas particulier déjà évoqué dans la partie 5.3.5. Ainsi, les M séparateurs que nous désirons fusionner ne diffèrent que par le nombre de composantes K_{1m} choisi pour modéliser la source de parole. De ce fait, nos deux approches de fusion deviennent comparables car les quantités $\mathcal{L}_m$ et π_m expriment alors respectivement l'énergie libre et la probabilité *a priori* de modèles équivalents. Rappelons toutefois que la fusion n'est rendue effective que par l'introduction d'un hyperparamètre, noté β, contrôlant l'entropie de la distribution *a posteriori* du nombre de composantes. Par conséquent, afin de procéder en pratique à la fusion adaptative selon les deux approches proposées dans ce chapitre, il nous faut avant toute chose déterminer les probabilités *a priori* π_m de chaque modèle et ce paramètre β. Pour ce faire, nous proposons dans cette partie d'apprendre leurs valeurs sur notre ensemble d'apprentissage, de façon similaire à la fusion statique par apprentissage introduite dans la partie 4.2.

Les M NMFs qui décrivent la source de parole étant estimées conjointement par le modèle VB-NMF à ordre multiple, nous ne pouvons envisager de baser notre apprentissage sur ce modèle. Par conséquent, nos données d'apprentissage seront obtenues par séparation préalable de notre ensemble d'apprentissage par les M modèles VB-NMF à ordre unique. Les valeurs de π_m et β ainsi apprises pourront être utilisées pour l'évaluation de nos méthodes sur les ensembles de validation et de test, aussi bien pour la fusion adaptative VB de NMFs à ordre unique que pour la fusion conjointe par NMF à ordre multiple. Nous supposons donc ici que notre ensemble d'apprentissage est composé de L mélanges $x^{(l)}(t)$ indexés par l et que pour chacun de ces exemples nous disposons :

— des J sources vraies $s_j^{(l)}(t)$ composant le mélange,

— des estimées $\tilde{s}_{jm}^{(l)}(t)$ de chacune de ces sources par chacun des M séparateurs,

— de l'énergie libre $\mathcal{L}_m^{(l)}$ estimée pour chacun des M séparateurs (ou des énergies libres $\mathcal{L}_{m,n}^{(l)}$ et $\mathcal{L}_{m,f}^{(l)}$ pour les cas variant en temps et variant en fréquence).

Comme pour la fusion statique, nous proposons alors d'apprendre les probabilités *a priori* de chaque modèle et le paramètre β soit par minimisation de l'EQM, soit par maximisation du SDR sur l'ensemble d'apprentissage, et ce pour les cas de fusions invariante, variant en temps et variant en fréquence.

Cas invariant

Pour rappel, dans le cas invariant, la fusion adaptative VB est donnée pour la $j^{\text{ième}}$ source et l'exemple l par :

$$\forall t, \quad \tilde{s}_j^{(l)}(t) = \frac{1}{\delta^{(l)}} \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_m^{(l)}/\beta} \, \tilde{s}_{jm}^{(l)}(t), \tag{5.112}$$

où $\delta^{(l)} = \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_m^{(l)}/\beta}$ représente la constante de normalisation pour l'exemple l.

Par minimisation de l'EQM La minimisation de l'erreur quadratique moyenne sur la base d'apprentissage relativement aux paramètres $\{\pi_m\}_{m=1..M}$ et β est obtenue par la résolution du problème suivant :

$$\begin{array}{ll} \underset{\{\pi_m\},\beta}{\operatorname{argmin}} & \sum_{l=1}^{L} \sum_t \left\| s_j^{(l)}(t) - \frac{1}{\delta^{(l)}} \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_m^{(l)}/\beta} \, \tilde{s}_{jm}^{(l)}(t) \right\|^2 \\ \text{avec} & \left\{ \begin{array}{l} \forall m, \, \pi_m \geq 0 \\ \beta \geq 1. \end{array} \right. \end{array} \tag{5.113}$$

En notant $\boldsymbol{\pi} = \{\pi_m\}_{m=1..M}$ le vecteur des probabilités *a priori* et $\boldsymbol{\mathcal{L}}^{(l)} = \left\{\mathcal{L}_m^{(l)}\right\}_{m=1..M}$ le vecteur composé des M énergies libres pour l'exemple l, le problème à résoudre peut être formulé sous la forme matricielle suivante :

$$\begin{array}{ll} \underset{\boldsymbol{\pi},\beta}{\operatorname{argmin}} & \sum_l \left[c^{(l)} + \frac{1}{\delta^{(l)\,2}} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right)^{\mathsf{T}} \widetilde{\mathbf{G}}^{(l)} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right) - \frac{2}{\delta^{(l)}} \tilde{\mathbf{d}}^{(l)\mathsf{T}} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right) \right] \\ \text{avec} & \left\{ \begin{array}{l} \forall m, \, \pi_m \geq 0 \\ \beta \geq 1, \end{array} \right. \end{array} \tag{5.114}$$

où les matrices de Gram $\widetilde{\mathbf{G}}^{(l)}$, les vecteurs $\tilde{\mathbf{d}}^{(l)}$ et les constantes $c^{(l)}$ sont définis comme aux équations (4.4), (4.5) et (4.6) pour la fusion statique invariante. Il est important de noter que ce problème n'est plus un simple problème PQ comme ce fut le cas pour la fusion statique. En effet, le terme $\delta^{(l)}$ dépendant des paramètres $\boldsymbol{\pi}$ et β à optimiser et l'introduction du paramètre β rendent le problème non-quadratique. De plus, étant donné que le terme d'énergie libre $\boldsymbol{\mathcal{L}}^{(l)}$ dépend de l'exemple l, il nous est impossible de simplifier plus encore la formulation de ce problème.

En pratique, nous résoudrons ce problème à l'aide de la fonction *fmincon* de *MATLAB*. Afin d'aider à la résolution, nous avons calculé et fourni le gradient de la fonction de coût dont l'expression est précisée en annexe D.3.

Par maximisation du SDR Maximiser le SDR moyen sur l'ensemble d'apprentissage revient à résoudre le problème suivant :

$$\begin{array}{ll} \underset{\{\pi_m\},\beta}{\operatorname{argmax}} & \sum_l 10 \log_{10} \dfrac{\sum_t \|\mathbf{s}_j^{(l)}(t)\|^2}{\sum_t \| \, \mathbf{s}_j^{(l)}(t) - \frac{1}{\delta^{(l)}} \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_m^{(l)}/\beta} \, \tilde{s}_{jm}^{(l)}(t) \, \|^2} \\ \text{avec} & \left\{ \begin{array}{l} \forall m, \, \pi_m \geq 0 \\ \beta \geq 1. \end{array} \right. \end{array} \tag{5.115}$$

En omettant le numérateur et en adoptant les notations matricielles introduites pour la minimisation de l'EQM, le problème à résoudre peut être formulé comme un problème de minimisation :

$$\begin{aligned} \underset{\boldsymbol{\pi},\beta}{\operatorname{argmin}} \quad & \sum_l \log_{10} \left[c^{(l)} + \tfrac{1}{\delta^{(l)\,2}} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right)^{\intercal} \widetilde{\mathbf{G}}^{(l)} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right) \right. \\ & \left. - \tfrac{2}{\delta^{(l)}} \tilde{\mathbf{d}}^{(l)\intercal} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right) \right] \\ \text{avec} \quad & \begin{cases} \forall m,\, \pi_m \geq 0 \\ \beta \geq 1, \end{cases} \end{aligned} \tag{5.116}$$

où les matrices de Gram $\widetilde{\mathbf{G}}^{(l)}$, les vecteurs $\tilde{\mathbf{d}}^{(l)}$ et les scalaires $c^{(l)}$ sont définis comme aux équations (4.4), (4.5) et (4.6). Comme pour la minimisation de l'EQM, l'optimisation sera menée grâce à la fonction *MATLAB fmincon* en fournissant l'expression du gradient de la fonction coût comme détaillée en annexe D.3.

Cas variant en fréquence

Dans le cas variant en fréquence, la fusion adaptative VB s'exprime pour chaque bande de fréquence f selon

$$\forall t, \quad \tilde{s}_j^{f\,(l)}(t) = \frac{1}{\delta_f^{(l)}} \sum_{m=1}^{M} \pi_{m,f}\, e^{\mathcal{L}_{m,f}^{(l)}/\beta_f}\, \tilde{s}_{jm}^{f\,(l)}(t), \tag{5.117}$$

où $\delta_f^{(l)} = \sum_{m=1}^{M} \pi_{m,f}\, e^{\mathcal{L}_{m,f}^{(l)}/\beta_f}$ représente la constante de normalisation pour l'exemple l et la bande de fréquence f. Nous faisons le choix ici de considérer un paramètre de contrôle de l'entropie différent pour chaque bande de fréquence f, d'où la notation β_f. Nous aurions tout à fait pu n'en considérer qu'un seul commun à toutes les bandes mais cela aurait nécessité que nous résolvions l'optimisation conjointement sur toutes les bandes de fréquence, faisant alors exploser la dimension du problème à résoudre, comme nous l'avions déjà noté dans la partie 4.2.2 pour la fusion statique variant en fréquence.

De ce fait, les problèmes à résoudre seront strictement identiques aux problèmes (5.114) et (5.116) exprimés dans le cas invariant. Ils seront simplement résolus indépendamment sur chacune des bandes de fréquence f. En pratique, nous utiliserons donc toujours la fonction *MATLAB fmincon*.

Par minimisation de l'EQM En notant $\boldsymbol{\pi}_f = \{\pi_{m,f}\}_{m=1..M}$ le vecteur des probabilités *a priori* et $\boldsymbol{\mathcal{L}}_f^{(l)} = \left\{\mathcal{L}_{m,f}^{(l)}\right\}_{m=1..M}$ le vecteur composé des M énergies libres pour l'exemple l et la bande de fréquence f, la minimisation de l'EQM sur chaque bande de fréquence s'écrit sous forme matricielle :

$$\begin{aligned} \underset{\boldsymbol{\pi}_f,\beta_f}{\operatorname{argmin}} \quad & \sum_l \left[c_f^{(l)} + \tfrac{1}{\delta_f^{(l)\,2}} \left(\boldsymbol{\pi}_f \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta_f}\right)^{\intercal} \widetilde{\mathbf{G}}_f^{(l)} \left(\boldsymbol{\pi}_f \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta_f}\right) \right. \\ & \left. - \tfrac{2}{\delta_f^{(l)}} \tilde{\mathbf{d}}_f^{(l)\intercal} \left(\boldsymbol{\pi}_f \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta_f}\right) \right] \\ \text{avec} \quad & \begin{cases} \forall m,\, \pi_{m,f} \geq 0 \\ \beta_f \geq 1, \end{cases} \end{aligned} \tag{5.118}$$

où les matrices de Gram $\widetilde{\mathbf{G}}_f^{(l)}$, les vecteurs $\tilde{\mathbf{d}}_f^{(l)}$ et les constantes $c_f^{(l)}$ sont définis comme aux équations (4.8), (4.9) et (4.10) pour la fusion statique invariante.

Par maximisation du SDR Maximiser le SDR moyen indépendamment sur chaque bande de fréquence f revient à résoudre le problème suivant sous forme matricielle :

$$\begin{aligned} \underset{\boldsymbol{\pi}_f, \beta_f}{\operatorname{argmin}} \quad & \sum_l \log_{10} \left[c_f^{(l)} + \tfrac{1}{{\delta_f^{(l)}}^2} \left(\boldsymbol{\pi}_f \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta_f} \right)^{\mathsf{T}} \widetilde{\mathbf{G}}_f^{(l)} \left(\boldsymbol{\pi}_f \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta_f} \right) \right. \\ & \left. - \tfrac{2}{\delta_f^{(l)}} \tilde{\mathbf{d}}_f^{(l)\mathsf{T}} \left(\boldsymbol{\pi}_f \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta_f} \right) \right] \\ \text{avec} \quad & \begin{cases} \forall m,\, \pi_{m,f} \geq 0 \\ \beta_f \geq 1, \end{cases} \end{aligned} \tag{5.119}$$

où les matrices de Gram $\widetilde{\mathbf{G}}_f^{(l)}$, les vecteurs $\tilde{\mathbf{d}}_f^{(l)}$ et les scalaires $c_f^{(l)}$ sont définis comme aux équations (4.8), (4.9) et (4.10).

Cas variant en temps

Dans le cas variant en temps, la fusion adaptative VB s'exprime pour chaque trame n selon

$$\forall t, \quad \tilde{s}_j^{n\,(l)}(t) = \frac{1}{\delta_n^{(l)}} \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_{m,n}^{(l)}/\beta} \, \tilde{s}_{jm}^{n\,(l)}(t), \tag{5.120}$$

où $\delta_n^{(l)} = \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_{m,n}^{(l)}/\beta}$ représente la constante de normalisation pour la trame n de l'exemple l. Dans la suite, nous noterons $N^{(l)}$ le nombre de trames de l'exemple l. Contrairement au cas variant en fréquence, il nous est ici impossible d'estimer une probabilité *a priori* différente pour chacune des trames n. Il en est de même pour le paramètre β. En pratique, l'apprentissage variant en temps revient donc à résoudre le même problème que pour le cas invariant mais plutôt que de minimiser l'EQM sur les L exemples de l'ensemble d'apprentissage, nous minimiserons l'EQM sur la totalité des trames de l'ensemble d'apprentissage, soit $\sum_l N^{(l)}$ trames. De même, plutôt que de maximiser le SDR moyen sur l'ensemble d'apprentissage, nous maximiserons le SDR moyen sur toutes les trames de l'ensemble d'apprentissage.

Par minimisation de l'EQM En notant $\boldsymbol{\pi} = \{\pi_m\}_{m=1..M}$ le vecteur des probabilités *a priori* et $\boldsymbol{\mathcal{L}}_n^{(l)} = \left\{\mathcal{L}_{m,n}^{(l)}\right\}_{m=1..M}$ le vecteur composé des M énergies libres pour la trame n de l'exemple l, la minimisation de l'EQM sur l'ensemble d'apprentissage s'écrit sous forme matricielle :

$$\begin{aligned} \underset{\boldsymbol{\pi}, \beta}{\operatorname{argmin}} \quad & \sum_{l,n} \left[c_n^{(l)} + \tfrac{1}{{\delta_n^{(l)}}^2} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_n^{(l)}/\beta} \right)^{\mathsf{T}} \widetilde{\mathbf{G}}_n^{(l)} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_n^{(l)}/\beta} \right) \right. \\ & \left. - \tfrac{2}{\delta_n^{(l)}} \tilde{\mathbf{d}}_n^{(l)\mathsf{T}} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_n^{(l)}/\beta} \right) \right] \\ \text{avec} \quad & \begin{cases} \forall m,\, \pi_{m,f} \geq 0 \\ \beta \geq 1, \end{cases} \end{aligned} \tag{5.121}$$

où les matrices de Gram $\widetilde{\mathbf{G}}_n^{(l)}$, les vecteurs $\tilde{\mathbf{d}}_n^{(l)}$ et les scalaires $c_n^{(l)}$ sont définis tels que

$$\forall l, \forall n, \quad \forall m_1, m_2, \quad \tilde{g}_{n,m_1m_2}^{(l)} = \sum_t \left\langle \tilde{\mathbf{s}}_{jm_1}^{n\,(l)}(t), \tilde{\mathbf{s}}_{jm_2}^{n\,(l)}(t) \right\rangle, \tag{5.122}$$

$$\forall m, \qquad \tilde{d}_{n,m}^{(l)} = \sum_t \left\langle \mathbf{s}_j^{n\,(l)}(t), \tilde{\mathbf{s}}_{jm}^{n\,(l)}(t) \right\rangle \tag{5.123}$$

$$\text{et} \qquad c_n^{(l)} = \sum_t \|\mathbf{s}_j^{n\,(l)}(t)\|^2. \tag{5.124}$$

On notera que, comparativement au problème invariant (5.114), la sommation sur l est remplacée par une sommation sur l et n.

Par maximisation du SDR Similairement, maximiser le SDR moyen des trames de l'ensemble d'apprentissage revient à résoudre le problème suivant :

$$\begin{aligned} \underset{\boldsymbol{\pi},\,\beta}{\operatorname{argmin}} \quad & \textstyle\sum_{l,n} \log_{10} \Bigg[c_n^{(l)} + \frac{1}{{\delta_n^{(l)}}^2} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_n^{(l)}/\beta} \right)^{\intercal} \widetilde{\mathbf{G}}_n^{(l)} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_f^{(l)}/\beta} \right) \\ & \qquad - \frac{2}{\delta_n^{(l)}} \tilde{\mathbf{d}}_n^{(l)\intercal} \left(\boldsymbol{\pi}_n \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}_n^{(l)}/\beta} \right) \Bigg] \\ \text{avec} \quad & \begin{cases} \forall m,\, \pi_m \geq 0 \\ \beta \geq 1, \end{cases} \end{aligned} \tag{5.125}$$

où les matrices de Gram $\widetilde{\mathbf{G}}_n^{(l)}$, les vecteurs $\tilde{\mathbf{d}}_n^{(l)}$ et les scalaires $c_n^{(l)}$ sont définis selon (5.122), (5.123) et (5.124).

5.5.3 Performances de fusion adaptative

Nous proposons dans cette partie d'évaluer les fusions adaptatives VB invariante, variant en fréquence et variant en temps sur la tâche de rehaussement de parole. Les paramètres de fusion (probabilités *a priori* π_m et paramètre de contrôle de l'entropie β) ont été appris sur l'ensemble d'apprentissage selon les méthodes exposées dans la partie 5.5.2 précédente. Les modèles retenus sont les modèles VB-NMF avec *a priori* gamma, pour les nombres de composantes $K_{1m} = 2^m$ et $m \in [1, 7]$, ceux-ci ayant, en moyenne, des SDRs individuels légèrement supérieurs aux modèles VB-NMF avec *a priori* de Jeffreys (voir partie 5.5.1). Les matrices de Gram $\widetilde{\mathbf{G}}^{(l)}$, les vecteurs $\tilde{\mathbf{d}}^{(l)}$ et les scalaires $c^{(l)}$, définis aux équations (4.4), (4.5) et (4.6) pour le cas invariant, (4.8), (4.9) et (4.10) pour le cas variant en fréquence et (5.122), (5.123) et (5.124) pour le cas variant en temps, ont donc été calculés pour les modèles VB-NMF considérés, et non pour les modèles ML-NMF comme ce fut le cas dans les chapitres 3 et 4.

Les paramètres ainsi appris ont également été utilisés pour évaluer les performances du modèle NMF à ordre multiple présenté dans la partie 5.3. Les paramètres de ce modèle ont été initialisés de la même manière que les paramètres des modèles NMF à ordre unique, selon la procédure présentée dans la partie 5.5.1. Nous avons également retenu l'*a priori* gamma pour ces expériences, comme défini à l'équation (5.6) avec $b = 0.2$.

Nous proposons en outre de comparer fusion adaptative VB, sélection VB et mo-NMF à la sélection VB et aux méthodes de fusion statique introduites dans le chapitre 4, en particulier, à la fusion statique invariante par apprentissage. Les résultats de sélection VB invariante ont été simplement obtenus en retenant pour chaque exemple le séparateur ayant l'énergie libre $\mathcal{L}_m$ la plus grande. De la même manière, la sélection VB variant en temps (respectivement, variant en fréquence) est obtenue en choisissant pour chaque trame n (respectivement, chaque bande de fréquence f) le séparateur ayant l'énergie libre $\mathcal{L}_{m,n}$ la plus grande (respectivement, $\mathcal{L}_{m,f}$). Pour la fusion statique invariante, l'apprentissage des coefficients a été mené sur le même ensemble d'apprentissage séparé à l'aide de modèles VB-NMF, et non ML-NMF comme ce fut le cas dans la partie 4.3.1.

Sélection VB et fusion VB

Les résultats obtenus par sélection VB, par fusion adaptative VB et par modèle mo-NMF sont représentés sur la figure 5.8 pour les cas de fusion invariante, variant en temps et variant en

fréquence. Dans un premier temps, quel que soit le cas de fusion adaptative VB, nous constatons, comme ce fut le cas pour la fusion statique dans le chapitre 4, que les performances globales de séparation données par le SDR sur la figure 5.8a sont sensiblement meilleures pour des paramètres appris par maximisation du SDR moyen sur l'ensemble d'apprentissage, selon le problème (5.116). En effet, la fusion adaptative VB invariante ainsi obtenue apporte un gain de 0.6 dB de SDR par rapport à l'apprentissage par minimisation de l'EQM selon (5.114). Dans les cas variant en fréquence et variant en temps, les gains sont légèrement plus faibles, à savoir 0.3 et 0.5 dB respectivement.

On constate également que, quelque soit la fonction de coût optimisée et le cas de fusion, la fusion VB donne toujours de meilleurs SDRs comparativement à la sélection VB. Ce résultat confirme ainsi les conclusions formulées sur les tests synthétiques de la partie 5.4.1 : l'énergie libre, qui pourtant est le critère bayésien de sélection de référence [Bishop, 2006], n'est pas corrélée à notre objectif de séparation. À titre de comparaison, la fusion statique par moyenne, introduite dans la partie 4.1, permet d'atteindre un SDR moyen sur les ensembles de validation et de test de 5.6 dB, soit un gain de 1.15 dB par rapport à la sélection VB invariante.

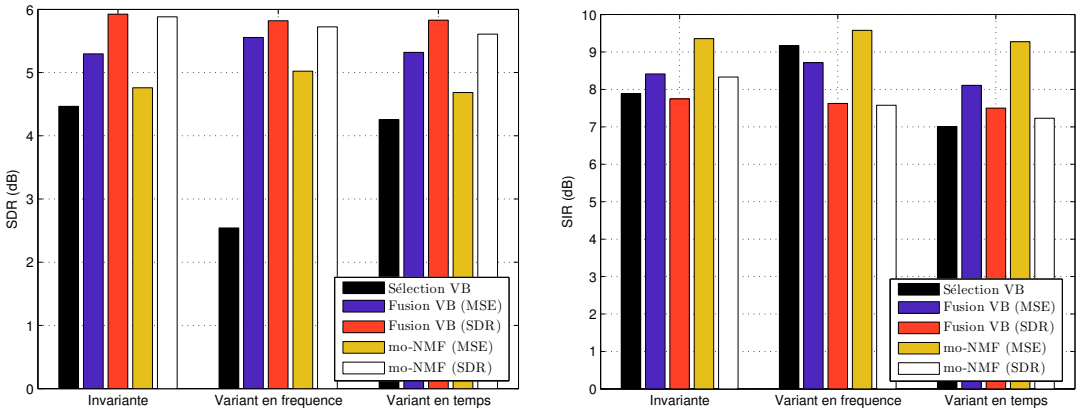
Par rapport à la fusion statique par moyenne, l'apport de la fusion adaptative VB est également à nuancer. La fusion adaptative VB permet, dans une certaine mesure, de pallier le défaut du critère de sélection et permet, dans le cas invariant par exemple, de gagner respectivement 0.8 dB et 1.5 dB de SDR en apprenant les probabilités *a priori* π_m et le paramètre β par minimisation de l'EQM et par maximisation du SDR. Toutefois, seule l'approche par maximisation du SDR fait mieux que la simple fusion statique par moyenne ($+0.3$ dB de SDR). De plus, nous constatons que, contrairement à ce que nous aurions pu espérer, les fusions adaptatives VB variant en temps et variant en fréquence donnent des résultats très similaires au cas invariant, suggérant que malgré le paramètre de contrôle de l'entropie β, le terme relatif à l'énergie libre $\mathcal{L}_m$ ne joue pas suffisamment son rôle d'adaptation des coefficients de fusion.

Fusion statique invariante et fusion VB invariante

À la vue des figures 5.8b et 5.8c représentant, entre autres, les SIR et SAR des méthodes de fusion adaptatives VB, nous pouvons confirmer que fusions adaptatives VB invariante, variant en fréquence et variant en temps donnent des résultats très similaires, tant en terme de SDR que de SIR et SAR. Le détail des valeurs numériques a été reporté dans le tableau 5.6. La partie basse du tableau fournit par ailleurs les résultats de fusion statique obtenus par apprentissage des coefficients de fusion selon (4.3) et (4.12). Nous constatons alors que les résultats de fusion adaptative VB sont également très similaires aux résultats de la fusion statique par apprentissage.

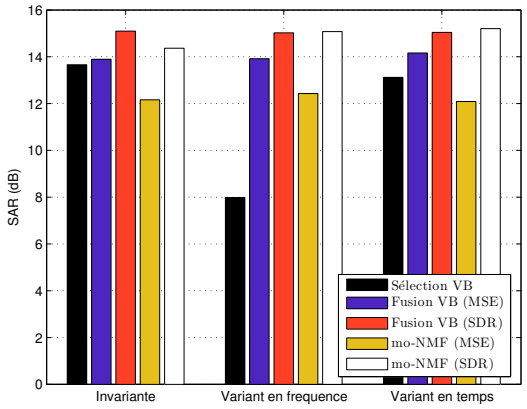
Ceci peut être logiquement expliqué lorsque l'on compare les probabilités *a priori* π_m et les coefficients de fusion statique invariante appris sur notre ensemble d'apprentissage. Leurs valeurs ont été tracées sur la figure 5.9. On peut alors remarquer que les valeurs sont très proches, quel que soit la fonction de coût optimisée (EQM ou SDR). Lorsque les valeurs apprises pour les probabilités *a priori* π_m et le paramètre β sont ensuite appliquées à un exemple, comme illustré sur la figure 5.10, nous constatons nettement que le terme de vraisemblance marginale de la forme $\exp(\mathcal{L}_m/\beta)$ représenté en rouge est proche d'une distribution uniforme et que la probabilité *a posteriori*, résultant du produit des probabilités *a priori* et de la vraisemblance marginale selon l'équation (5.110), est alors très proche de la probabilité *a priori* π_m apprise. On comprend alors que les valeurs de β apprises ($\beta = 1.8 \times 10^4$ dans le cas EQM et $\beta = 2.2 \times 10^4$ dans le cas SDR) ont tendance à uniformiser les termes de vraisemblance marginale, nous faisant ainsi perdre la potentielle adaptation des coefficients de fusion par le biais du terme d'énergie libre $\mathcal{L}_m$ et nous ramenant alors à une simple fusion statique invariante telle qu'étudiée dans le chapitre 4.

Nous avons mené de nombreuses autres expériences espérant améliorer ces résultats et tirer partie de l'information donnée par l'énergie libre. En particulier, conscients que les fonctions ob-



(a) SDR

(b) SIR



(c) SAR

FIGURE 5.8 – Performance (SDR, SIR, SAR) de la sélection VB, des fusions adaptatives VB et des modèles mo-NMF pour les cas de fusion invariante, variant en fréquence et variant en temps. Les résultats ont été moyennés sur les ensembles de validation et de test.

Fusion adaptative			SDR		ISR		SIR		SAR	
Cas	Type	App.	valid	test	valid	test	valid	test	valid	test
Invariant	Sélection VB		4.10	4.83	11.31	11.54	7.48	8.29	13.36	13.96
	Fusion VB	EQM	5.03	5.57	8.82	8.87	8.12	8.71	13.76	14.03
		SDR	**5.68**	6.17	12.02	12.11	7.51	7.99	14.95	15.25
	mo-NMF	EQM	4.47	5.05	8.66	8.49	8.99	9.73	11.95	12.37
		SDR	5.56	**6.20**	12.51	12.83	8.05	8.61	14.11	14.62
Variant en fréquence	Sélection VB		2.41	2.67	3.96	3.84	8.89	9.45	8.04	7.93
	Fusion VB	EQM	5.24	5.87	8.94	9.36	8.44	8.99	13.82	14.02
		SDR	**5.67**	**5.98**	13.02	13.27	7.49	7.76	14.96	15.09
	mo-NMF	EQM	4.57	5.47	8.74	9.26	9.27	9.89	12.22	12.64
		SDR	5.60	5.85	13.52	13.80	7.49	7.66	14.97	15.18
Variant en temps	Sélection VB		4.13	4.38	12.53	12.46	6.71	7.29	12.89	13.36
	Fusion VB	EQM	5.10	5.54	8.84	8.85	7.84	8.38	14.06	14.26
		SDR	**5.69**	**5.97**	13.90	13.74	7.34	7.66	14.96	15.13
	mo-NMF	EQM	4.41	4.96	8.52	8.37	8.91	9.64	11.90	12.28
		SDR	5.41	5.81	14.23	14.32	7.06	7.40	15.11	15.30
Fusion statique			SDR		ISR		SIR		SAR	
Cas		App.	valid	test	valid	test	valid	test	valid	test
Invariant		EQM	5.08	5.52	8.79	8.79	7.84	8.38	14.04	14.23
		SDR	**5.70**	**6.17**	11.97	12.04	7.47	7.94	15.00	15.29
	Par moyenne		5.43	5.80	11.33	11.43	6.45	6.91	16.23	16.50

TABLEAU 5.6 – Performance (SDR, ISR, SIR et SAR) de la fusion VB et du modèle mo-NMF pour les cas invariant, variant en fréquence et variant en temps.

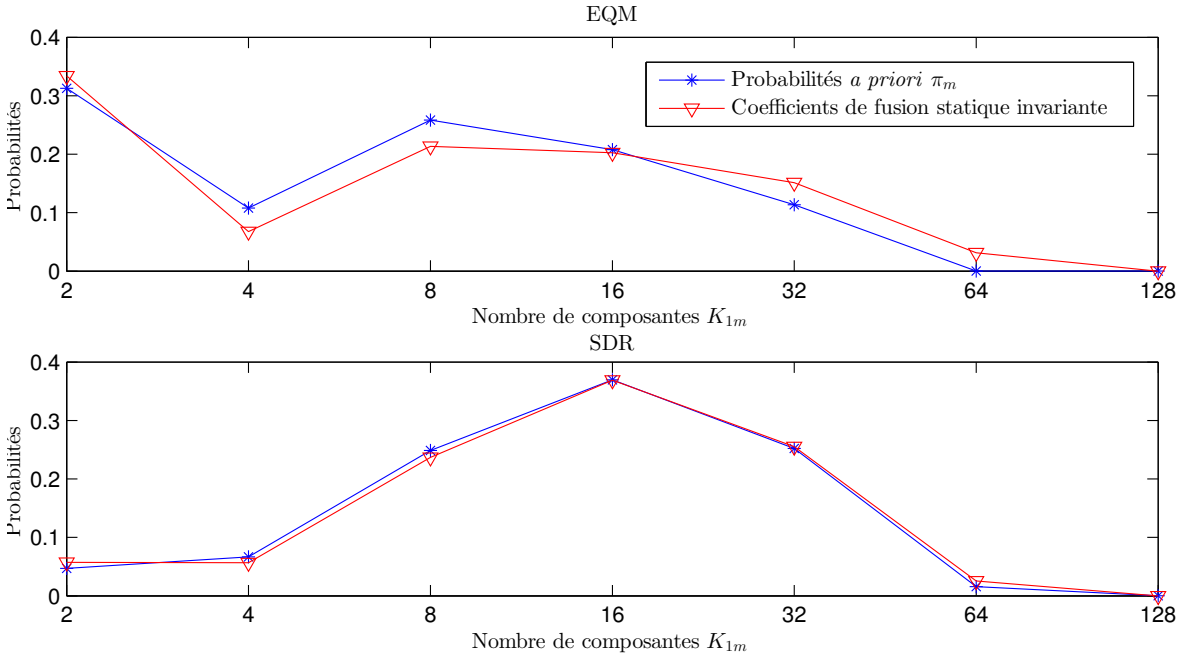


FIGURE 5.9 – Coefficients de fusion statique invariante et probabilités *a priori* π_m, appris par minimisation de l'EQM ou maximisation du SDR moyen sur l'ensemble d'apprentissage.

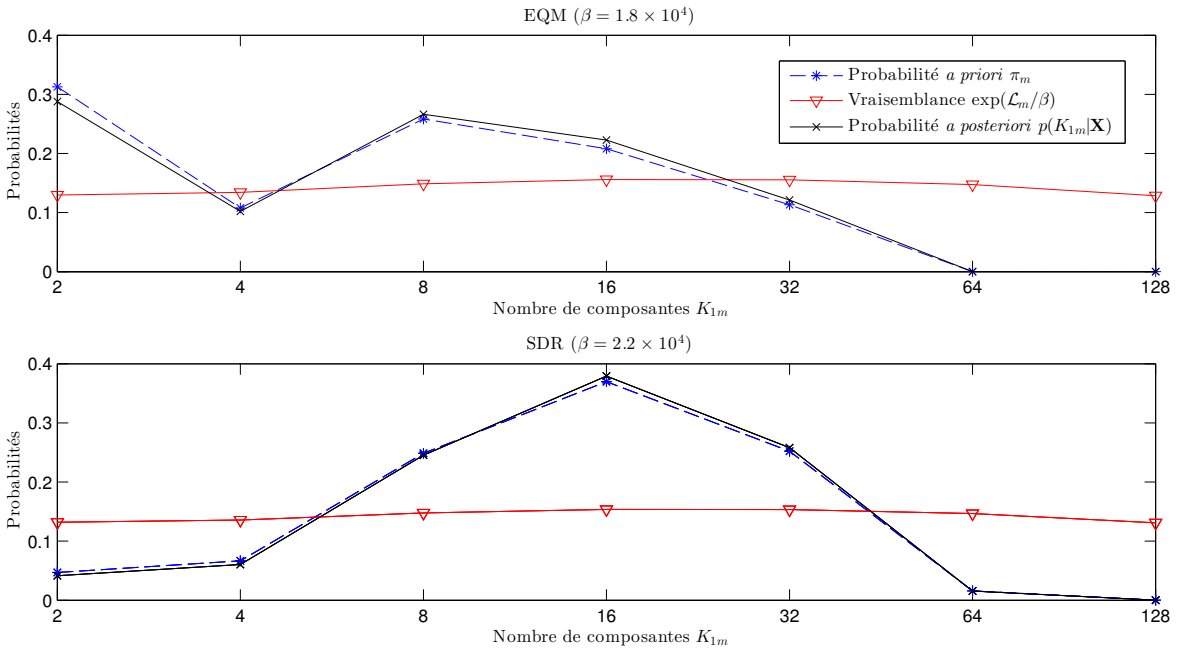


FIGURE 5.10 – Probabilités *a priori* π_m, vraisemblance marginale $\exp(\mathcal{L}_m/\beta)$ et probabilités *a posteriori* $p(K_{1m}|\mathbf{X})$ dans le cas invariant, pour l'exemple `0dB/s10_bbav2p.wav`.

jectifs (5.114) et (5.116) que nous cherchons à minimiser sont des fonctions difficiles à optimiser, avec potentiellement de nombreux minima locaux, nous avons tenté de réduire la complexité du problème en fixant le paramètre β à différentes valeurs et en optimisant alors les fonctions (5.114) et (5.116) par rapport aux probabilités *a priori* π_m uniquement. La figure 5.11, représentant les optima locaux ainsi obtenus en fonction de la valeur de β, nous indique alors très clairement que plus β est grand, plus le minimum local de l'EQM atteint est bas (respectivement, le maximum local du SDR atteint est haut). Ce faisant, nous constatons que le meilleur optimum local que nous sommes capables d'atteindre se trouve en $\beta \to +\infty$. Notant alors que $\forall m, \lim_{\beta \to +\infty} \exp(\mathcal{L}_m/\beta) = 1$, il devient évident que le meilleur minimum local que nous pouvons atteindre ramène notre fusion adaptative VB à une stricte équivalence avec la fusion statique par apprentissage introduite au chapitre 4, de telle sorte que $\forall m, \pi_m \equiv \alpha_m$. Toutefois, notons bien que ce résultat ne nous garantit pas pour autant que l'optimum global de nos fonctions de coût se trouve en $\beta \to +\infty$.

NMF à ordre multiple et fusion VB de NMFs à ordre unique

Comme nous l'avons évoqué en introduction, les paramètres π_m et β appris ont également été employés pour évaluer les performances du modèle NMF à ordre multiple. Les résultats obtenus sont donnés dans le tableau 5.6 ainsi qu'illustrés sur la figure 5.8. Nous constatons un comportement équivalent à la fusion adaptative VB, en ce que les paramètres appris par maximisation du SDR donnent de meilleures performances que les paramètres appris par minimisation de l'EQM. On notera que pour les paramètres appris par minimisation de l'EQM, le SDR moyen obtenu par le modèle mo-NMF est nettement inférieur au SDR de fusion adaptative VB, et ce, quel que soit le cas de fusion (invariant, variant en temps ou variant en fréquence). Toutefois, lorsque la fonction de coût retenue est le SDR, les SDRs obtenus par le modèle mo-NMF sont très proches (perte de -0.1 dB de SDR en moyenne, dont une perte de seulement -0.04 dB de SDR dans le cas invariant).

Comme pour la fusion adaptative VB, l'adaptation des coefficients de fusion au niveau de

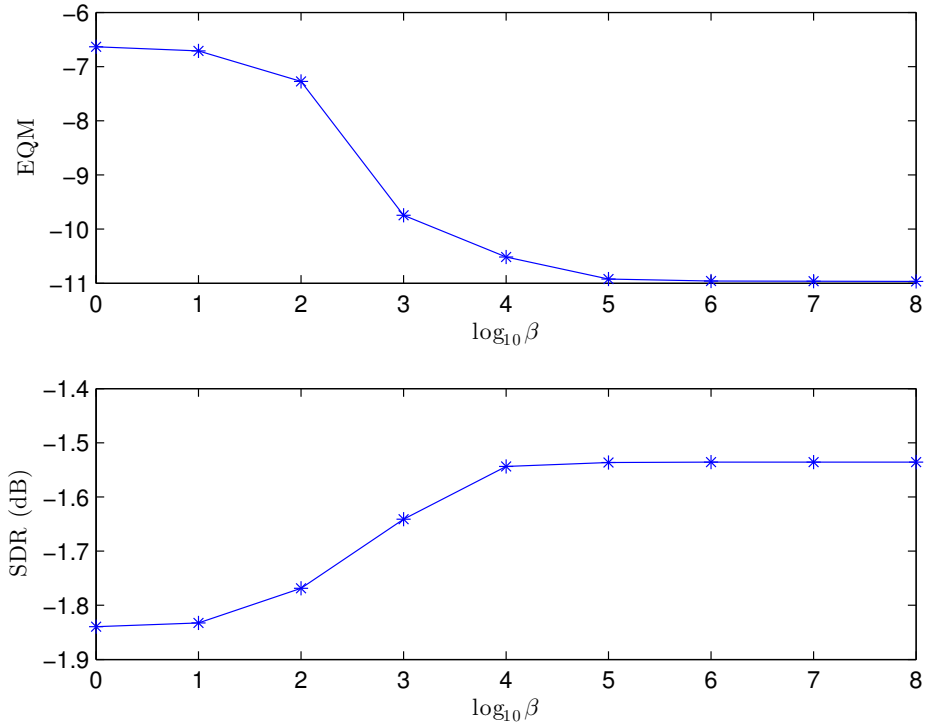


FIGURE 5.11 – Minima locaux de l'EQM et maxima locaux du SDR obtenus par optimisation par rapport aux probabilités *a priori* π_m et pour différentes valeurs de β.

la trame ou au niveau de la bande de fréquence ne permet pas d'améliorer les performances de fusion. On remarquera cependant que, dans le cas invariant avec apprentissage par maximisation du SDR, le SIR obtenu par le modèle mo-NMF est légèrement supérieur à celui obtenu par la fusion adaptative VB invariante (gain de 0.6 dB). En contrepartie, le SAR se trouve lui diminué de 0.7 dB.

Enfin, nous avons mesuré le temps de calcul moyen nécessaire pour la séparation et la fusion d'un exemple des ensembles de validation et de test. Pour la fusion adaptative VB, la séparation par les $M = 7$ modèles et l'opération de fusion demandent environ 54.4 s alors que la fusion conjointe par mo-NMF ne demande que 13.4 s. Si les résultats globaux de séparation sont équivalents, nos expériences sur le corpus CHiME montrent donc que le modèle mo-NMF permet d'améliorer significativement le temps de calcul moyen, en rendant la fusion approximativement quatre fois plus rapide.

5.6 Conclusion

Nous avons introduit dans ce chapitre la formulation de la NMF variationnelle bayésienne. Par application du principe de moyennage de modèles bayésien, nous avons pu identifier les coefficients de fusion introduits dans notre cadre général aux probabilités *a posteriori* des modèles à fusionner. La fusion ainsi définie devient adaptative par le biais du terme de vraisemblance marginale qui constitue un critère bayésien de sélection de modèles.

Toutefois, des expériences synthétiques préliminaires nous ont permis de montrer qu'en l'état, la fusion adaptative VB était inopérante. Plus en détail, nos expériences ont montré d'une part, que le critère de sélection basé sur la vraisemblance marginale ne permettait pas toujours de retrouver le modèle ayant généré les données, et d'autre part, qu'il n'était pas corrélé à notre objectif de séparation. Afin de pallier ce défaut et poursuivre notre objectif de fusion adaptative, nous avons

donc proposé l'introduction d'un paramètre de contrôle de l'entropie de la distribution *a posteriori* des modèles à fusionner. Sur nos données synthétiques, nous avons pu valider qu'il permettait de rendre la fusion effective et d'améliorer la qualité de séparation par rapport à la simple sélection VB.

Pour aller plus loin, nous avons également proposé un nouveau modèle génératif pour la NMF variationnelle bayésienne qui modélise le nombre de composantes comme une variable aléatoire, au même titre que les paramètres de la NMF. Nous avons montré que ce modèle, nommé *NMF à ordre multiple*, partageait d'évidentes similitudes avec le moyennage bayésien de modèles NMF à ordre unique. En effet, en considérant des modèles NMF à ordre unique ne variant que par le nombre de composantes de la source j à extraire, l'expression des coefficients de fusion adaptative VB associés se trouve être équivalente à la distribution *a posteriori* des nombres de composantes du modèle NMF à ordre multiple correspondant. Nos tests synthétiques ont alors montré que la NMF à ordre multiple obtenait des performances très similaires à celles de la fusion adaptative VB, pour un temps de calcul bien inférieur.

Finalement, l'évaluation de nos méthodes de fusion sur données réelles a montré les limites de l'approche bayésienne. En effet, si l'introduction du paramètre de contrôle de l'entropie et l'apprentissage des paramètres de fusion nous ont permis de dépasser les résultats obtenus par simple sélection VB, nous avons constaté que la fusion adaptative VB ainsi obtenue était équivalente, *in fine*, à la fusion statique par apprentissage étudiée au chapitre 4. Selon nos expériences, ce phénomène est principalement dû aux caractéristiques des fonctions de coût à optimiser pour déterminer les paramètres de fusion. Qu'il soit question de minimiser l'EQM ou de maximiser le SDR, les fonctions de coût se trouvent être complexes à optimiser, car non-convexes. Nos expériences ne nous ont ainsi permis que d'atteindre un optimum local en $\beta \to +\infty$, rendant alors à nouveau inopérant le terme relatif à la vraisemblance marginale supposé adapter les coefficients de fusion au signal considéré. Nous n'avons pour autant aucune garantie que l'optimum global des fonctions de coût se trouve en ce même point. Dès lors, une recherche approfondie portée sur les méthodes d'optimisation pourrait apporter une nette amélioration des résultats de fusion, au prix toutefois d'un éloignement certain du champ d'étude principal de cette thèse.

Sans rentrer dans ces considérations d'optimisation numérique, notre modèle NMF à ordre unique nous suggère également d'autres pistes de recherche. En effet, nous avons déjà noté que la fusion formulée par le modèle mo-NMF diffère dans sa forme de notre cadre général introduit dans le chapitre 3. En effet, plutôt que de fusionner les variables de sources $s_{jm,fn}$, le modèle mo-NMF fusionne selon l'équation (5.92) les termes $C_{jm,fn}$ définis à l'équation (5.89). En acceptant de sortir du cadre général de fusion proposé dans ce rapport, nous pourrions donc tout à fait envisager de formuler une nouvelle source fusionnée à partir de M modèles NMF à ordre unique, mais à la manière du modèle mo-NMF. En effet, le principe du moyennage bayésien de modèle défini à l'équation (5.61) peut s'appliquer à n'importe quelle grandeur dépendante des paramètres du modèle. De ce fait, rien ne nous empêche de calculer selon (5.92) le moyennage des termes $C_{jm,fn}$ calculés selon (5.37) pour différents modèles NMF à ordre unique indexés par m. La source fusionnée pourrait être alors simplement obtenue par estimation de la distribution $q^*(\mathbf{s}_{fn})$ en remplaçant dans la définition (5.44) de ses paramètres le terme $C_{j,fn}$ par sa version fusionnée

$$C_{j,fn} = \left(\sum_{m=1}^{M} p(\mathbf{K}_m|\mathbf{X})\, C_{jm,fn}^{-1} \right)^{-1} . \tag{5.126}$$

La fonction de coût qui en résulte pour l'apprentissage des probabilités *a priori* et du paramètre de contrôle de l'entropie se trouverait donc modifiée et il serait peut-être plus facile d'en trouver un bon optimum local.

Outre la difficulté des problèmes d'optimisation envisagés ici, il convient également de rappeler que plusieurs approximations se sont rendues indispensables à la formulation d'une solution analy-

tique de la NMF bayésienne. Ces approximations peuvent expliquer en partie le défaut du critère de maximum de vraisemblance marginale que nous avons observé. En effet, rappelons qu'en pratique, nous ne pouvons pas estimer la log-vraisemblance marginale $\log p(\mathbf{X}|\mathbf{K}_m)$ et nous remplaçons donc sa valeur par une borne inférieure incarnée par l'énergie libre $\mathcal{L}_m$. De surcroît, l'énergie libre n'ayant pas non plus d'expression analytique, elle doit elle aussi être approchée par une borne inférieure. Le critère théorique bayésien devient donc en pratique basé sur une double approximation de la vraisemblance marginale. Nous pouvons donc espérer qu'en recherchant une meilleure approximation de cette grandeur, nous pourrions également améliorer nos résultats de fusion adaptative bayésienne. Nous pourrions par exemple envisager l'*inférence variationnelle stochastique structurée* proposée récemment dans [Hoffman, 2014b] afin de restaurer certaines dépendances entre les paramètres du modèle. Sans cela, il semble que nos expériences de fusion se soient heurtées aux limites pratiques de la sélection bayésienne de modèles.

Chapitre 6

Fusion adaptative : approche déterministe par réseaux de neurones

Sommaire

La fusion adaptative VB introduite au chapitre 5 précédent nécessite que les séparateurs considérés pour la fusion disposent tous d'un modèle bayésien complet, ce qui est notamment le cas pour les modèles NMF que nous avons étudiés. Toutefois, un tel prérequis nous prive d'un bon nombre de méthodes de séparation proposées dans la littérature qui ne disposent pas de formulation probabiliste. De plus, nos expériences menées sur la tâche de rehaussement de la parole ont montré que l'approche bayésienne proposée laissait un gain potentiel important par rapport aux performances oracles étudiées dans le chapitre 3. Rappelons aussi que, dans les parties 3.3.6 et 3.4.4, nous avons pu constater que la fusion oracle variant en temps était la plus performante, en terme de SDR, et ce aussi bien sur la tâche de rehaussement de la parole que sur la tâche d'extraction de voix chantée. À ce titre, nous proposons dans ce dernier chapitre d'estimer des coefficients de fusion adaptative variant en temps de façon purement déterministe afin de pouvoir potentiellement envisager dans notre cadre de fusion tous les séparateurs non probabilistes disponibles dans la littérature.

Pour cela, nous allons proposer, comme nous l'avons déjà fait dans les chapitres précédents, d'apprendre ces coefficients de fusion variant en temps sur un ensemble d'apprentissage représentatif des mélanges à séparer. Notre objectif devient donc d'estimer les coefficients $\alpha_{m,n}$ pour une trame temporelle n donnée, à partir de la seule connaissance du mélange $\mathbf{x}^n(t)$ et des M estimées à fusionner $\tilde{\mathbf{s}}_{jm}^n(t)$. Traditionnellement, la résolution d'un tel problème est obtenue en deux étapes successives [Bishop, 2006]. La première étape consiste simplement à calculer des descripteurs des entrées disponibles. Dans un second temps, il convient alors de modéliser le lien entre ces descripteurs des entrées et la sortie désirée, ici le vecteur de coefficients de fusion oracle variant en temps $\{\alpha_{m,n}\}$.

Le choix des descripteurs d'entrée est souvent déterminant bien qu'il ne soit pas aisé tant le nombre de descripteurs proposés dans la littérature est important. Parmi les descripteurs les plus populaires en traitement audio, on citera par exemple les coefficients MFCC (pour *Mel-Frequency Cepstral Coefficients* en anglais) particulièrement usités en traitement de la parole, les *chroma* ou les coefficients LPC (pour *Linear Prediction Coefficients*) [Mathieu et al., 2010]. Pour la deuxième étape dite de modélisation, les modèles de mélange de gaussiennes (GMMs pour *Gaussian Mixture Models*) ont été souvent utilisés, notamment en reconnaissance de la parole [Rabiner et Juang, 1993].

Toutefois, très récemment, les performances des modèles GMM ont été mises à mal dans de nombreux domaines par l'avènement des réseaux de neurones profonds [Hinton et al., 2012a]. Leur principal avantage réside dans leur capacité à réaliser conjointement l'étape de description des entrées et l'étape de modélisation du lien entre entrées et sorties. C'est pourquoi nous proposons dans ce chapitre de recourir aux réseaux de neurones pour résoudre notre problème d'estimation de coefficients de fusion adaptative variant en temps.

Les réseaux de neurones profonds ont déjà été exploités pour la séparation de sources audio. On pourra citer par exemple la prédiction de masques oracles introduite dans [Maas et al., 2012; Narayanan et Wang, 2013] pour le rehaussement de la parole et dans [Huang et al., 2014b,c] pour l'extraction de voix chantée. Ces approches ont pour point commun d'appréhender la séparation de sources comme une tâche de classification. À l'inverse, ici, nous ne chercherons pas à modéliser l'étape de séparation de sources mais considèrerons plutôt les réseaux de neurones comme une méthode de fusion tardive basée sur un apprentissage, qui intervient postérieurement à l'étape de séparation de sources, à la manière du *stacking* dont nous avons introduit le principe dans la partie 2.3.2. Notre étude débutera dans la partie 6.1 par une brève présentation des réseaux de neurones, de leur principe aux récentes avancées qui en ont fait une technique aujourd'hui aussi populaire que performante. Nous présenterons ensuite dans la partie 6.2 la méthode d'apprentissage par réseaux de neurones que nous proposons pour estimer des coefficients de fusion variant en temps. Enfin, la partie 6.3 sera consacrée à l'évaluation de notre méthode sur les corpus *CHiME*

et *ccMixter*.

6.1 Apprentissage profond par réseaux de neurones

Comme son nom l'indique, un réseau de neurones est composé d'un bloc de base nommé *neurone* [Bishop, 1995]. Le modèle de neurone généralement admis a été proposé dans [Rosenblatt, 1958] sous le nom de *perceptron*. Le perceptron est en fait un simple classifieur dont le fonctionnement peut être représenté selon la figure 6.1. En effet, en réponse à une entrée $\mathbf{x} = [x_1, ..., x_d, ..., x_D]^\mathsf{T}$ de dimension D, le perceptron fournit en sortie une valeur binaire z valant 0 ou 1 et obtenue selon

$$z = h(a) = h(\mathbf{w}^\mathsf{T}\mathbf{x} + b), \tag{6.1}$$

où la quantité a, nommée *activité*, est obtenue en multipliant l'entrée par le vecteur $\mathbf{w} = [w_1, ..., w_D]^\mathsf{T}$ de dimension D, nommé vecteur de *poids*, et en y ajoutant le scalaire b nommé communément *biais*. Poids et biais étant réels, la fonction h dite d'*activation* permet d'obtenir une valeur binaire en sortie. Originellement, la fonction d'activation était une fonction à seuil, de type fonction de *Heaviside*, définie comme

$$z = h(a) = \begin{cases} 1 & \text{si} \quad a > 0 \\ 0 & \text{sinon} \end{cases} . \tag{6.2}$$

Toutefois, des fonctions d'activation continues, telles que la fonction *tangente hyperbolique* ($y = \tanh(a)$) ou la fonction *sigmoïde* ($y = \sigma(a) = (1 + \mathrm{e}^{-a})^{-1}$) ont par la suite été préférées pour leurs bonnes propriétés de dérivabilité. Quelle que soit la fonction choisie, il convient alors d'entraîner le modèle de neurone de manière supervisée afin d'apprendre ses paramètres, poids et biais.

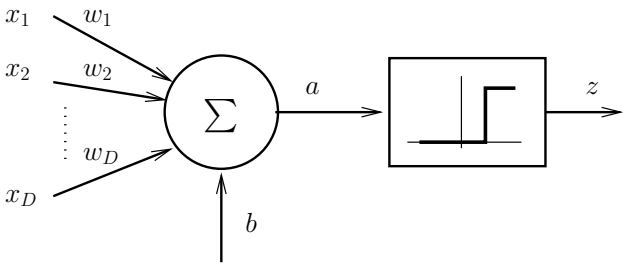
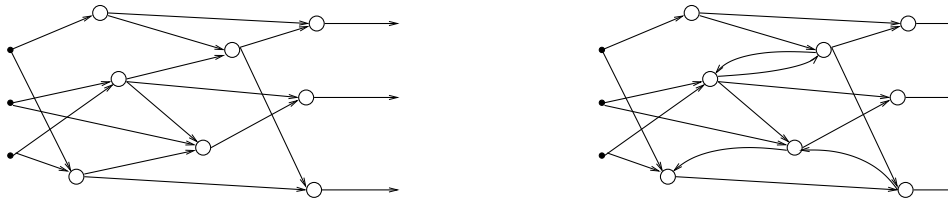


Figure 6.1 – Perceptron.

Pour aller du perceptron originel aux réseaux de neurones que nous allons ici exploiter, il suffit simplement d'organiser plusieurs neurones sous forme de réseaux connectés de sorte que les sorties de certains neurones composent les entrées d'un ou plusieurs autres. De ce principe élémentaire, nous pouvons d'ores et déjà envisager deux grandes familles de réseaux de neurones illustrées sur la figure 6.2 : les réseaux de neurones proactifs (*feedforward* en anglais) et les réseaux de neurones rétroactifs (*feedback* en anglais). Ces deux types de réseaux se distinguent par l'orientation des connexions liant les neurones entre eux : les premiers ne sont composés que de connexions unidirectionnelles, propageant les données de l'entrée vers la sortie du réseau uniquement (de la gauche vers la droite sur la figure 6.2a), contrairement aux seconds qui admettent aussi des connexions rétroactives propageant alors l'information de la sortie vers l'entrée (de la droite vers la gauche sur la figure 6.2b). Quelle que soit la famille considérée, les réseaux de neurones sont alors généralement organisés en couches successives, chaque couche étant composée d'un certain nombre de neurones. Ces réseaux de neurones *multicouches* peuvent être organisés selon différentes architectures définies entre autres par l'agencement des couches entre elles, les fonctions d'activation choisies ou encore la topologie des connexions entre neurones.



(a) Réseau proactif (b) Réseau rétroactif

FIGURE 6.2 – Exemples de réseaux proactif et rétroactif.

Pour notre problème de fusion, nous ne nous intéresserons ci-après qu'aux réseaux proactifs multicouches, et notamment au cas particulier du perceptron multicouche que nous présenterons brièvement dans la partie 6.1.1 suivante. Nous introduirons ensuite le principe de rétropropagation des erreurs très largement utilisé pour l'apprentissage des réseaux de neurones dans la partie 6.1.2. Dans la partie 6.1.3, nous présenterons les récentes avancées ayant permis l'avènement de l'apprentissage par réseaux de neurones profonds. Enfin, dans la partie 6.1.4, nous traiterons des techniques de régularisation permettant de limiter le phénomène de sur-apprentissage.

6.1.1 Perceptron multicouche

Le plus commun des réseaux de neurones proactifs est le *perceptron multicouche* (abrégé *MLP* pour *Multi-Layer Perceptron* en anglais), initialement proposé dans [RUMELHART et al., 1986]. Comme son nom l'indique, son architecture est organisée en C couches. Chaque couche c est composée de N_c neurones définis selon le modèle de neurone (6.1) illustré sur la figure 6.1. Le MLP est un réseau de neurones *complètement connecté*, ce qui signifie que chaque neurone de la couche c reçoit en entrée toutes les sorties de la couche $c-1$ précédente et les sorties des N_c neurones de la couche c forment les entrées de chacun des neurones de la couche $c+1$ suivante. Un MLP peut être représenté schématiquement comme sur la figure 6.3. Le MLP ainsi représenté n'est composé que d'une couche dite *cachée*, située au milieu de la figure, et d'une couche *de sortie* à droite, soit un total de $C = 2$ couches. Ce modèle peut être étendu à un nombre quelconque $C > 2$ de couches. La sortie du réseau, notée $\mathbf{y}$, ou $\mathbf{z}_C$ car correspondant à la dernière couche du réseau, est obtenue en appliquant d'abord les poids, biais et fonctions d'activation de chaque neurone de la première couche cachée à l'entrée $\mathbf{x}$ du réseau, puis en procédant de la même manière successivement pour les couches suivantes jusqu'à atteindre la couche de sortie. Par exemple, la sortie de la $c^{\text{ième}}$ couche cachée est obtenue selon :

$$\mathbf{z}_c = h_c\left(\mathbf{a}_c\right) = h_c\left(\mathbf{W}_c\mathbf{z}_{c-1} + \mathbf{b}_c\right) \tag{6.3}$$

où $\mathbf{z}_c = [z_{c,1}, ..., z_{c,n}, ..., z_{c,N_c}]^\mathsf{T}$ représente le vecteur composé des sorties des N_c neurones pour la couche c et $\mathbf{a}_c$ représente le vecteur des activités pour cette même couche. De la même manière, $\mathbf{b}_c$ est un vecteur composé des biais $b_{c,n}$ de chacun des N_c neurones de la couche c. La fonction h_c représente la fonction d'activation choisie pour la couche c, supposant que tous les neurones d'une même couche partagent la même fonction d'activation. Enfin, la matrice $\mathbf{W}_c$ de taille $N_c \times N_{c-1}$ est composée des N_c vecteurs de poids $\mathbf{w}_{c,n}$ relatifs à la couche c et qui en forment donc les lignes. Le vecteur $\mathbf{w}_{c,n}$, de longueur N_{c-1}, est lui relatif au $n^{\text{ième}}$ neurone de la couche c. En d'autres termes, la sortie du $n^{\text{ième}}$ neurone de la couche c peut s'écrire

$$z_{c,n} = h_c(a_{c,n}) \quad \text{avec} \quad a_{c,n} = \sum_{m=1}^{N_{c-1}} w_{c,nm} z_{c-1,m} + b_{c,n} \tag{6.4}$$

où $w_{c,nm}$ désigne le poids du $n^{\text{ième}}$ neurone de la couche c relatif à la $m^{\text{ième}}$ sortie de la couche précédente $c-1$.

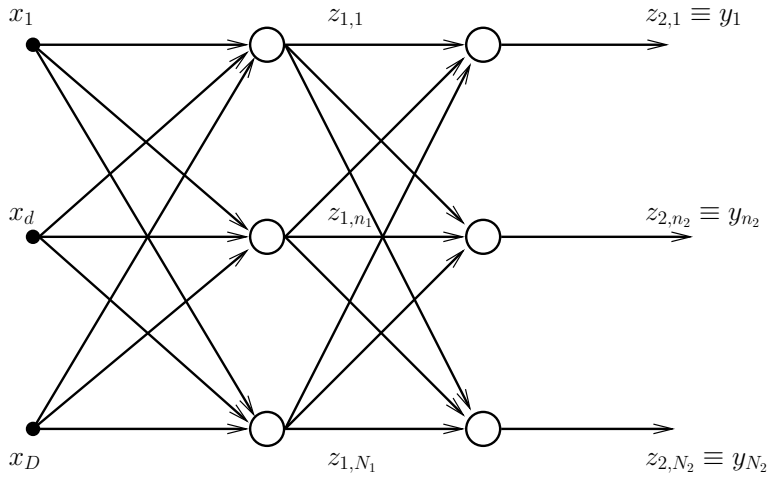


FIGURE 6.3 – Perceptron multicouche à une couche cachée.

Les fonctions d'activation, ici notées h_c, sont l'un des éléments clés d'un réseau de neurones puisqu'elles introduisent des non-linéarités dans l'expression de la sortie en fonction de l'entrée. Sans ces non-linéarités, il a été montré qu'un MLP est toujours équivalent à un réseau d'une seule couche. Comme nous l'avons déjà mentionné, plusieurs fonctions d'activation ont été proposées dans la littérature, telles que la fonction sigmoïde ou la fonction tangente hyperbolique. La fonction *softmax* est également souvent choisie, notamment pour la dernière couche. Cette dernière permet en effet de normaliser le vecteur de sortie $\mathbf{y} \equiv \mathbf{z}_C$ du réseau de sorte que chacun de ses éléments $z_{c,n}$ soit compris entre 0 et 1 et que la somme de ses éléments fasse 1. Ainsi, la sortie du $n^{\text{ième}}$ neurone pour la couche de sortie est donnée par :

$$y_n = z_{C,n} = \text{softmax}(a_{C,n}) = \frac{\mathrm{e}^{a_{C,n}}}{\sum_{n'=1}^{N_C} \mathrm{e}^{a_{C,n'}}} \tag{6.5}$$

où $a_{C,n} = \mathbf{w}_{C,n}^{\mathsf{T}} \mathbf{z}_{C-1} + b_{C,n}$ représente l'activité du neurone n de la couche C. Cette fonction d'activation est particulièrement utilisée dans les applications de classification multiclasse où la sortie du $n^{\text{ième}}$ neurone de la couche de sortie peut alors être interprétée comme la probabilité que l'exemple $\mathbf{x}$ présenté à l'entrée du réseau appartienne à la classe n, soit $z_{C,n} = p(\mathbf{x} \in \text{classe}\, n)$.

6.1.2 Apprentissage par rétropropagation des erreurs

Les paramètres d'un MLP, ou plus largement d'un réseau de neurones proactif, doivent être appris à partir d'un ensemble d'apprentissage. Généralement, cet apprentissage est mené de façon supervisée, c'est-à-dire que pour chaque exemple $\mathbf{x}_l$ de la base, nous connaissons la valeur $\mathbf{t}_l$ que nous souhaitons que prenne la sortie du réseau $\mathbf{y}(\mathbf{x}_l)$. De ce fait, l'apprentissage des paramètres du réseau, biais et poids, peut être mené itérativement de façon à minimiser une certaine fonction de coût φ. Généralement, φ est une fonction de la sortie du réseau $\mathbf{y}(\mathbf{x}_l)$ et de la sortie désirée $\mathbf{t}_l$, pour tout ou partie des exemples de l'ensemble d'apprentissage. De plus, pour un exemple l, l'erreur est souvent formée d'une somme de termes relatifs à chaque neurone de la couche de sortie. Ainsi, la fonction de coût peut généralement s'écrire

$$\varphi = \sum_l \sum_{n=1}^{N_c} e(y_n^{(l)}, t_n^{(l)}) \tag{6.6}$$

où e est une fonction mesurant l'erreur entre la sortie actuelle du $n^{\text{ième}}$ neurone de la couche de sortie, $y_n^{(l)}$, et la valeur cible désirée pour ce même neurone, $t_n^{(l)}$. Cette fonction pourra par exemple calculer la distance euclidienne entre ces deux termes.

La méthode la plus employée est alors basée sur le principe de *descente du gradient*. La procédure consiste dans un premier temps à initialiser aléatoirement les paramètres du réseau puis à les faire évoluer dans la direction opposée au gradient de la fonction de coût $\nabla\varphi$. À chaque itération τ, les paramètres, poids et biais, peuvent alors être mis à jour selon

$$\theta^{(\tau+1)} = \theta^{(\tau)} - \eta\,\nabla\varphi\left[\theta^{(\tau)}\right] \tag{6.7}$$

où θ symbolise un paramètre du réseau (soit un biais b_c, soit un poids $w_{c,nm}$), $\eta > 0$ représente le *taux d'apprentissage* et $\nabla\varphi\left[\theta^{(\tau)}\right]$ la valeur du gradient de la fonction de coût au point $\theta^{(\tau)}$ courant. Comme nous l'avons déjà évoqué, les fonctions de coût choisies pour l'apprentissage sont souvent exprimées comme une somme de termes φ_l, chacun étant relatif à un exemple l de l'ensemble d'apprentissage. Par conséquent, le gradient peut lui aussi être exprimé comme une somme de termes dépendant de l'exemple l seulement :

$$\forall\theta, \quad \nabla\varphi\left[\theta\right] = \sum_l \nabla\varphi_l\left[\theta\right]. \tag{6.8}$$

De par son architecture en couches, la descente de gradient pour l'apprentissage des paramètres d'un MLP prend alors une forme séquentielle particulière que nous allons détailler ci-après.

Nous nous intéressons ici à l'estimation du gradient de la fonction de coût pour l'exemple l, $\nabla\varphi_l\left[\theta\right]$. La valeur de la fonction de coût φ_l est obtenue en propageant l'exemple $\mathbf{x}_l$ afin de calculer la sortie correspondante du réseau $\mathbf{y}(\mathbf{x}_l)$ selon l'équation (6.3). Prenons pour exemple le calcul du gradient de la fonction de coût $\nabla\varphi\left[w_{c,nm}\right]$ par rapport au poids $w_{c,nm}$ du $n^{\text{ième}}$ neurone de la couche c. La composante du gradient relative à l'exemple l peut s'écrire

$$\nabla\varphi_l\left[w_{c,nm}\right] = \frac{\partial\varphi_l}{\partial w_{c,nm}}. \tag{6.9}$$

Dans un premier temps, il convient de noter que la fonction de coût φ ne dépend du poids $w_{c,nm}$ que par le biais de l'activité $a_{c,n}$ calculée pour l'exemple l courant. De ce fait, la composante du gradient relative à cet exemple peut être réécrite selon la règle de dérivation en chaîne :

$$\nabla\varphi_l\left[w_{c,nm}\right] = \frac{\partial\varphi_l}{\partial a_{c,n}}\frac{\partial a_{c,n}}{\partial w_{c,nm}}. \tag{6.10}$$

Notons que nous omettons volontairement l'indice l sur le terme d'activité $a_{c,n}$ afin de ne pas alourdir l'écriture. Il en sera de même pour les équations prochaines. Rappelant alors que $a_{c,n} = \mathbf{w}_{c,n}^{\mathsf{T}}\mathbf{z}_{c-1} + b_{c,n}$, il vient immédiatement que

$$\frac{\partial a_{c,n}}{\partial w_{c,nm}} = z_{c-1,m}, \tag{6.11}$$

où $z_{c-1,m}$ représente la sortie du neurone m de la couche précédente $c-1$, calculée pour l'exemple l. Ce terme est donc obtenu simplement en propageant l'exemple l à travers le réseau jusqu'à la couche $c-1$. Afin de mettre à jour le paramètre $w_{c,nm}$, il ne reste donc qu'à calculer le terme $\delta_{c,n} \equiv \frac{\partial\varphi_l}{\partial a_{c,n}}$ dans l'expression du gradient (6.10). Pour cela, ce terme peut être réécrit selon la règle de dérivation en chaîne en fonction des activités des neurones de la couche suivante $c+1$:

$$\delta_{c,n} = \frac{\partial\varphi_l}{\partial a_{c,n}} = \sum_{m=1}^{N_{c+1}} \frac{\partial\varphi_l}{\partial a_{c+1,m}}\frac{\partial a_{c+1,m}}{\partial a_{c,n}}. \tag{6.12}$$

En injectant la définition de $\delta_{c,n}$ dans l'équation (6.12) précédente et en utilisant l'expression de la sortie d'un neurone définie à l'équation (6.4), nous obtenons finalement la formule suivante dite de *rétropropagation* :

$$\delta_{c,n} = h'_c(a_{c,n}) \sum_{m=1}^{N_{c+1}} w_{c+1,mn}\delta_{c+1,m} \tag{6.13}$$

où $h'_c(a_{c,n})$ correspond à la valeur de la dérivée de la fonction d'activation h_c calculée au point $a_{c,n}$. Le terme de rétropropagation provient du fait que pour calculer les quantités $\delta_{c,n}$ pour la couche c, il faut au préalable calculer les quantités $\delta_{c+1,n}$ correspondantes pour la couche $c+1$ suivante.

Ainsi, l'apprentissage se réalise en deux temps principaux et peut être représenté schématiquement selon la figure 6.4. Dans un premier temps (figure 6.4a), il convient de *propager* un exemple $\mathbf{x}$ dans le réseau afin de calculer les sorties $\mathbf{z}_c$ de chaque couche et la sortie du réseau $\mathbf{y_c}$. Cette première étape de propagation permet également de calculer la valeur de la fonction de coût pour cet exemple (figure 6.4b). Dans un second temps (figures 6.4c et 6.4d), il s'agit de calculer la dérivée de la fonction de coût par rapport aux paramètres de chacune des couches, en commençant par la couche de sortie (figure 6.4c) jusqu'à la première couche cachée (figure 6.4d), par *rétropropagation* des quantités $\boldsymbol{\delta}_c$ de couche en couche. Au final, chaque paramètre du réseau pourra être mis à jour selon la règle suivante en fonction de ces deux quantités selon :

$$w_{c,nm}^{(\tau+1)} = w_{c,nm}^{(\tau)} - \eta\, \delta_{c,n} z_{c-1,m}. \tag{6.14}$$

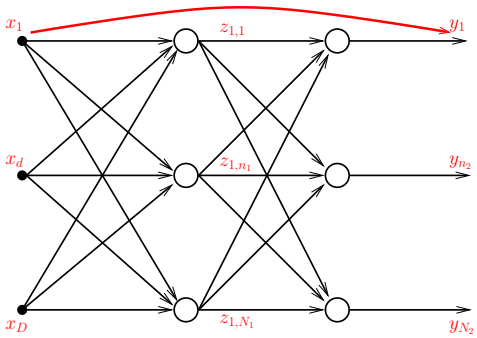
Le calcul sera mené de façon identique pour les biais de chaque couche.

Afin de mener un bon apprentissage, il convient généralement de disposer d'un ensemble d'apprentissage conséquent. De ce fait, le recours à une descente de gradient classique par rétropropagation où à chaque itération l'intégralité de l'ensemble d'apprentissage est présentée au réseau peut engendrer des temps d'apprentissage excessifs. Pour y remédier, il est proposé de présenter à chaque itération un sous-ensemble $\mathcal{B}$ d'exemples seulement. La méthode employée se trouve alors être une *descente de gradient stochastique* [Bishop, 2006; Duda et al., 2012] et l'expression du gradient à chaque itération prend alors la forme

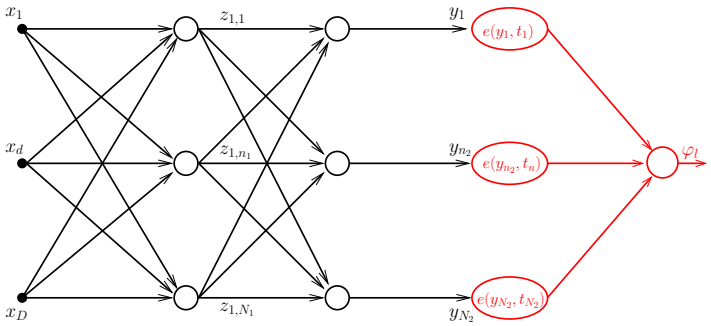
$$\forall\theta, \quad \nabla\varphi\,[\theta] = \sum_{l\in\mathcal{B}} \nabla\varphi_l\,[\theta]\,. \tag{6.15}$$

La descente de gradient stochastique se trouve être un cas intermédiaire entre descente de gradient classique, où $\mathcal{B}$ est l'ensemble complet, et descente de gradient *on-line*, où $\mathcal{B}$ contient une seule trame temporelle.

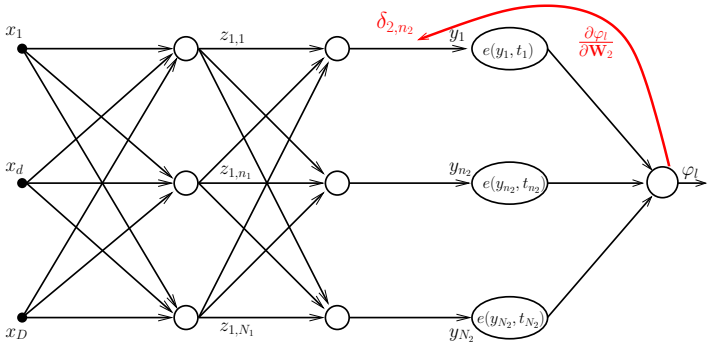
Dans le vocabulaire relatif aux réseaux de neurones, l'ensemble d'exemples $\mathcal{B}$ présenté à chaque itération est souvent appelé *mini-batch*. Au cours de l'apprentissage, chaque exemple est donc présenté plusieurs fois au réseau, et pas forcément au sein d'un même mini-batch. Il convient toutefois de présenter la totalité de l'ensemble d'apprentissage. En pratique, les mini-batchs sont donc formés à chaque itération de sorte que les exemples sélectionnés n'aient pas encore été présentés au réseau. Ce processus est répété jusqu'à ce que l'intégralité de l'ensemble d'apprentissage ait été présenté au réseau. La procédure peut alors continuer en présentant à nouveau les exemples de l'ensemble d'apprentissage sous forme de mini-batches, l'ensemble ayant été préalablement mélangé aléatoirement afin que les exemples ne soient pas présentés dans le même ordre. On comptera alors une *iteration* lorsqu'un mini-batch est présenté au réseau pendant la phase d'apprentissage et une *époque* lorsque l'intégralité de l'ensemble d'apprentissage a été présenté (autrement dit, quand le nombre d'itérations nécessaires pour présenter l'intégralité de l'ensemble d'apprentissage a été atteint).



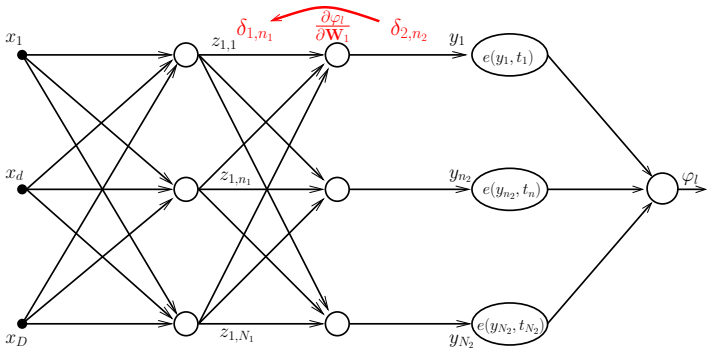
(a) Propagation d'un exemple.



(b) Calcul de la fonction de coût $\varphi_l = \sum_{n_2=1}^{N_2} e(y_{n_2}, t_{n_2})$.



(c) Calcul du gradient de la fonction de coût par rapport aux paramètres de la couche de sortie $\frac{\partial \varphi_l}{\partial w_{n_2 n_1}} = \delta_{2,n_2} z_{1,n_1}$.



(d) Rétropropagation des quantités δ_{2,n_2} pour le calcul du gradient de la fonction de coût par rapport aux paramètres de la couche cachée avec $\delta_{1,n_1} = h_1'(a_{1,n_1}) \sum_{n_2=1}^{N_2} w_{2,n_2 n_1} \delta_{2,n_2}$.

FIGURE 6.4 – Principe de la rétropropagation des erreurs.

6.1.3 Apprentissage profond

Bien que le principe de réseau de neurones multicouche ait été proposé dès la fin des années 1980 [LeCun et al., 1989; Rumelhart et al., 1986], la difficulté et le temps de calcul nécessaire pour l'apprentissage de réseaux ayant plus de deux couches cachées ont longtemps freiné l'essor de ces modèles en apprentissage automatique, au profit de modèles plus simples tels que les machines à vecteurs supports. Toutefois, des avancées récentes [Hinton et al., 2006; Vincent et al., 2010b] ont rendu accessible l'apprentissage de réseaux de neurones dits *profonds*. Les DNNs (pour *Deep Neural networks* en anglais) se sont alors montrés très compétitifs dans de nombreux domaines, notamment en reconnaissance automatique de la parole [Hinton et al., 2012a]. De plus, l'essor des méthodes d'apprentissage pour DNNs s'est accompagné par des progrès essentiels en terme de puissance de calcul, progrès principalement portés par l'avènement du calcul sur processeur graphique (*GPU* pour *Graphics Processing Unit* en anglais).

D'un point de vue algorithmique, la véritable découverte a été publiée dans [Hinton et al., 2006] et consiste en une étape de pré-entraînement non supervisé du réseau, préalable à l'apprentissage supervisé tel que présenté dans la partie 6.1.2. Dans [Hinton et al., 2006], il est en effet proposé de pré-entraîner chaque couche d'un réseau proactif à l'aide d'un type particulier de réseau de neurones rétroactif, les machines de Boltzmann restreintes (*RBM* pour *Restricted Boltzmann Machine* en anglais). Sans rentrer dans le détail des RBMs, le principe consiste simplement à d'abord pré-entraîner de façon non-supervisée la première couche cachée du réseau de neurones en considérant qu'elle compose la couche cachée d'une RBM. Une fois cette première couche pré-entraînée, la seconde couche cachée peut être pré-entraînée de la même manière. La seconde couche cachée forme alors la couche cachée d'une nouvelle RBM dont les entrées sont formées des sorties de la première couche cachée déjà pré-entraînée. La procédure peut être ainsi répétée jusqu'à la dernière couche cachée. La couche de sortie est finalement initialisée aléatoirement et l'apprentissage classique par rétropropagation peut alors être mené comme exposé dans la partie 6.1.2 afin d'apprendre les paramètres de la couche de sortie et d'affiner les valeurs des paramètres des couches cachées pré-entraînées. Le même principe a plus tard été proposé dans [Bengio, 2009; Vincent et al., 2010b]. La seule différence réside dans le modèle utilisé pour le pré-entraînement non-supervisé : il s'agit ici d'utiliser un autoencodeur plutôt qu'une RBM.

Cette proposition originale de pré-entraînement couche par couche a véritablement insufflé un nouvel élan à la recherche sur les réseaux de neurones et de nombreuses autres contributions importantes ont suivi. En particulier, la technique dite de *dropout* a été proposée dans [Hinton et al., 2012b]. Elle consiste à bruiter l'apprentissage en désactivant aléatoirement à chaque itération une partie des neurones du réseau. La méthode est similaire aux principes de *bagging* et de moyennage de modèles que nous avons déjà évoqués dans le chapitre 2. De la même manière, la méthode dite de *dropconnect* introduite dans [Wan et al., 2013] propose plutôt de désactiver au cours de l'apprentissage un certain nombre de poids en forçant leurs valeurs à zéro. Ces deux méthodes ont montré qu'elles pouvaient contribuer à améliorer la capacité de généralisation des réseaux de neurones profonds, c'est-à-dire leur capacité à classifier correctement un exemple ne faisant pas partie de l'ensemble d'apprentissage.

Très récemment, les performances des réseaux de neurones ont été rendues encore plus compétitives grâce à l'introduction de nouvelles fonctions d'activation, remplaçant les traditionnelles fonctions tangente hyperbolique et sigmoïde. En particulier, [Glorot et al., 2011; Nair et Hinton, 2010] ont introduit la fonction de *correction* (*rectifier* en anglais) définie par

$$h(a) = \max(0, a) = \begin{cases} a & \text{si} \quad a > 0 \\ 0 & \text{sinon.} \end{cases} \tag{6.16}$$

Dans la littérature, les neurones composés d'une telle fonction d'activation sont alors nommés *rectified linear units* (*ReLUs*). Les réseaux composés de ReLUs ont notamment montré de meilleures

caractéristiques de convergence, tant en terme de rapidité que de capacité de généralisation [Zeiler et al., 2013]. De surcroît, les ReLUs permettent de s'affranchir de l'étape de pré-apprentissage.

Enfin, dernièrement, la fonction d'activation dite *maxout* proposée dans [Goodfellow et al., 2013] a rencontré un engouement certain [Cai et al., 2013; Miao et al., 2013; Swietojanski et al., 2014; Zhang et al., 2014]. La fonction *maxout* est une généralisation de la fonction de *correction* définie à l'équation (6.16). Le principe est simple. Il consiste à envisager pour chaque neurone d'un réseau plusieurs ensembles de paramètres $\{\mathbf{w}_j, b_j\}$ plutôt qu'un seul. Supposant par exemple que nous envisageons J ensembles de paramètres, la fonction *maxout* retourne l'activité maximale obtenue parmi les J activités calculées selon la définition de l'équation (6.1). Formellement, la fonction *maxout* peut s'écrire

$$h(a_1, ..., a_j, ..., a_J) = \max_{j \in \{1,...,J\}} a_j \tag{6.17}$$

où a_j est l'activité calculée pour les paramètres indexés par j, soit $a_j = \mathbf{w}_j^\intercal \mathbf{z}_{-1} + b_j$ où $\mathbf{z}_{-1}$ représente ici l'entrée du neurone considéré (soit la sortie de la couche précédente). La fonction *maxout* est particulièrement performante lorsqu'elle est combinée à la technique de *dropout*. Selon [Goodfellow et al., 2013], elle permet aussi l'apprentissage de réseaux plus profonds que la fonction de *correction*, sans pour autant requérir une étape de pré-apprentissage.

6.1.4 Sur-apprentissage

En dépit de ces récentes avancées, il est un phénomène bien connu du monde de l'apprentissage automatique qu'il convient toujours de surveiller : le sur-apprentissage. Pour un problème donné, il est très probable qu'un réseau avec beaucoup de paramètres parvienne à bien modéliser les données d'apprentissage mais modélise moins bien d'autres données non présentées pendant l'apprentissage. On parle alors de sur-apprentissage. À l'inverse, lorsqu'un réseau permet de bien modéliser des données non utilisées pour l'apprentissage, on dira du réseau qu'il a une bonne capacité à *généraliser*.

Le phénomène de sur-apprentissage est intimement lié au dimensionnement du réseau de neurones, autrement dit, à sa *complexité*. Le nombre de couches cachées et de neurones par couche cachée définissent le nombre total de paramètres du réseau, biais et poids, et reflètent donc cette complexité. Comme pour les modèles de NMF que nous avons étudiés plus tôt, il existe donc un réseau optimal étant donné le problème posé exprimant un bon compromis entre adéquation aux données et complexité. Pour contrôler la complexité d'un réseau, traditionnellement, des méthodes de *régularisation* sont employées. Le principe de régularisation cherche généralement à pénaliser cette complexité. Dans le cas des réseaux de neurones, une méthode simple consiste à favoriser les petites valeurs de poids et biais. En effet, il a été démontré dans [Geman et al., 1992] qu'un réseau ayant des poids de valeur absolue élevée avait plus tendance au sur-apprentissage qu'un même réseau ayant des poids de plus faible valeur absolue. Pratiquement, il s'agit d'ajouter un terme de pénalité à la fonction de coût qui devient alors :

$$\tilde{\varphi}(\boldsymbol{\theta}) = \varphi(\boldsymbol{\theta}) + \frac{\lambda}{2}\|\boldsymbol{\theta}\|^2 \tag{6.18}$$

où $\lambda > 0$ est un paramètre à choisir et $\boldsymbol{\theta}$ symbolise l'ensemble des paramètres du réseau, poids et biais. Cette méthode est nommée *méthode de dégradation des pondérations* (*weight decay* en anglais).

Une autre méthode permettant d'éviter le sur-apprentissage consiste simplement à contrôler l'évolution de l'erreur sur un ensemble de données distinct de l'ensemble d'apprentissage. Cet ensemble est généralement nommé *ensemble de validation*. Si la descente de gradient employée pour mettre à jour les paramètres du réseau fait toujours diminuer la valeur de la fonction de

coût évaluée sur l'ensemble d'apprentissage (tout du moins, dans le cas de la descente de gradient classique), la fonction de coût évaluée sur un ensemble de validation distinct ne suit généralement cette tendance qu'en début d'apprentissage. En effet, cette dernière augmente dès lors que le réseau commence à sur-apprendre. Cette technique, très populaire et couramment nommée *validation croisée*, consiste alors à stopper l'apprentissage lorsque la fonction de coût sur l'ensemble de validation commence à augmenter. De ce fait, le terme *early stopping* est très employé dans la littérature relative aux réseaux de neurones.

Pour aller plus loin, il est également possible d'adapter le taux d'apprentissage η introduit dans l'équation (6.7) relativement à la valeur de la fonction de coût évaluée sur l'ensemble de validation, que nous nommerons ci-après *erreur de validation* par souci de concision. Une telle stratégie a notamment été proposée dans [Duffner et Garcia, 2007] pour la descente de gradient *on-line*. Il s'agit à chaque itération τ de calculer dans un premier temps :

— la variation relative de l'erreur de validation :

$$\Delta^{(\tau)} = \frac{\varphi^{(\tau)} - \varphi^{(\tau-1)}}{\varphi^{(\tau)}},$$

— ainsi que sa moyenne glissante :

$$\bar{\Delta}^{(\tau)} = \zeta\, \Delta^{(\tau)} + (1-\zeta)\, \bar{\Delta}^{(\tau-1)}$$

où ζ est un paramètre à choisir contrôlant le poids des itérations passées dans l'expression de la moyenne.

Le taux d'apprentissage η est alors mis à jour selon :

$$\eta^{(\tau)} = \begin{cases} d\,\eta^{(\tau-1)} & \text{si } \Delta^{(\tau)}\,\bar{\Delta}^{(\tau-1)} < 0 \quad \text{et} \quad |\bar{\Delta}^{(\tau-1)}| > \vartheta, \\ u\,\eta^{(\tau-1)} & \text{sinon,} \end{cases} \tag{6.19}$$

avec $u > 1$ et $d < 1$ deux paramètres réels à choisir permettant respectivement d'augmenter et de diminuer la valeur du taux d'apprentissage. Ainsi, si le signe de la variation de l'erreur de validation Δ change entre l'itération courante τ et l'itération précédente $\tau - 1$, alors le taux d'apprentissage est diminué d'un facteur d. Sinon, il est augmenté d'un facteur u. De petites variations peuvent être tolérées sans pour autant diminuer le taux d'apprentissage, grâce à l'introduction de la grandeur $\vartheta > 0$. Notons bien que le premier changement de signe interviendra probablement en début de sur-apprentissage, en accord avec le principe d'*early stopping*. Une telle procédure est semblable à la méthode *Delta-Bar-Delta* introduite dans [Jacobs, 1988]. Cette dernière nécessite toutefois le calcul des dérivées secondes de la fonction de coût. On citera aussi à titre d'exemple les méthodes *AdaGrad* [Zeiler, 2012] et *ADADELTA* [Duchi et al., 2011] qui proposent d'autres procédures alternatives d'adaptation du taux d'apprentissage.

En plus de la règle d'adaptation du taux d'apprentissage (6.19), [Duffner et Garcia, 2007] propose également une étape de raffinement lorsqu'un minimum de l'erreur de validation est atteint. Si un minimum de l'erreur de validation est ainsi atteint à l'itération τ_{min}, plutôt que de stopper net l'apprentissage, il est conseillé de continuer l'apprentissage pendant quelques itérations afin de vérifier que l'erreur de validation ne diminue pas de nouveau. Si au bout de ces quelques itérations, l'erreur de validation n'a pas de nouveau diminué, il est alors suggéré de revenir à l'itération $\tau_{\text{min}} - 1$, de diminuer le taux d'apprentissage et d'observer pendant quelques itérations supplémentaires si l'erreur de validation ne diminue pas de nouveau. En pratique, on notera que cette stratégie nécessite d'implémenter la sauvegarde de l'état du réseau aux itérations τ_{min} et $\tau_{\text{min}} - 1$.

Finalement, nous noterons que pour évaluer l'erreur de validation, rien ne nous empêche d'utiliser une autre fonction que la fonction de coût choisie pour l'apprentissage. En particulier, nous proposerons dans nos expériences de la partie 6.3 une erreur de validation liée à notre objectif de séparation, à savoir le SDR.

6.2 Réseau pour l'estimation des coefficients de fusion

Nous avons introduit dans la partie précédente les grands principes et avancées qui ont fait la popularité des réseaux de neurones pour résoudre des problèmes de regression et de classification. Nous proposons dans cette partie de tirer partie de leur puissance afin de mettre en place un MLP, potentiellement profond, pour estimer des coefficients de fusion variant en temps à partir de la seule connaissance des mélanges $\mathbf{x}^n(t)$ et des M estimées $\tilde{\mathbf{s}}_{jm}^n(t)$ de la source à fusionner. À la lumière du problème posé, nous décrirons les spécifications des couches d'entrée et de sortie dans les parties 6.2.2 et 6.2.1 respectivement. La partie 6.2.3 apportera quelques précisions sur l'architecture générale et l'agencement des couches cachées. Enfin, la partie 6.2.4 détaillera les fonctions de coût proposées pour l'apprentissage et qui seront exploitées dans la partie expérimentale 6.3.

6.2.1 Sortie du réseau

Nous cherchons ici à déterminer des coefficients de fusion variant en temps $\alpha_{m,n}$ tels que définis dans la partie 3.2.1, à l'équation (3.11). La sortie de notre réseau sera donc nécessairement composée de M neurones, un par coefficient de fusion. De plus, comme nous l'avons introduit dans la partie 3.1, nos coefficients de fusion doivent respecter deux contraintes : quelle que soit la trame n, les M coefficients doivent être positifs ou nuls et se sommer à un. Afin de respecter ces contraintes, la fonction *softmax* introduite à l'équation (6.5) sera choisie comme unique fonction d'activation de la couche de sortie de notre réseau.

6.2.2 Entrée du réseau

Si la dimension de la sortie du réseau reste raisonnable, la dimension de l'entrée du réseau doit être nécessairement plus grande. Pour rappel, nous avons à notre disposition pour mener l'inférence des coefficients de fusion la seule connaissance du mélange à séparer $\mathbf{x}^n(t)$ ainsi que celle des M estimées de la source j à fusionner, $\tilde{\mathbf{s}}_{jm}^n(t)$. De ce fait, nous proposons que l'entrée de notre réseau soit composée des spectres de log-puissance QERB de ces signaux. Ce choix semble cohérent avec la littérature de reconnaissance de la parole puisqu'il a été montré dans [Li et al., 2012] que les *spectres de log-puissance en échelle Mel* (c'est-à-dire, obtenus par filtrage en bandes Mel du spectrogramme de puissance) donnaient de meilleurs performances de reconnaissance que les MFCCs ou les spectres de log-puissance ou log-amplitude. Ce type de représentation a été également retenu dans [Huang et al., 2014b; Narayanan et Wang, 2013; Weninger et al., 2014] pour l'estimation de masques temps-fréquence idéaux. La seule différence par rapport à la littérature existante repose donc sur le choix de l'échelle ERB plutôt que de l'échelle Mel, ces deux échelles étant toutefois très proches. Toutefois, il a été montré dans [Weninger et al., 2014] que le choix d'une représentation temps-fréquence complète type TFCT en entrée était préférable dans le cas où la compression de l'échelle de fréquence par emploi de l'échelle Mel était trop forte (c'est-à-dire lorsque le nombre de bandes est petit). Par conséquent, nous tâcherons de garder un nombre de bandes de fréquence ERB suffisamment grand pour éviter cet écueil.

Dans le domaine connexe de la reconnaissance de parole, les réseaux de neurones prennent généralement en entrée une trame n, ou un descripteur de cette trame, ainsi que les trames adjacentes afin de tirer partie du contexte de la trame centrale [Seltzer et al., 2013]. De la même manière, nous proposons ici de composer notre entrée des spectres relatifs à la trame n pour laquelle nous souhaitons estimer les coefficients de fusion, ainsi que des spectres relatifs aux c trames précédentes et aux c trames suivantes.

Pour résumer, nous proposons que l'entrée relative à la trame n prenne la forme matricielle

suivante :

$$\begin{bmatrix} \log|\mathbf{x}_{n-c}|^2 & \cdots & \log|\mathbf{x}_{n}|^2 & \cdots & \log|\mathbf{x}_{n+c}|^2 \\ \log|\tilde{\mathbf{s}}_{j1,n-c}|^2 & \cdots & \log|\tilde{\mathbf{s}}_{j1,n}|^2 & \cdots & \log|\tilde{\mathbf{s}}_{j1,n+c}|^2 \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ \log|\tilde{\mathbf{s}}_{jm,n-c}|^2 & \cdots & \log|\tilde{\mathbf{s}}_{jm,n}|^2 & \cdots & \log|\tilde{\mathbf{s}}_{jm,n+c}|^2 \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ \log|\tilde{\mathbf{s}}_{jM,n-c}|^2 & \cdots & \log|\tilde{\mathbf{s}}_{jM,n}|^2 & \cdots & \log|\tilde{\mathbf{s}}_{jM,n+c}|^2 \end{bmatrix} \tag{6.20}$$

où la première ligne est composée des spectres de log-puissance du mélange avec $|\mathbf{x}_n|^2 = \left[|\mathbf{x}_{1n}|^2, ..., |\mathbf{x}_{fn}|^2, ..., |\mathbf{x}_{Fn}|^2\right]$ et les M lignes suivantes correspondent aux spectres de log-puissance des M sources estimées avec $|\tilde{\mathbf{s}}_{jm,n}|^2 = \left[|\tilde{\mathbf{s}}_{jm,1n}|^2, ..., |\tilde{\mathbf{s}}_{jm,fn}|^2, ..., |\tilde{\mathbf{s}}_{jm,Fn}|^2\right]$. Comme nous l'avons déjà exploité dans nos expériences menées sur le corpus *CHiME*, les spectrogrammes de puissance considérés sont monocanaux. Lorsque le signal observé est multicanal, le spectrogramme monocanal peut alors être simplement obtenu en sommant les spectrogrammes de chacun de ses canaux. Chaque trame temporelle est donc représentée par un unique vecteur de longueur F, où F représente le nombre de bandes de fréquence calculées par transformation temps-fréquence. Par conséquent, le vecteur d'entrée du réseau, obtenu par concaténation des lignes de la matrice (6.20), est de longueur $F(2c+1)(M+1)$.

En fonction de la taille du contexte c et du nombre de bandes de fréquence F, la dimension de l'entrée pourrait être très élevée. De ce fait, il se peut qu'un bon apprentissage ne soit accessible qu'au prix d'un réseau de grande taille et d'un ensemble d'apprentissage de taille également importante. En effet, il a été montré dans [Ellis et Morgan, 1999] que les réseaux avec beaucoup de paramètres requièrent généralement un ensemble d'apprentissage de grande taille. Par ailleurs, il est probable que l'entrée constituée selon la matrice (6.20) présente certaines redondances, par exemple entre trames adjacentes ou entre sources estimées par différents séparateurs. Pour ces raisons, il peut s'avérer judicieux de réduire la dimension de l'entrée, de façon à limiter les redondances et diminuer la taille du réseau. L'analyse en composantes principales (*PCA* pour *Principal Component Analysis* en anglais) est notamment une méthode de réduction de dimension très populaire, permettant de ne retenir que les composantes pertinentes de données de très grande dimension [Bishop, 2006]. Pour nos expériences, nous emploierons cette méthode de réduction car elle présente l'avantage de permettre de contrôler la quantité de variance conservée après réduction de la dimension de l'ensemble d'apprentissage, grâce au taux cumulé de variance expliquée [Bishop, 2006]. Nous en étudierons les conséquences dans la partie expérimentale 6.3.

Enfin, la mise à l'échelle préalable des données d'apprentissage peut également avoir une influence certaine sur la performance d'un réseau [Duda et al., 2012]. Pour éviter ce problème, il est conseillé de normaliser les données d'apprentissage de sorte que la distribution de chacune des composantes de l'entrée sur l'ensemble d'apprentissage soit de moyenne nulle et de variance unité. Par exemple, la composante relative au spectre de log-puissance du mélange pour la trame n de l'exemple l sera normalisée comme suit :

$$\forall f \in [1, F], \quad \frac{\log|\mathbf{x}_{fn}^{(l)}|^2 - \mu_{x,f}}{\sigma_{x,f}} \tag{6.21}$$

où $\mu_{x,f}$ et $\sigma_{x,f}$ désignent respectivement la moyenne et l'écart-type des termes $\log|\mathbf{x}_{fn}^{(l)}|^2$ sur l'ensemble d'apprentissage, pour chaque bande de fréquence f. Dans nos expériences de la partie 6.3, nous normaliserons ainsi les données d'apprentissage avant et après réduction par PCA. De la même manière, les exemples des ensembles de validation et de test devront être normalisés par la moyenne et l'écart-type de l'ensemble d'apprentissage.

6.2.3 Architecture

Si les dimensions des couches d'entrée et de sortie sont en partie définies par le problème formulé, le nombre, la taille et la topologie des couches cachées sont autant de paramètres qui peuvent influer sur la qualité de l'apprentissage. Afin de réduire le nombre de paramètres dont nous étudierons l'influence, nous n'étudierons pas ici l'influence du choix de la fonction d'activation. Par conséquent, nous ne considérerons ici que des fonctions d'activation de type fonctions de *correction* (ReLUs) que nous avons déjà définies à l'équation (6.16), afin notamment de nous affranchir de l'étape de pré-entraînement non-supervisé. Rappelons que, comme nous l'avons précisé dans la partie 6.2.1, la couche de sortie est elle dotée d'une fonction d'activation *softmax*.

Pour les autres paramètres, nous proposerons dans la partie 6.3 de faire varier le nombre de couches cachées de 1 à 4 et pour chaque couche cachée, de faire varier le nombre de neurones par multiples entiers de la taille de la couche de sortie M.

6.2.4 Fonctions de coût

Le choix de la fonction de coût pour l'apprentissage supervisé peut être un élément déterminant pour atteindre des performances satisfaisantes. Dans cette partie, nous introduisons quatre fonctions de coût distinctes dont nous étudierons les performances dans la partie expérimentale 6.3. Les deux premières fonctions introduites sont les fonctions traditionnellement utilisées en classification, respectant la forme générale introduite à l'équation (6.6). Les deux autres fonctions sont elles liées à notre objectif de séparation de sources et ne respectent pas cette même forme générale. Toutes les quatre seront toutefois exprimées comme la somme de termes φ_n relatifs à chaque trame selon

$$\varphi = \sum_{n \in \mathcal{B}} \varphi_n. \tag{6.22}$$

Afin de simplifier les notations, nous avons volontairement omis ici l'exposant (l) indexant l'exemple considéré, en ce sens que chaque trame n de chaque exemple l de l'ensemble d'apprentissage constitue un exemple d'apprentissage.

Fonctions classiques

Dans les applications classiques de classification, la fonction de coût est choisie afin de minimiser une certaine distance entre la sortie courante du réseau $\mathbf{y}$ et la sortie souhaitée $\mathbf{t}$ pour ce même exemple. Pour notre problème, la sortie du réseau est identifiée au vecteur de coefficients de fusion variant en temps $\tilde{\boldsymbol{\alpha}}_n = \{\tilde{\alpha}_{m,n}\}$. Par comparaison aux problèmes de classification, la valeur cible peut elle être identifiée au vecteur de coefficients de fusion oracle $\boldsymbol{\alpha}_n = \{\alpha_{m,n}\}$ obtenus par la résolution du problème QP (3.19).

Le choix le plus immédiat consiste alors à minimiser l'erreur quadratique moyenne (EQM) entre le vecteur de coefficients estimés et le vecteur de coefficients de fusion oracle. La fonction de coût s'écrit alors pour la trame n

$$\varphi_n^{EQMo} = \sum_{m=1}^{M} \left(\alpha_{m,n} - \tilde{\alpha}_{m,n}\right)^2. \tag{6.23}$$

Dans la suite de ce rapport, nous nous référerons à cette fonction de coût par le terme de *fonction d'EQM oracle (EQMo)*.

Un autre choix possible pour la fonction de coût consiste à considérer la fonction d'entropie croisée (EC) généralisée aux problèmes multiclasses non-binaires, utilisée notamment pour l'apprentissage des auto-encodeurs [Socher et al., 2011]. Pour notre problème, cette fonction de

coût s'écrit

$$\varphi_n^{EC} = -\sum_{m=1}^{M} \alpha_{m,n} \log \frac{\tilde{\alpha}_{m,n}}{\alpha_{m,n}}. \tag{6.24}$$

On notera que son expression est équivalente à la divergence de KL entre $\alpha_{m,n}$ et $\tilde{\alpha}_{m,n}$, comme définie dans la partie 2.1.3. Dans la suite, nous nommerons cette fonction de coût *fonction d'entropie croisée*. Notons bien que la fonction d'EQM oracle et la fonction d'entropie croisée nécessitent toutes deux de connaître les coefficients de fusion oracle de chaque trame de l'ensemble d'apprentissage.

Fonctions liées à l'objectif de séparation

L'estimation des coefficients de fusion oracle sur l'ensemble d'apprentissage nécessite une étape préalable à l'apprentissage du réseau. Cette étape supplémentaire, coûteuse en temps de calcul, peut être en partie évitée en utilisant une fonction de coût ne dépendant pas de la connaissance de ces coefficients de fusion oracle. En s'inspirant des problèmes d'optimisation formulés dans le chapitre 4 pour l'apprentissage de coefficients de fusion statique et le chapitre 5 pour l'apprentissage de coefficients de fusion adaptative VB, nous proposons ci-après deux autres fonctions de coût que nous emploierons dans la partie expérimentale 6.3 pour l'apprentissage du réseau.

Dans la partie 4.2.1, nous proposions d'apprendre des coefficients de fusion statique invariante par minimisation de l'erreur quadratique moyenne sur l'ensemble d'apprentissage. De la même manière, nous proposons ici d'apprendre les paramètres de notre réseau de neurones par minimisation de l'erreur quadratique moyenne entre la trame n de la source vraie $\mathbf{s}_j^n(t)$ et l'estimée fusionnée de la trame correspondante $\tilde{\mathbf{s}}_j^n(t) = \sum_{m=1}^{M} \tilde{\alpha}_{m,n} \tilde{\mathbf{s}}_{jm}^n(t)$, où $\tilde{\alpha}_{m,n}$ représentent les coefficients de fusion estimés par le réseau. La fonction de coût peut alors s'écrire sous la forme matricielle suivante :

$$\varphi_n^{\text{EQMs}} = c_n + \tilde{\boldsymbol{\alpha}}_n^{\intercal} \widetilde{\mathbf{G}}_n \tilde{\boldsymbol{\alpha}}_n - 2\, \tilde{\mathbf{d}}_n^{\intercal} \tilde{\boldsymbol{\alpha}}_n \tag{6.25}$$

avec

$$\begin{aligned} \forall n, \quad \forall m_1, m_2, \quad & \tilde{g}_{n,m_1m_2} = \textstyle\sum_t \left\langle \tilde{\mathbf{s}}_{jm_1}^n(t), \tilde{\mathbf{s}}_{jm_2}^n(t) \right\rangle, \\ \forall m, \quad & \tilde{d}_{n,m} = \textstyle\sum_t \left\langle \mathbf{s}_j^n(t), \tilde{\mathbf{s}}_{jm}^n(t) \right\rangle \\ \text{et} \quad & c_n = \textstyle\sum_t \|\mathbf{s}_j^n(t)\|^2. \end{aligned} \tag{6.26}$$

Rappelons que nous omettons ici volontairement l'exposant (l) indexant l'exemple considéré. En effet, contrairement au cas statique invariant exposé au chapitre 4, l'EQM est calculée pour chaque trame n de chaque exemple l, et non globalement sur chaque exemple l. Pour un même exemple, la fonction de coût aura donc une valeur différente à chaque trame n, et l'exposant (l) devient en ce sens redondant avec l'indice de trame. Dans la suite, nous nommerons cette fonction de coût *fonction d'EQM source (EQMs)* afin de la distinguer de la fonction d'EQM oracle introduite par l'équation (6.23).

Comme nous l'avons fait pour les fusion statique et adaptative VB, nous proposons alternativement d'apprendre notre réseau de neurones de façon à maximiser le SDR sur l'ensemble d'apprentissage. La fonction de coût s'exprime alors simplement comme

$$\varphi_n^{\text{SDR}} = 10 \log_{10} \left(c_n + \tilde{\boldsymbol{\alpha}}_n^T \widetilde{\mathbf{G}}_n \tilde{\boldsymbol{\alpha}}_n - 2\, \tilde{\mathbf{d}}_n^T \tilde{\boldsymbol{\alpha}}_n \right) \tag{6.27}$$

où les composantes $\widetilde{\mathbf{G}}_n$, $\tilde{\mathbf{d}}_n$ et c_n sont définies comme à l'équation (6.26). Dans la suite, nous nommerons simplement cette fonction de coût *fonction SDR*.

Contrairement aux fonctions de coût (6.23) et (6.24), les fonctions de coût (6.25) et (6.27) ici définies ne requièrent pas le calcul préalable des coefficients de fusion oracle. De surcroît,

nous noterons que les fonctions de coût ainsi définies sont directement liées à notre objectif de séparation, en ce qu'elles dépendent toutes deux de la source vraie $\mathbf{s}_j^n(t)$ que nous cherchons à séparer. Nous noterons aussi que d'autres fonctions de coût pourrait aussi être envisagées, comme le SAR, le SIR ou éventuellement une combinaison de ces mesures. De telles fonctions, même si elles ne respectent pas la forme générale introduite à l'équation (6.6), permettent tout de même d'appliquer le principe de rétropropagation pour opérer la descente de gradient et mettre à jour les paramètres du réseau, puisqu'elles dépendent de la sortie du réseau par le biais de la source fusionnée $\tilde{\mathbf{s}}_j^n(t) = \sum_{m=1}^{M} \tilde{\alpha}_{m,n} \tilde{\mathbf{s}}_{jm}^n(t)$. Cet apprentissage reste par ailleurs supervisé de par la connaissance indispensable de la source vraie $\mathbf{s}_j^n(t)$ pour le calcul de la fonction de coût.

Enfin, au cours de l'apprentissage du réseau, il est à noter que la fonction de coût va être évaluée de nombreuses fois. Une évaluation trop coûteuse en temps sera donc pénalisante. Nous noterons donc que les fonctions de coût EQMs (6.25) et SDR (6.27) peuvent être évaluées aussi rapidement que les fonctions de coût classiques EQMo (6.23) et EC (6.24), en calculant préalablement les grandeurs $\widetilde{\mathbf{G}}_n$, $\tilde{\mathbf{d}}_n$ et c_n pour toutes les trames de l'ensemble d'apprentissage. Nous noterons aussi que ces calculs préalables sont tout autant nécessaires dans le cas des fonctions EQMo et EC pour la détermination des coefficients de fusion oracle selon le problème QP (3.19). Toutefois, l'évaluation des fonctions EQMs et SDR ne nécessitant pas la résolution de ce problème oracle, l'étape de calcul préalable sera donc sensiblement moins longue pour ces deux fonctions.

6.3 Expériences et discussion

Dans cette partie, nous proposons d'évaluer sur les corpus *CHiME* et *ccMixter* la fusion adaptative par réseaux de neurones que nous venons d'introduire. Nous aimerions ici souligner que notre implémentation des réseaux de neurones a été grandement facilitée par l'excellente librairie *Theano*[1] développée principalement par l'université de Montreal [Bastien et al., 2012; Bergstra et al., 2010]. Cette librairie *Python* permet l'implémentation facile d'algorithmes mathématiques sous forme symbolique et leur compilation optimisée pour exécution sur CPU ou GPU, selon le choix, et ce de façon totalement transparente pour l'utilisateur. Ainsi, tous les calculs relatifs aux réseaux de neurones présentés ici ont été réalisés sur processeurs graphiques grâce à notre propre implémentation exploitant la puissance de *Theano*.

Dans la partie 6.3.1, nous présenterons les résultats obtenus sur le corpus *CHiME* pour le rehaussement de la parole et dans la partie 6.3.2, nous discuterons des performances sur le corpus *ccMixter* d'extraction de voix chantée. Enfin, nous conclurons succinctement dans la partie 6.4.

6.3.1 Corpus CHiME

Comme nous l'avons déjà évoqué dans la partie 6.2, l'apprentissage d'un réseau de neurones implique un grand nombre de choix d'hyperparamètres tels que le nombre, la taille, la topologie des couches cachées, *etc.* Il serait trop ambitieux d'étudier l'influence de tous ces hyperparamètres et par conséquent, il nous a fallu faire des choix.

En particulier, nous avons choisi de fixer notre choix de fonction d'activation à la fonction ReLU définie à l'équation (6.16). Comme cela a déjà été évoqué plus tôt, ce type de fonction d'activation a montré de bonnes propriétés pour l'apprentissage de réseaux profonds, permettant notamment d'éviter l'étape de pré-apprentissage particulièrement coûteuse en temps de calcul [Zeiler et al., 2013].

Les dimensions du réseau sont en partie fixées par les dimensions du problème. Pour rappel, nous considérons sur le corpus *CHiME* la fusion de $M = 7$ modèles de NMF définis dans le tableau 3.1, pour des nombres de composantes $K_{1m} = 2^m$ avec $m = 1..M$. De ce fait, la couche de sortie

1. http://deeplearning.net/software/theano/

de notre réseau sera composée de $M = 7$ neurones de fonction d'activation *softmax* (6.5), de façon à respecter la contrainte de sommation à 1 des coefficients de fusion (3.3).

La taille de la couche d'entrée de notre réseau peut elle varier en fonction du type d'entrée considéré, dont la forme générale a été définie à l'équation (6.20). En particulier, nous proposons de mesurer l'influence du contexte en comparant des réseaux dont les entrées prennent compte du contexte avec $C = 2$ à des réseaux dont les entrées n'intègrent pas ce contexte ($C = 0$). Indépendamment du contexte, l'entrée du réseau peut varier en fonction des spectres qui composent les descripteurs d'entrée. Nous avons envisagé les trois configurations suivantes :

— seuls les spectres de log-puissance des mélanges ($\log|\mathbf{x}_n|^2$) composent l'entrée,

— seuls les spectres de log-puissance des sources estimées ($\log|\tilde{\mathbf{s}}_{jm,n}|^2$) composent l'entrée,

— les spectres de log-puissance des mélanges et des sources estimées sont tous deux considérés.

De plus, comme suggéré dans la partie 6.2.2, nous avons réduit la dimension des matrices ainsi constituées grâce à une PCA afin de garder 85% de la variance des données, quels que soient la taille du contexte et les spectres choisis pour composer l'entrée. Les spectres ont été obtenus par transformée QERB, avec une fenêtre sinusoïdale de 1024 échantillons, un recouvrement de 50% et 350 bandes de fréquence. Les données d'apprentissage ont de plus été normalisées par moyenne et variance avant et après PCA.

Ci-après, nous étudierons également l'influence de la taille et du nombre de couches cachées. En particulier, nous ferons varier le nombre de couches cachées de 1 à 4. Indépendamment de ce nombre, nous avons fixé arbitrairement toutes les couches à la même taille choisie comme un multiple de la taille de la couche de sortie $M = 7$, parmi un total de 11 tailles différentes $\{7, 14, 28, 56, 112, 224, 448, 896, 1792, 3584, 7168\}$.

Enfin, nous avons fait varier la fonction de coût choisie pour l'apprentissage, parmi les quatre propositions introduites dans la partie 6.2.4 : fonction d'EQM oracle (6.23), fonction d'entropie croisée (EC) (6.24), fonction d'EQM source (6.25) et fonction SDR (6.27). Quelle que soit la fonction de coût choisie pour l'apprentissage et quels que soient les autres hyperparamètres envisagés, nous avons appris les paramètres des réseaux par rétropropagation et contrôlé le sur-apprentissage par évaluation à chaque époque du SDR sur l'ensemble de validation, selon la technique introduite par [Duffner et Garcia, 2007] et présentée dans la partie 6.1.4. Notons bien que, indépendamment de la fonction de coût choisie pour l'apprentissage, nous avons toujours évalué l'erreur de validation par calcul du SDR. Le taux d'apprentissage a été initialisé à la valeur $\eta_{\text{init}} = 10^{-3}$ et sa valeur a été ensuite adaptée selon les formules données à l'équation (6.19), avec $d = 0.9$, $u = 1.2$ et $\vartheta = 0.05$.

Il serait illusoire de vouloir ici rendre compte de tous les tests que nous avons réalisés en pratique, comme il a été impossible de tester toutes les combinaisons d'hyperparamètres que nous venons de lister. Toutefois, nous avons testé un grand nombre de ces configurations possibles et proposons donc de présenter nos résultats de la manière suivante. Nous commencerons par commenter l'architecture ayant donné les meilleurs résultats sur notre ensemble de validation, puis nous commenterons les performances des architectures ne différant que par le changement d'un des hyperparamètres que nous avons listés.

Meilleure architecture

Les meilleurs performances sur l'ensemble de validation ont été obtenues pour un réseau composé d'une seule couche cachée, appris par minimisation de la fonction SDR (6.27) et dont l'entrée était composée des spectres de log-puissance du mélange et des sources estimées avec un contexte de taille $C = 2$. La dimension de l'entrée a de plus été réduite par PCA, permettant une réduction de la dimension de 14 000 (soit, 350 bandes de fréquence $\times\, 5$ trames de contexte $\times$ (7 sources séparées $+\, 1$ mélange)) à seulement 154 unités.

La figure 6.5 représente les SDRs obtenus sur les ensembles de validation et de test pour les 11 tailles envisagées pour la couche cachée. Nous pouvons alors constater que le meilleur SDR a été obtenu pour 896 neurones, atteignant ainsi 8.3 dB sur l'ensemble de validation et 8.5 dB sur l'ensemble de test. On peut également remarquer que les performances obtenues pour 448 et 1792 neurones, soit respectivement deux fois moins et deux fois plus de neurones, sont équivalentes puisque le SDR sur l'ensemble de test atteint dans ces deux cas 8.5 dB également. Toutefois, pour des nombres d'unités supérieurs à 1792 ou inférieurs à 448, les performances diminuent jusqu'à atteindre 7.8 dB pour le réseau ne disposant que de 7 neurones, soit une perte de 0.6 dB.

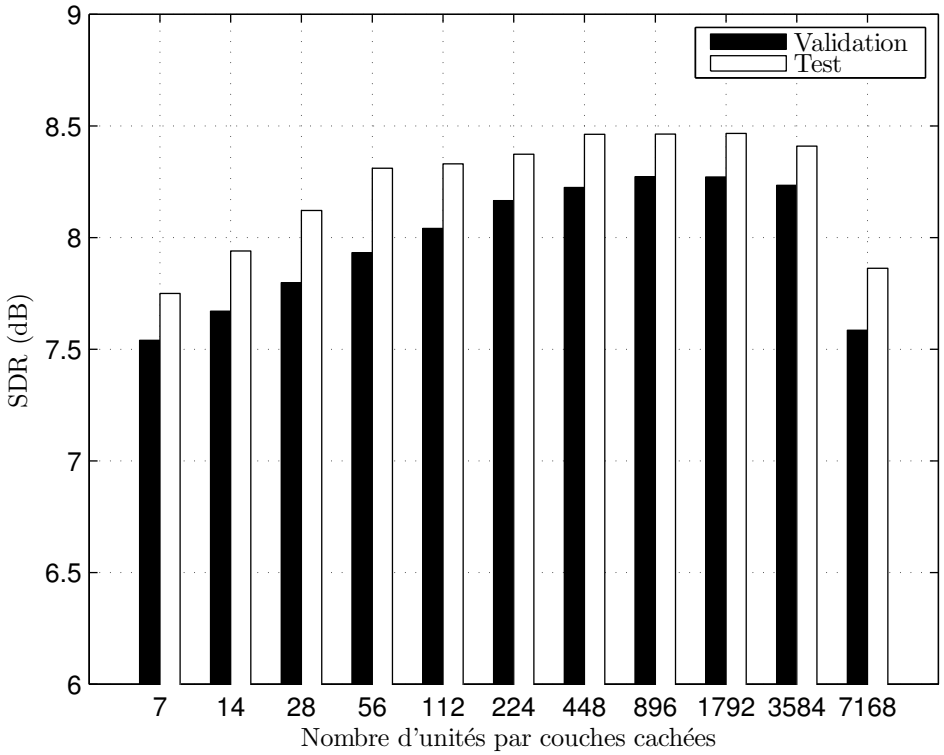


FIGURE 6.5 – Performances (SDR) évaluées sur les ensembles de validation et de test pour la meilleure architecture et avec différents nombres d'unités cachées.

Si cette perte de SDR peut paraître indésirable, elle est toutefois à relativiser. En effet, la figure 6.6 permet de comparer les résultats obtenus pour la meilleure architecture avec 896 neurones sur la couche cachée aux principaux résultats évoqués dans les chapitres précédents, notamment les résultats de fusion statique par moyenne, par médiane et de fusion statique invariante par minimisation de l'EQM et par maximisation du SDR. Nous constatons alors que la fusion adaptative variant en temps par réseau de neurones offre un gain de SDR très important par rapport aux fusions statiques. Par exemple, elle permet de gagner 2.8 dB de SDR par rapport à l'apprentissage de coefficients de fusion invariante. En comparaison à la simple sélection statique invariante par maximisation du SDR moyen, le gain atteint même 3.3 dB de SDR. Nous pouvons de plus constater, en nous référant aux résultats oracles introduits dans le tableau 3.3 de la partie 3.3.6, que les performances dépassent également le SDR de fusion oracle invariante de plus de 0.9 dB. Cependant, en comparaison aux résultats oracles de fusion variant en temps donnés dans ce même tableau, nous noterons que la marge de progression reste relativement importante puisque le SDR de la fusion oracle variant en temps pour la même fenêtre d'analyse est de presque 2 dB supérieur au SDR obtenu ici en pratique.

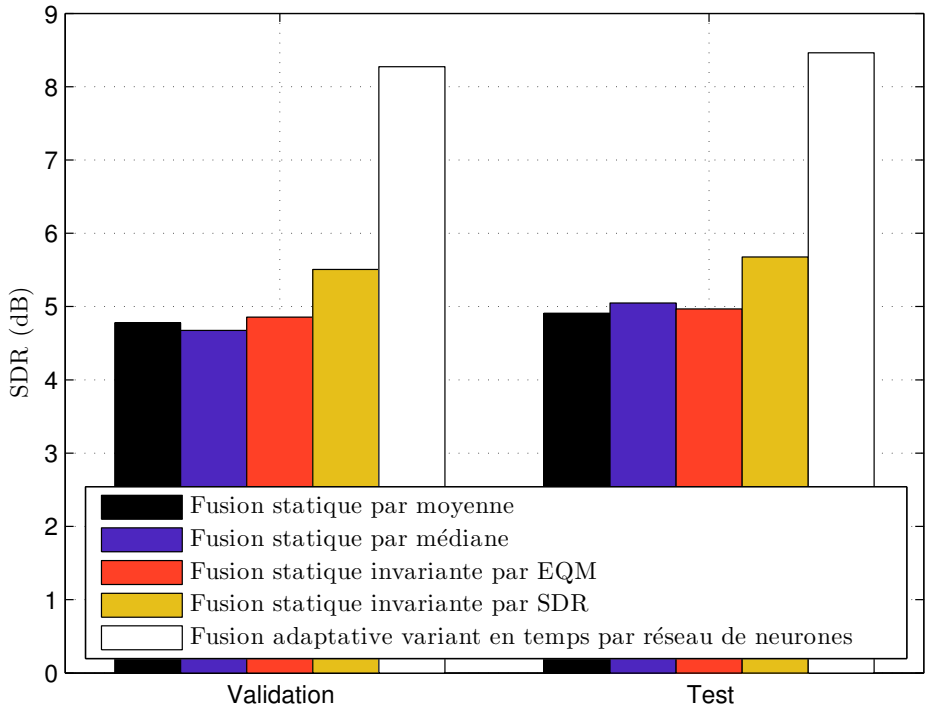


FIGURE 6.6 – Performances (SDR) évaluées sur les ensembles de validation et de test pour la meilleure architecture et les fusions statiques invariantes par moyenne, médiane, minimisation de l'EQM et maximisation du SDR.

Influence du nombre de couches

Bien que les résultats en terme de SDR semblent satisfaisants, il peut paraître déroutant d'obtenir ces performances à partir d'un réseau à une seule couche cachée, n'exploitant donc pas la puissance potentielle des réseaux de neurones profonds. Nous avons regroupé dans le tableau 6.1 quelques résultats choisis faisant varier le nombre de couches cachées (précisé dans la première colonne) et le nombre de neurones par couche (précisé dans la deuxième colonne). Nous avons également indiqué dans la troisième colonne le nombre de paramètres à estimer (biais et poids) que contient chacun des réseaux de neurones considérés. Ainsi, nous pouvons constater que les réseaux composés de deux ou trois couches cachées ont des SDRs très similaires, quoiqu'un peu inférieurs, à ceux composés d'une seule couche cachée, et ce pour des nombres de paramètres équivalents aux réseaux à une seule couche. Seuls les réseaux composés de quatre couches ont des performances nettement inférieures, comparables aux résultats obtenus par fusion statique invariante par maximisation du SDR moyen, indiquant alors très probablement un problème de sur-apprentissage ou en tout cas la convergence vers un minimum local non-optimal.

Influence des autres hyperparamètres

Nous proposons finalement d'observer l'influence des autres hyperparamètres du réseau en ne faisant varier qu'un seul hyperparamètre à la fois. Nous avons donc conservé un réseau à une seule couche composée de 896 neurones. Les résultats obtenus sont illustrés sur la figure 6.7, pour les ensembles de validation et de test.

La première barre représente le SDR obtenu par la meilleure architecture plus tôt évoquée. Les deuxième, troisième et quatrième barres représentent les SDRs obtenus pour les trois fonctions de coût, autre que le SDR, introduites dans la partie 6.2.4. Nous pouvons alors constater que les deux

Nombre de couches cachées	Nombre de neurones par couche	Nombre de paramètres	SDR sur l'ensemble de test (dB)
1	224	36,295	8.32
	448	72,583	8.46
	896	145,159	8.46
	1792	290,311	8.47
	3584	580,615	8.41
2	224	86,695	8.29
	448	273,735	8.32
3	224	137,095	8.22
	448	474,887	8.24
4	224	187,495	5.66
	448	676,039	5.66

TABLEAU 6.1 – Nombre de paramètres et performance de quelques réseaux en fonction du nombre de couches cachées et du nombre de neurones par couche.

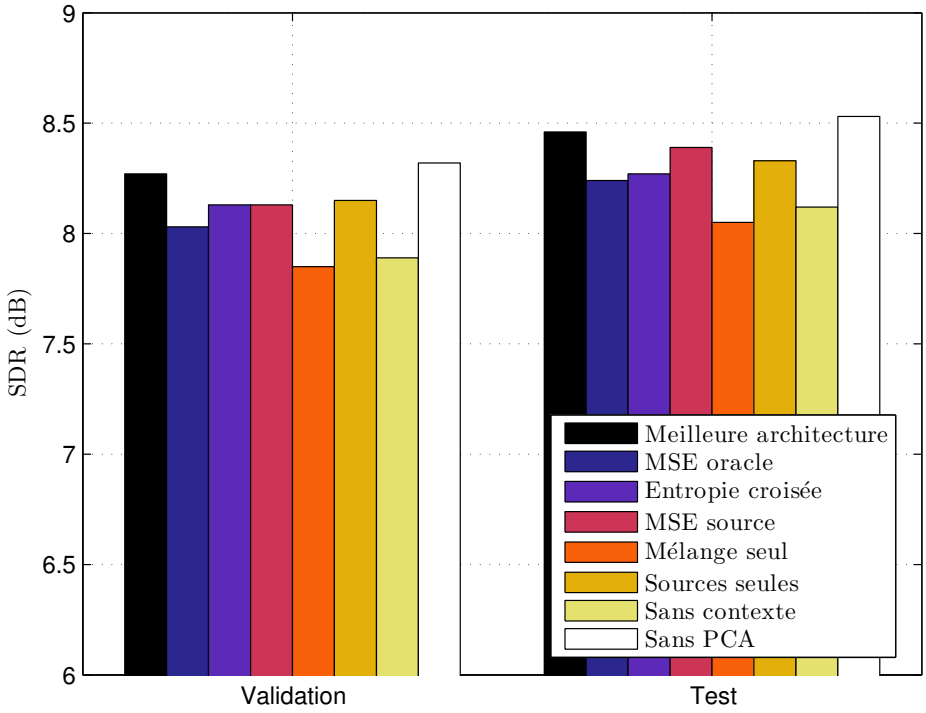


FIGURE 6.7 – Influence de certains hyperparamètres sur les performances (SDR) de la meilleure architecture évaluées sur les ensembles de validation et de test.

fonctions de coût n'exploitant pas la connaissance des coefficients de fusion oracle (c'est-à-dire, la fonction SDR et la fonction d'EQM source) permettent un gain de 0.2 dB de SDR environ par rapport aux fonctions classiques d'entropie croisée et d'EQM oracle, sur l'ensemble de test. Si, sur l'ensemble de validation, fonctions d'entropie croisée et d'EQM source ont une performance semblables, le léger gain apporté par la fonction d'EQM source sur l'ensemble de test suggère que cette dernière permet une meilleure généralisation.

Les quatre barres suivantes portent sur le type d'entrée considérée. Les cinquième et sixième barres correspondent, respectivement, à une entrée composée des spectres de log-puissance des mélanges uniquement et à une entrée composée des spectres de log-puissance des sources uniquement. Les SDRs obtenus étant inférieurs au SDR obtenu par la meilleure des architectures, ceux-ci suggèrent que l'information portée par la connaissance du mélange et l'information portée par la connaissance des sources ont toutes deux une importance certaine pour l'apprentissage. Les sources séparées semblent à ce titre contenir plus d'information utile que le mélange. L'avant-dernière barre permet elle de mesurer l'influence de la prise en compte du contexte. Comme pour les applications en reconnaissance de la parole, notre expérience montre que la prise en compte d'un contexte de taille $C = 2$ permet un gain de 0.35 dB.

Enfin, sur la toute dernière barre, nous avons fait figurer les performances obtenues en n'effectuant pas de réduction de la dimension de l'entrée par PCA. Nous constatons que le SDR obtenu sur l'ensemble de test est supérieur de 0.07 dB par rapport à la meilleur architecture employant une PCA pour réduire la dimension de l'entrée, ce qui suggère que la PCA a effectivement permis de conserver les composantes essentielles de l'entrée sans trop dégrader les performances de l'apprentissage. Toutefois, nous noterons que ce petit gain s'est fait au prix d'un effort supplémentaire de calcul considérable puisqu'il nous a fallu 3 jours, 9 heures et 50 minutes pour entraîner notre réseau sans PCA contre seulement 1 heure et 30 minutes dans le cas avec PCA (sur un PC muni d'une carte graphique *NVIDIA Quadro 600* et d'un CPU à quatre cœurs *Intel Xeon*).

6.3.2 Corpus ccMixter

Comme pour les chapitres précédents, nous avons également évalué le potentiel de la fusion adaptative variant en temps sur notre corpus d'extraction de voix chantée. Pour ce corpus, nous nous sommes inspirés des résultats obtenus sur le corpus CHiME et n'avons donc pas testé autant d'architectures que précédemment. Ainsi, nous n'avons fait varier ici que le nombre de couches (de 1 à 4) ainsi que la fonction de coût employée pour l'apprentissage (fonctions SDR et EQM source). La taille des couches cachées a été fixée à 512 unités (ce qui correspond au même rapport entre taille de la couche cachée et taille de la couche de sortie que dans le cas du rehaussement de la parole pour la meilleure architecture). L'entrée du réseau a été composée selon la définition (6.2.2) avec spectres du mélange et sources estimées ($M = 4$ dans ce cas) ainsi qu'un contexte de taille $C = 2$. Cette fois, la transformation QERB a été utilisée avec une fenêtre sinusoïdale de 2048 échantillons, un recouvrement de 50% ainsi que 350 bandes de fréquence. Les données d'apprentissage ont été normalisées par la moyenne et l'écart-type de l'ensemble d'apprentissage puis réduites par PCA avant d'être à nouveau normalisées par moyenne et variance après PCA.

Les résultats obtenus sont regroupés dans le tableau 6.2 pour chaque groupe de données et la performance moyenne sur les cinq groupes est donnée dans la dernière colonne. Contrairement aux expériences menées sur le corpus *CHiME*, nous pouvons remarquer que c'est la minimisation de l'EQM source qui donne cette fois les meilleures performances, avec un gain faible cependant.

Mais la différence fondamentale avec les expériences menées sur le corpus CHiME se trouve révélée lorsque l'on compare les résultats de fusion adaptative variant en temps présentés dans le tableau 6.2 aux résultats de fusion statique invariante et variant en fréquence présentés dans le tableau 4.3. En effet, alors que la fusion adaptative variant en temps permettait de gagner près de 3 dB par rapport à la fusion statique sur le corpus CHiME, la fusion adaptative ne permet ici

Fonction de coût	Nombre de couches	Groupe 1	Groupe 2	Groupe 3	Groupe 4	Groupe 5	Moyenne des groupes
EQMs	1	4.92	5.01	3.81	3.35	3.91	4.20
EQMs	2	4.88	5.02	3.77	3.29	3.96	4.18
EQMs	3	4.90	5.01	3.78	3.28	3.95	4.18
EQMs	4	4.87	5.01	3.72	3.27	3.99	4.17
SDR	1	4.27	5.06	3.68	3.15	3.71	4.01
SDR	2	4.48	4.98	3.62	3.30	3.94	4.06
SDR	3	4.44	5.03	3.74	3.05	4.00	4.05
SDR	4	4.40	5.02	3.77	3.29	3.96	4.08

TABLEAU 6.2 – Performance (SDR en dB) de fusion adaptative variant en temps pour différents nombres de couches cachées et pour les fonctions de coût EQM source (EQMs) et SDR.

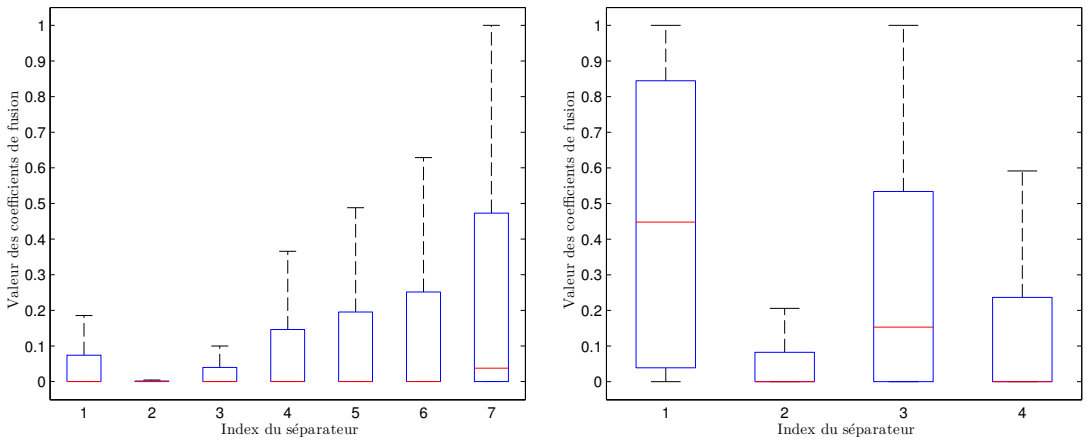
qu'un gain de 0.3 dB par rapport à la fusion statique invariante et moins de 0.2 dB par rapport à la fusion statique variant en fréquence. De plus, nous constatons que, quel que soit le nombre de couches, la performance reste relativement la même, ce qui nous laisse craindre une convergence vers un minimum local non optimal. Ce résultat est peut-être en partie dû à l'attention moindre que nous avons porté à la recherche de la meilleure architecture relativement aux données de ce corpus. Toutefois, d'autres pistes peuvent également expliquer ce phénomène et nous proposons d'en discuter quelques unes dans la partie suivante.

6.4 Conclusion

Dans ce chapitre, nous avons proposé d'apprendre des coefficients de fusion variant en temps en exploitant des réseaux de neurones potentiellement profonds. Nous avons pour cela décrit une architecture de réseau proactif adaptée à notre problème de fusion, dont la sortie est composée des coefficients de fusion variant en temps pour une trame temporelle donnée et l'entrée des spectres de log-puissance QERB du mélange et des sources estimées ainsi que de leur contexte. Nous avons par ailleurs proposé de modifier le schéma d'apprentissage classique en introduisant deux fonctions de coût directement liées à notre objectif de séparation.

Évaluée sur nos deux corpus, notre approche s'est révélée particulièrement efficace dans le cas du rehaussement de la parole. En effet, les performances obtenues en terme de SDR ont dépassé très largement les performances jusqu'alors atteintes par les approches de fusion statique et adaptative VB. Pour la tâche d'extraction de voix chantée, les résultats paraissent moins attrayants car le gain en SDR par rapport aux fusions statiques invariante et variant en fréquence, obtenu au prix d'un apprentissage plus lourd en calculs, semble dérisoire.

Cet écart de performance peut sans doute s'expliquer par les caractéristiques spécifiques de chacun des problèmes que nous nous sommes proposés de résoudre. En effet, le fait que les exemples du corpus CHiME ainsi que les modèles envisagés pour la fusion soient assez homogènes semble rendre l'apprentissage plus aisé. Au contraire, le cas de l'extraction de voix chantée présente plus de variabilité aussi bien dans les exemples qui forment le corpus que dans les séparateurs que nous avons envisagés pour la fusion. La complexité de la tâche d'apprentissage à réaliser diffère donc fortement entre les deux corpus et dans le cas du corpus *ccMixter*, l'ensemble d'apprentissage réuni n'est peut être pas assez grand pour permettre la convergence vers un meilleur optimum. Cette hypothèse semble confirmée par l'observation des distributions de coefficients de fusion oracle variant en temps sur les ensembles d'apprentissage, comme représentées sur la figure 6.8. On constate en effet que la variabilité des coefficients de fusion est bien plus importante sur le corpus ccMixter que sur le corpus CHiME. La tâche d'apprentissage en est d'autant plus compliquée.



(a) Corpus CHiME (b) Corpus ccMixter

FIGURE 6.8 – Distribution des coefficients de fusion oracle variant en temps sur les ensembles d'apprentissage des corpus CHiME et ccMixter.

Concernant le choix de la fonction de coût, nos expériences ont montré que le choix d'une fonction de coût liée à l'objectif de séparation, que ce soit le SDR ou l'EQM source, permettait en pratique d'améliorer les performances de fusion, comparativement à des fonctions de coût plus classiques comme l'EQM oracle ou l'entropie croisée. Nos résultats ne nous permettent cependant pas d'émettre une préférence entre SDR et EQM source, car les performances obtenus avec ces deux fonctions sont équivalentes en terme de SDR (plus précisément, légèrement supérieures avec le SDR sur le corpus *CHiME*, et légèrement supérieures avec l'EQM source sur le corpus *ccMixter*). Il pourrait être alors judicieux d'étudier d'autres fonctions de coûts afin de peut-être améliorer plus encore la performance. Suivant la proposition faite dans [WENINGER et al., 2014], il pourrait être également envisagé de procéder à l'apprentissage d'un réseau en deux temps : d'abord, en optimisant les paramètres une première fois par rapport à une fonction de coût classique (EQM oracle ou entropie croisée par exemple), puis en affinant l'apprentissage par une deuxième étape d'optimisation des paramètres par rapport à une autre fonction de coût liée cette fois à l'objectif de séparation (EQM source ou SDR). Dans [WENINGER et al., 2014], pour l'apprentissage d'un masque temps-fréquence, il a été en effet démontré que les performances de séparation étaient améliorées lorsque l'apprentissage du réseau était d'abord mené par rapport à l'EQM entre masques estimés et masques oracles puis par rapport à l'EQM entre sources estimées et sources oracles.

Nous croyons également que les performances de fusion adaptative variant en temps obtenues dans ce chapitre peuvent être encore améliorées, au seul prix d'une adaptation plus poussée des hyperparamètres des réseaux considérés ou, éventuellement, en recourant à d'autres architectures. Nous avons en ce sens mené d'autres expériences, en employant en particulier des réseaux récurrents ainsi que la technique de *dropout* présentée dans la partie 6.1.3. Si les réseaux récurrents appris par minimisation de l'EQM oracle n'ont pas apporté de gains, le *dropout* nous a permis d'améliorer les performances obtenues sur le corpus CHiME sur des réseaux à 4 couches cachées, sans toutefois dépasser les performances que nous avons présentées dans la partie 6.3.1. Pour aller plus loin, les perspectives les plus prometteuses consisteraient à envisager des moyens d'augmenter le contexte C pris en compte en entrée. En effet, étant donné le choix de trames calculés à l'aide de fenêtres avec recouvrement de 50%, il apparaît que la taille de contexte choisie dans nos expériences ($C = 2$) semble trop petite au vue des tailles choisies dans la littérature du traitement de la parole qui sont plutôt de $C = 4$ ou $C = 5$ (soit . Pour cela, de nouveaux tests avec $C = 5$ pourraient être menés. Une autre piste consisterait à employer des réseaux récurrents appris cette fois par optimisation d'une fonction liée à l'objectif de séparation (minimisation de l'EQM source ou maximisation du

SDR). Ce sont en effets de tels réseaux qui se sont montrés les plus performants pour l'estimation de masques temps-fréquence dans [Weninger et al., 2014].

Enfin, afin d'apprécier au mieux les performances de la fusion adaptative variant en temps, il conviendrait sans doute de la comparer aux approches de séparation par réseaux de neurones profonds évoqués en introduction de ce chapitre. Nous pensons toutefois que la fourniture en entrée des estimées des sources constitue une information forte permettant de guider l'apprentissage. De ce fait, les approches de séparation par réseaux de neurones profonds pourraient également profiter de ce principe en fournissant à leur entrée les sources estimées par d'autres séparateurs, ce qui reviendrait simplement à changer la sortie des réseaux que nous avons ici étudiés pour qu'ils n'estiment plus des coefficients de fusion mais directement des masques temps-fréquence ou les spectres d'amplitude ou de puissance des sources à estimer.

Chapitre 7

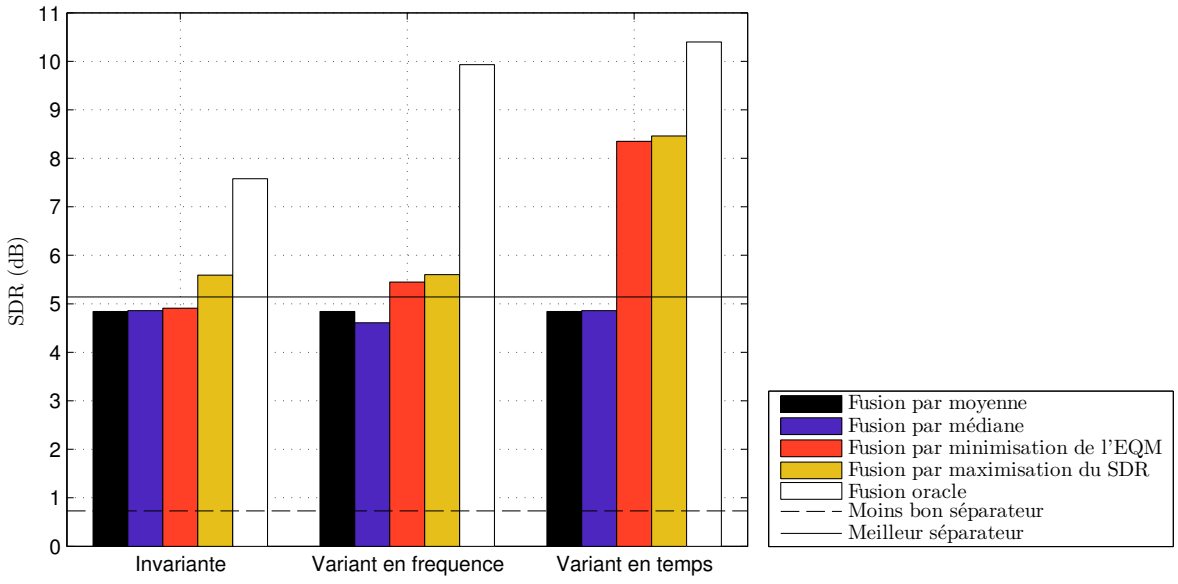
Conclusion et perspectives

Dans cette thèse, nous avons introduit un cadre général de fusion pour la séparation de sources et évalué son potentiel sur des cas d'application pratiques. Nous proposons dans ce dernier chapitre de conclure notre travail en en résumant les principales contributions et en proposant quelques idées de travaux futurs qui pourront, nous l'espérons, mener à de nouveaux progrès.

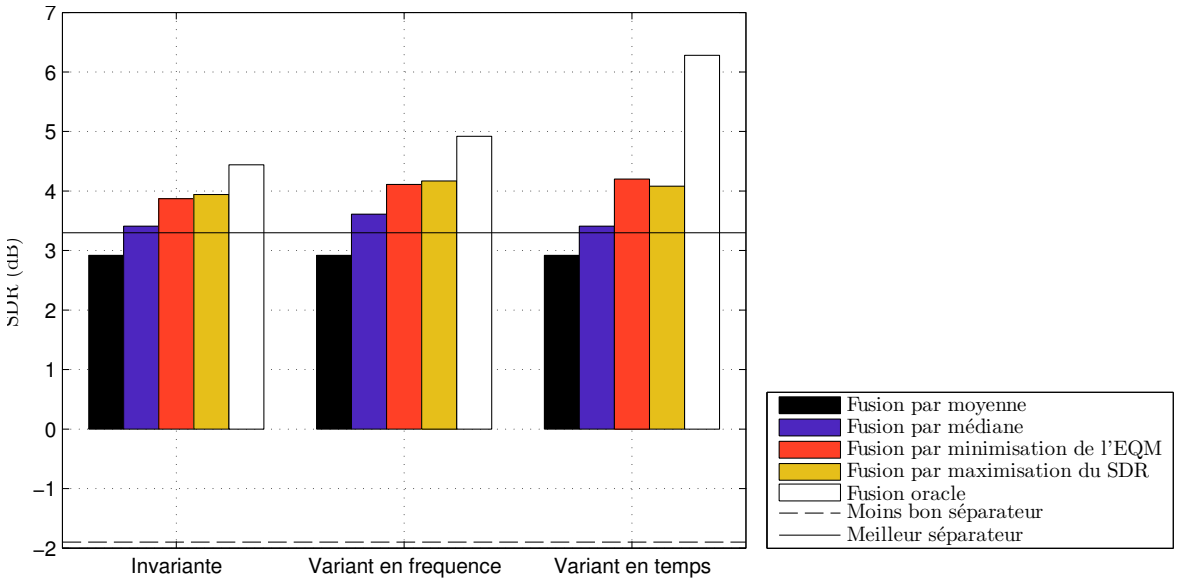
7.1 Mieux vaut fusionner que sélectionner

Le premier enseignement que nous retenons est que, comme dans bien d'autres domaines, la fusion de séparateurs est une approche à privilégier, si possible, par rapport à la simple sélection. En effet, nous avons montré tout au long de ce manuscrit que les performances de fusion permettait toujours d'améliorer la qualité de séparation globale, et ce qu'elle soit invariante, variant en temps ou variant en fréquence, qu'elle soit oracle, statique ou adaptative.

De plus, si l'on se rappelle les raisons initiales qui nous ont amené à entreprendre ce travail et que nous avons évoquées en introduction, toutes les méthodes de fusion introduites ici nous semblent être des alternatives intéressantes par rapport aux procédés de séparation généralement employés. En effet, rappelons qu'en pratique, face à un problème de séparation donné, le choix d'un séparateur est généralement unique et guidé par l'expérience. Grâce à nos approches de fusion, nous avons pu mettre en place des procédures permettant d'améliorer la qualité de séparation par rapport à un processus de séparation classique. Pour illustrer ce propos, nous avons tracé sur la figure 7.1 les principaux résultats de fusion que nous avons obtenus sur nos deux corpus d'évaluation. Nous avons également tracé deux résultats de séparation sans fusion. En trait plein, figure le SDR obtenu par le séparateur dont le SDR moyen sur l'ensemble de test (ou sur les ensembles de test dans le cas du corpus *ccMixter*) est le plus élevé parmi les séparateurs fusionnés. En trait pointillé, figure le SDR obtenu par le séparateur dont le SDR moyen sur l'ensemble de test est le moins élevé. Nous pouvons alors constater que, quel que soit le corpus, nos méthodes de fusion font toujours mieux que le moins bon des séparateurs envisagés pour la fusion, et presque toujours mieux que le meilleur des séparateurs. En pratique donc, plutôt que de ne sélectionner qu'un séparateur, fusionner les résultats obtenus par plusieurs séparateurs, selon les règles de fusion que nous avons introduites, semble être une alternative très séduisante. En particulier, la fusion variant en temps est celle qui nous a permis d'atteindre les meilleurs performances.



(a) Corpus *CHiME*



(b) Corpus *ccMixter*

FIGURE 7.1 – Performance (SDR) de la fusion statique invariante par moyenne, des fusions statiques invariante et variant en fréquence par médiane, par minimisation de l'EQM et par maximisation du SDR, des fusions adaptatives VB variant en temps par minimisation de l'EQM et par maximisation du SDR, pour les corpus *CHiME* et *ccMixter*. Pour chaque corpus, sont également tracés le meilleur SDR individuel moyen en trait plein et le moins bon SDR individuel moyen en trait pointillé.

7.2 Fusion par apprentissage : pour aller plus loin

De toutes les méthodes que nous avons envisagées, celles faisant appel à une étape d'apprentissage sont sans aucun doute les méthodes les plus performantes. Ces dernières requièrent toutefois de constituer un ensemble d'apprentissage représentatif de la tâche de séparation à effectuer. Dans ce cas, nous avons montré qu'il était possible de déterminer des coefficients de fusion par minimisation d'une fonction de coût sur cet ensemble. Ainsi, que ce soit pour l'apprentissage de coefficients de fusion statique invariante ou variant en fréquence, ou pour l'apprentissage de coefficients de fusion adaptative variant en temps par réseaux de neurones, nous avons montré que la minimisation d'une fonction de coût corrélée à l'objectif de séparation permettait effectivement d'améliorer la qualité globale de séparation par rapport aux approches sans apprentissage ainsi que par rapport aux approches par sélection correspondantes.

Plusieurs pistes pourraient être explorées pour améliorer nos résultats. En effet, quel que soit le corpus et comme le montre la figure 7.1, il reste un gain potentiel important entre les performances obtenues par apprentissage et les performances oracles. Si ces performances de fusion oracle resteront sans doute inaccessibles, il ne nous semble pas impossible d'améliorer les résultats obtenus pour nos trois cas de fusion par apprentissage.

En premier lieu, quel que soit le cas de fusion par apprentissage, il pourrait être envisagé d'étudier d'autres fonctions de coût. Les fonctions de coût que nous avons retenues dans notre étude avaient pour avantage non-négligeable de permettre une résolution aisée et peu coûteuse en temps de calcul, qualité particulièrement appréciable dans le cas d'un apprentissage par réseau de neurones. Il ne peut être ignoré que d'autres fonctions de coût puissent avoir ce même avantage. Mais d'autres fonctions de coût, même plus complexes, pourraient mener à de substantielles améliorations des performances. Nous pensons en particulier aux autres mesures présentées dans la partie 2.1.5 (SIR, SAR [Vincent et al., 2006] ou d'autres mesures objectives [Vincent, 2012]) ou éventuellement une combinaison de certaines de ces mesures. De même, d'autres types de divergence pourraient être envisagés, telles que les β-divergences [Kompass, 2007].

Pour le cas particulier de la fusion variant en fréquence, nous aimerions de plus rappeler que la détermination des coefficients de fusion a été menée indépendamment sur chaque bande de fréquence afin de simplifier le problème d'optimisation à résoudre. En minimisant ces mêmes fonctions de coût conjointement sur toutes les bandes de fréquence considérées, il est fort probable que les performances de fusion s'en trouvent améliorées, au prix tout de même d'un effort de calcul plus important.

Enfin, pour le cas variant en temps, l'apprentissage mené par réseau de neurones a montré des performances bien supérieures aux autres approches sur le corpus *CHiME*, mais plus mitigées (bien que supérieures également) sur le corpus *ccMixter*. Nos expériences ont également montré que la puissance des réseaux de neurones profonds n'étaient pas pleinement exploitées. À ce titre, il nous semble que plusieurs voies pourraient être envisagées pour améliorer les performances de fusion et approcher les performances de fusion oracle variant en temps. En premier lieu, d'autres architectures de réseau (réseau convolutionnel [Sprechmann et al., 2015], réseau récurrent [Weninger et al., 2014]) pourraient être étudiées pour cette tâche, en veillant à optimiser de tels réseaux relativement à une fonction de coût liée à l'objectif de séparation. De même, l'influence du choix des descripteurs de l'entrée pourrait être étudiée et il pourrait être par exemple envisagé de considérer plus de contexte, de ne pas réduire d'autant la dimension de l'entrée par PCA, de choisir d'autres descripteurs de l'entrée que les spectres de log-puissance ERB, *etc.*. Finalement, ces choix pourrait être menés de façon automatique afin de découvrir la meilleure topologie de réseau, étant donné le problème posé. Des travaux de recherche en ce sens ont d'ailleurs débuté récemment [Shinozaki et Watanabe, 2015] et il pourrait donc être opportun de s'en inspirer.

7.3 Fusion sans apprentissage : défauts de l'approche bayésienne

Si les fusions avec apprentissage nous ont permis d'atteindre de bonnes performances en terme de qualité de séparation, il nous a semblé important de travailler également à proposer des méthodes de fusion ne requérant pas cet étape d'apprentissage. En effet, en pratique, il n'est pas toujours possbile de constituer un ensemble d'apprentissage dédié à la tâche de séparation considérée. Outre les approches simples par moyenne ou médiane, l'étude de la NMF bayésienne nous a permis d'évoluer en ce sens. Toutefois, nous n'avons pas obtenu les résultats escomptés.

En effet, dans la littérature relative à la sélection de modèles, l'approche bayésienne fait bien souvent référence [Bishop, 2006]. Dans nos expérimentations menées sur la NMF bayésienne, nous avons pourtant touché certaines de ses limites. En particulier, nos expériences sur le corpus *CHiME* ont montré que, si le critère de sélection bayésien permettait parfois de retrouver le nombre de composantes ayant été utilisé pour générer des données synthétiques, il ne permettait jamais sur données réelles de sélectionner le nombre de composantes maximisant le SDR. De plus, nous avons également montré que le moyennage bayésien de modèles ne permettait pas en pratique d'opérer une fusion.

L'introduction d'un paramètre de contrôle de l'entropie de la distribution *a posteriori* du nombre de composantes nous a permis de réduire ces défauts dans une certaine mesure. Nous avons montré sur données synthétiques que ce paramètre rendait la fusion par moyennage bayésien effective. Sur données réelles, la mise en œuvre de ce paramètre nous a toutefois demandé de mettre en place une procédure d'apprentissage de la valeur de ce paramètre ainsi que des probabilités *a priori* des nombres de composantes. De ce fait, l'objectif initial de fusion sans apprentissage n'a pas été atteint.

Pour atteindre tout de même cet objectif de fusion sans apprentissage, l'approche bayésienne proposée pourrait être modifiée de plusieurs façons. Par exemple, en conservant l'approche variationnelle que nous avons développée, il pourrait être envisagé d'améliorer les diverses approximations rendues nécessaires pour parvenir à une solution analytique. En particulier, nous avons déjà évoqué la possibilité d'exploiter l'*inférence variationnelle stochastique structurée* [Hoffman, 2014b] pour approcher au mieux la vraisemblance marginale d'un modèle. Le recours à d'autres méthodes d'approximation que l'approche variationnelle, telles que les méthodes par échantillonnage [Bishop, 2006], pourrait être également étudié, bien que l'inférence variationnelle ait l'avantage d'être peu gourmande en temps de calcul, comparativement aux approches par échantillonnage. Enfin, nous avons également évoqué en conclusion du chapitre 5 la possibilité de formuler la fusion d'autres paramètres que les estimées des sources, ce qui nous mènerait toutefois hors du champ d'application du cadre de fusion que nous avons développé ici.

7.4 Vers une fusion par apprentissage moins contrainte

Dans cette thèse, nous avons finalement montré que notre cadre de fusion pouvait amener de très bonnes performances, notamment pour le cas variant en temps, à condition de constituer un bon ensemble d'apprentissage, caractéristique du problème de séparation envisagé. En ce sens, notre proposition répond bien à l'objectif industriel initial qu'a essayé de résoudre ce travail. Cependant, il nous paraît légitime de se demander si le choix de la fusion par apprentissage est judicieux alors même que nous pouvons être amené à fusionner des modèles de séparation ne requérant pas d'apprentissage.

Pour tenter d'y répondre, il convient de prendre un peu de distance avec notre problème et de constater simplement qu'avec l'avènement de l'apprentissage profond et du *big data*, l'apprentissage sur des ensembles de grande dimension ne semble plus être un objectif inaccessible [Chen et

Lin, 2014]. Pour cette raison, la fusion par apprentissage nous semble être la direction à privilégier, avec pour objectif de rendre l'apprentissage moins contraint. Pour ce faire, deux voies nous semblent particulièrement indiquées.

Nos expériences ayant montré que les performances de fusion adaptative par réseau de neurones dépendaient notamment de la qualité de l'ensemble d'apprentissage, il conviendrait donc, dans un premier temps, d'envisager des moyens pour améliorer la qualité de ces ensembles. Nous avons en effet constaté que la fusion variant en temps par réseau de neurones était bien moins performante sur le corpus *ccMixter* que sur le corpus *CHiME*. Pour combler cet écart de performance, il conviendrait donc de grossir l'ensemble d'apprentissage *ccMixter* et probablement de le diversifier. Pour ce faire, il pourrait être simplement envisagé de réunir en un seul grand ensemble les corpus *CHiME* et *ccMixter*. L'ensemble résultant ne serait donc plus dédié qu'à une seule tâche de séparation mais couvrirait à la fois le cas de l'extraction de voix chantée et le cas du rehaussement de la parole. Dans la littérature, il est fait référence à ce principe sous le nom d'*apprentissage multi-conditions* [Huang et al., 2014d] et les réseaux de neurones profonds semblent bien appropriés à la résolution de tels problèmes.

Afin de grossir plus encore l'ensemble d'apprentissage, il pourrait être également envisagé de recourir plutôt à un apprentissage semi-supervisé, où seule une partie des exemples de l'ensemble d'apprentissage permet effectivement un apprentissage supervisé [Huang et al., 2014a]. Par rapport aux fonctions de coût que nous avons étudiées, cela reviendrait à ne connaître les sources vraies (ou les coefficients de fusion oracle) que pour une partie seulement des exemples de l'ensemble. De ce fait, il deviendrait alors possible de grossir plus encore les ensembles d'apprentissage et, grâce à la puissance des réseaux de neurones, d'approcher au plus près des performances de fusion oracle.

En combinant ces deux idées, le recours à des ensembles d'apprentissage semi-supervisé et multi-conditions nous semble être l'un des moyens les plus prometteurs pour améliorer plus encore les performances de notre cadre de fusion, et ce quel que soit le problème de séparation considéré. Nous croyons de plus que, par comparaison aux approches classiques de séparation par réseaux de neurones profonds, l'utilisation en entrée des sources estimées constitue une information forte aidant à l'apprentissage. Nous espérons donc que ce travail, combiné à l'idée d'un apprentissage semi-supervisé multi-conditions, mènera très prochainement à de nouveaux progrès.

Références

Adiloğlu, K. et E. Vincent. 2012, «Variational Bayesian inference for source separation and robust feature extraction», rapport de recherche RT-0428, Inria. 93, 94, 95, 96, 97, 98, XIV

Akaike, H. 1992, «Information theory and an extension of the maximum likelihood principle», dans *Breakthroughs in statistics*, Springer, p. 610–624. 36

Alinaghi, A., W. Wang et P. J. B. Jackson. 2011, «Integrating binaural cues and blind source separation method for separating reverberant speech mixtures», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 209–212. 43, 44

Andrieu, C., N. De Freitas et A. Doucet. 1999, «Sequential MCMC for Bayesian model selection», dans *Proc. of the IEEE Signal Processing Workshop on Higher-Order Statistics*, p. 130–134. 35, 40

Araki, S., F. Nesta, E. Vincent, Z. Koldovský, G. Nolte, A. Ziehe et A. Benichoux. 2012, «The 2011 signal separation evaluation campaign (sisec2011) :-audio source separation», dans *Proc. of Int. Conf. on Latent Variable Analysis and Signal Separation (LVA/ICA)*, Springer, p. 414–422. 42

Arberet, S., A. Ozerov, F. Bimbot et R. Gribonval. 2012, «A tractable framework for estimating and combining spectral source models for audio source separation», *Signal Processing*, vol. 92, n° 8, p. 1886–1901. 44

Artin, E. 1931, «The Gamma function», *AMC*, vol. 10, p. 12. 95

Barker, J., E. Vincent, N. Ma, H. Christensen et P. Green. 2013, «The PASCAL CHiME speech separation and recognition challenge», *Computer Speech & Language*, vol. 27, n° 3, p. 621–633. 53

Bastien, F., P. Lamblin, R. Pascanu, J. Bergstra, I. Goodfellow, A. Bergeron, N. Bouchard, D. Warde-Farley et Y. Bengio. 2012, «Theano : new features and speed improvements», Workshop on Deep Learning and Unsupervised Feature Learning (NIPS). 153

Benaroya, L., L. Mcdonagh, F. Bimbot et R. Gribonval. 2003, «Non negative sparse representation for Wiener based source separation with a single sensor», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 6, p. VI–613–VI–616. 14

Benesty, J., S. Makino et J. Chen. 2005, *Speech enhancement*, Springer. 25

Bengio, Y. 2009, «Learning deep architectures for AI», *Foundations and trends in Machine Learning*, vol. 2, n° 1, p. 1–127. 146

Bergstra, J., O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley et Y. Bengio. 2010, «Theano : a CPU and GPU math expression compiler», dans *Proc. of the Python for Scientific Computing Conference (SciPy)*, vol. 4, p. 3–10. 153

Bertin, N., R. Badeau et G. Richard. 2007, «Blind signal decompositions for automatic transcription of polyphonic music : NMF and K-SVD on the benchmark», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, p. I–65–68. 33, 42, 59

Bertin, N., R. Badeau et E. Vincent. 2009a, «Fast Bayesian NMF algorithms enforcing harmonicity and temporal continuity in polyphonic music transcription.», dans *WASPAA*, p. 29–32. 20

Bertin, N., R. Badeau et E. Vincent. 2010, «Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription», *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, n° 3, p. 538–549. 20

Bertin, N., C. Févotte et R. Badeau. 2009b, «A tempering approach for Itakura-Saito non-negative matrix factorization. with application to music transcription», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 1545–1548. 17

Bertsekas, D. P. 1999, *Nonlinear programming*, Athena Scientific. 51

Bishop, C. M. 1995, *Neural networks for pattern recognition*, Clarendon press Oxford. 140

Bishop, C. M. 2006, *Pattern recognition and machine learning*, Springer. 34, 35, 37, 40, 93, 96, 131, 139, 144, 150, 165, XVI

Bishop, C. M. et J. Winn. 2003, «Structured variational distributions in VIBES», *Proc. of Artificial Intelligence and Statistics*, p. 244–251. 107

Bloch, I. 2003, *Fusion d'informations en traitement du signal et des images*, Hermes Science Publications. 37

Bobin, J., Y. Moudden, J.-L. Starck et M. Elad. 2005, «Multichannel morphological component analysis», dans *Proc. of Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS)*, p. 103–106. 15

Boll, S. 1979, «Suppression of acoustic noise in speech using spectral subtraction», *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, n° 2, p. 113–120. 25

Bregman, A. S. 1994, *Auditory scene analysis : The perceptual organization of sound*, MIT press. 13

Breiman, L. 1996a, «Bagging predictors», *Machine learning*, vol. 24, n° 2, p. 123–140. 38

Breiman, L. 1996b, «Stacked regressions», *Machine learning*, vol. 24, n° 1, p. 49–64. 39

Breiman, L. 2001, «Random forests», *Machine learning*, vol. 45, n° 1, p. 5–32. 38

Burnham, K. P. et D. R. Anderson. 2004, «Multimodel inference understanding AIC and BIC in model selection», *Sociological methods & research*, vol. 33, n° 2, p. 261–304. 40

Cai, M., Y. Shi et J. Liu. 2013, «Deep maxout neural networks for speech recognition», dans *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, p. 291–296. 147

Candès, E. J., X. Li, Y. Ma et J. Wright. 2011, «Robust principal component analysis ?», *Journal of the ACM (JACM)*, vol. 58, n° 3, p. 11. 29

Cardoso, J. 1998, «Multidimensional independent component analysis», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, p. 1941–1944. 11, 14

Casey, M. A. et A. Westner. 2000, «Separation of mixed audio sources by independent subspace analysis», dans *Proc. of the International Computer Music Conference*, p. 154–161. 14

Cemgil, A. T. 2009, «Bayesian inference for nonnegative matrix factorisation models», *Computational Intelligence and Neuroscience*. Article ID 785152. 42

Chandna, S. et A. T. Walden. 2013, «Simulation methodology for inference on physical parameters of complex vector-valued signals», *IEEE Transactions on Signal Processing*, vol. 61, n° 21, p. 5260–5269. 44

Chandna, S. et W. Wang. 2014, «Improving model-based convolutive blind source separation techniques via bootstrap», dans *Proc. of IEEE Statistical Signal Processing Workshop (SSP)*, p. 424–427. 43

Chen, X.-W. et X. Lin. 2014, «Big data deep learning : Challenges and perspectives», *IEEE Access*, vol. 2, doi :10.1109/ACCESS.2014.2325029, p. 514–525, ISSN 2169-3536. 165

Chib, S. 1995, «Marginal likelihood from the Gibbs output», *Journal of the American Statistical Association*, vol. 90, n° 432, p. 1313–1321. 42

Christensen, H., J. Barker, N. Ma et P. D. Green. 2010, «The CHiME corpus : a resource and a challenge for computational hearing in multisource environments.», dans *Proc. of Interspeech*, p. 1918–1921. 53

Cichocki, A., R. Zdunek, A. H. Phan et S. Amari. 2009, *Nonnegative matrix and tensor factorizations : applications to exploratory multi-way data analysis and blind source separation*, John Wiley & Sons. 15, 18, 93

Cohen, I. 2003, «Noise spectrum estimation in adverse environments : Improved minima controlled recursive averaging», *IEEE Transactions on Speech and Audio Processing*, vol. 11, n° 5, p. 466–475. 45

Cooke, M., J. Barker, S. Cunningham et X. Shao. 2006, «An audio-visual corpus for speech perception and automatic speech recognition», *The Journal of the Acoustical Society of America*, vol. 120, n° 5, p. 2421–2424. 53

Cortes, C., M. Mohri et U. Syed. 2014, «Deep boosting», dans *Proc. of International Conference on Machine Learning (ICML)*, p. 1179–1187. 39

Dessein, A., A. Cont et G. Lemaitre. 2010, «Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence», dans *Proc. of International Symposium on Music Information Retrieval (ISMIR)*, p. 489–494. 20

Doclo, S. et M. Moonen. 2002, «GSVD-based optimal filtering for single and multimicrophone speech enhancement», *IEEE Transactions on Signal Processing*, vol. 50, n° 9, p. 2230–2244. 25

Domingos, P. 2000, «Bayesian averaging of classifiers and the overfitting problem», dans *Proc. of International Conference on Machine Learning (ICML)*, p. 223–230. 40

Dréo, J., A. Pétrowski, É. D. Taillard et P. Siarry. 2003, *Métaheuristiques pour l'optimisation difficile*, Éditions Eyrolles. 4

Duan, Z., Y. Zhang, C. Zhang et Z. Shi. 2008, «Unsupervised single-channel music source separation by average harmonic structure modeling», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, n° 4, p. 766–778. 42

Duchi, J., E. Hazan et Y. Singer. 2011, «Adaptive subgradient methods for online learning and stochastic optimization», *The Journal of Machine Learning Research*, vol. 12, p. 2121–2159. 148

Duda, R. O., P. E. Hart et D. G. Stork. 2012, *Pattern classification*, John Wiley & Sons. 37, 144, 150

Duffner, S. et C. Garcia. 2007, «An online backpropagation algorithm with validation error-based adaptive learning rate», *Artificial Neural Networks*, p. 249–258. 148, 154

Duffy, N. et D. Helmbold. 2002, «Boosting methods for regression», *Machine Learning*, vol. 47, n° 2-3, p. 153–200. 39

Duong, N. Q. K., E. Vincent et R. Gribonval. 2010, «Under-determined reverberant audio source separation using a full-rank spatial covariance model», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n° 7, p. 1830–1840. I, II

Durrieu, J.-L., B. David et G. Richard. 2011, «A musically motivated mid-level representation for pitch estimation and musical audio source separation», *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, n° 6, p. 1180–1191. 28, 70

Durrieu, J.-L., G. Richard et B. David. 2009, «An iterative approach to monoral musical mixture de-soloing», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 105–108. 23, 26, 27

Durrieu, J.-L., G. Richard, B. David et C. Févotte. 2010, «Source/filter model for unsupervised main melody extraction from polyphonic audio signals», *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, n° 3, p. 564–575. 27

Efron, B. 1983, «Estimating the error rate of a prediction rule : improvement on cross-validation», *Journal of the American Statistical Association*, vol. 78, n° 382, p. 316–331. 33

Efron, B. et R. J. Tibshirani. 1994, *An introduction to the bootstrap*, CRC press. 33

Eggert, J. et E. Korner. 2004, «Sparse coding and NMF», dans *Proc. of IEEE Int. Joint Conference on Neural Networks*, vol. 4, p. 2529–2533. 20

Eggink, J. et G. J. Brown. 2003, «Application of missing feature theory to the recognition of musical instruments in polyphonic audio», dans *Proc. of International Symposium on Music Information Retrieval (ISMIR)*, p. 125–131. 13

Ellis, D. et N. Morgan. 1999, «Size matters : An empirical study of neural network training for large vocabulary continuous speech recognition», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, p. 1013–1016. 150

Ellis, D. P. W. 1996, *Prediction-driven computational auditory scene analysis*, thèse de doctorat, Massachusetts Institute of Technology. 13

Emiya, V., E. Vincent, N. Harlander et V. Hohmann. 2011, «Subjective and objective quality assessment of audio source separation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, n° 7, p. 2046–2057. 24

Ephraim, Y. 1992, «Statistical-model-based speech enhancement systems», *Proc. of the IEEE*, vol. 80, n° 10, p. 1526–1555. 25

Ephraim, Y. et H. L. Van Trees. 1995, «A signal subspace approach for speech enhancement», *IEEE Transactions on Speech and Audio Processing*, vol. 3, n° 4, p. 251–266. 25

Ewert, S. et M. Müller. 2012, «Using score-informed constraints for nmf-based source separation», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 129–132. 20

Ewert, S., M. Müller et M. Sandler. 2013, «Efficient data adaption for musical source separation methods based on parametric models», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 46–50. 20

Fastl, H. et E. Zwicker. 2007, *Psychoacoustics : facts and models*, vol. 22, Springer Science & Business Media. I

Févotte, C., N. Bertin et J.-L. Durrieu. 2009, «Nonnegative matrix factorization with the Itakura-Saito divergence : With application to music analysis», *Neural Computation*, vol. 21, n° 3, p. 793–830. 17, 20, 21, 93

Févotte, C. et J. Idier. 2011, «Algorithms for nonnegative matrix factorization with the β-divergence», *Neural Computation*, vol. 23, n° 9, p. 2421–2456. 17, 18

Fiscus, J. G. 1997, «A post-processing system to yield reduced word error rates : Recognizer output voting error reduction (ROVER)», dans *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding*, p. 347–354. 38, 45

FitzGerald, D. 2004, *Automatic Drum Transcription and Source Separation*, thèse de doctorat, Dublin Institute of Technology. 14

FitzGerald, D., M. Cranitch et E. Coyle. 2005, «Non-negative tensor factorisation for sound source separation», dans *Proc. of Irish Signals and Systems Conference*, p. 8–12. 21

FitzGerald, D., M. Cranitch et E. Coyle. 2008, «Extended nonnegative tensor factorisation models for musical sound source separation», *Computational Intelligence and Neuroscience*, p. 15, Article ID 872 425. 23, 27

Foucard, R., S. Essid, M. Lagrange et G. Richard. 2011, «Multi-scale temporal fusion by boosting for music classification», dans *Proc. of International Symposium on Music Information Retrieval (ISMIR)*, p. 1873–1876. 39

Fox, B. et B. Pardo. 2007, «Towards a model of perceived quality of blind audio source separation», dans *Proc. of IEEE Int. Conf. on Multimedia Expo*, p. 1898–1901. 24

Freund, Y. et R. E. Schapire. 1997, «A decision-theoretic generalization of on-line learning and an application to boosting», *Journal of Computer and System Sciences*, vol. 55, n° 1, p. 119–139. 39

Gannot, S., D. Burshtein et E. Weinstein. 2001, «Signal enhancement using beamforming and nonstationarity with applications to speech», *IEEE Transactions on Signal Processing*, vol. 49, n° 8, p. 1614–1626. 25

Garofalo, J., D. Graff, D. Paul et D. Pallett. 2007, «CSR-I (WSJ0) Complete», *Linguistic Data Consortium, Philadelphia*. 53

Geiger, J. T., F. Weninger, A. Hurmalainen, J. F. Gemmeke, M. Wöllmer, B. Schuller, G. Rigoll et T. Virtanen. 2013, «The TUM + TUT + KUL approach to the 2nd CHiME challenge : Multi-stream ASR exploiting BLSTM networks and sparse NMF», dans *Proc. of CHiME*, p. 25–30. 26, 53

Geman, S., E. Bienenstock et R. Doursat. 1992, «Neural networks and the bias/variance dilemma», *Neural computation*, vol. 4, n° 1, p. 1–58. 147

Gemmeke, J. F., A. Hurmalainen et T. Virtanen. 2013, «HMM-regularization for NMF-based noise robust ASR», dans *Proc. of CHiME*, p. 47–52. 26

Glorot, X., A. Bordes et Y. Bengio. 2011, «Deep sparse rectifier networks», dans *Proc. of International Conference on Artificial Intelligence and Statistics*, vol. 15, p. 315–323. 146

Godsmark, D. et G. J. Brown. 1999, «A blackboard architecture for computational auditory scene analysis», *Speech Communication*, vol. 27, n° 3, p. 351–366. 13

Goodfellow, I. J., D. Warde-Farley, M. Mirza, A. Courville et Y. Bengio. 2013, «Maxout networks», *arXiv preprint arXiv :1302.4389*. 147

Grais, E. M., M. U. Sen et H. Erdogan. 2014, «Deep neural networks for single channel source separation», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 3734–3738. 14

Gribonval, R. et S. Lesage. 2006, «A survey of sparse component analysis for blind source separation : principles, perspectives, and new challenges», dans *Proc. of European Symposium on Artificial Neural Networks*, p. 323–330. 13, 14

Han, J. et C.-W. Chen. 2011, «Improving melody extraction using probabilistic latent component analysis», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 33–36. 27

Hao, J., H. Attias, S. Nagarajan, T.-W. Lee et T. J. Sejnowski. 2009, «Speech enhancement, gain, and noise spectrum adaptation using approximate Bayesian estimation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, n° 1, p. 24–37. 26

Hennequin, R. 2010, *Décomposition de spectrogrammes musicaux informée par des modèles de synthèse spectrale*, thèse de doctorat, Telecom ParisTech. 15, 19

Hennequin, R., R. Badeau et B. David. 2010, «Time-dependent parametric and harmonic templates in non-negative matrix factorization», dans *Proc. of Conf. on Digital Audio Effects (DAFx)*, p. 246–253. 20

Hennequin, R., B. David et R. Badeau. 2011, «Score informed audio source separation using a parametric model of non-negative spectrogram», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 45–48. 20

Hinton, G., L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath et al.. 2012a, «Deep neural networks for acoustic modeling in speech recognition : The shared views of four research groups», *IEEE Signal Processing Magazine*, vol. 29, n° 6, p. 82–97. 139, 146

Hinton, G., S. Osindero et Y.-W. Teh. 2006, «A fast learning algorithm for deep belief nets», *Neural Computation*, vol. 18, n° 7, p. 1527–1554. 146

Hinton, G. E., N. Srivastava, A. Krizhevsky, I. Sutskever et R. R. Salakhutdinov. 2012b, «Improving neural networks by preventing co-adaptation of feature detectors», *arXiv preprint arXiv :1207.0580*. 146

Hlawatsch, F. et G. F. Boudreaux-Bartels. 1992, «Linear and quadratic time-frequency signal representations», *IEEE Signal Processing Magazine*, vol. 9, n° 2, p. 21–67. I

Hoeting, J. A., D. Madigan, A. E. Raftery et C. T. Volinsky. 1999, «Bayesian model averaging : a tutorial», *Statistical science*, p. 382–401. 40

Hoffman, M., D. M. Blei et P. R. Cook. 2010, «Bayesian nonparametric matrix factorization for recorded music», dans *Proc. of International Conference on Machine Learning (ICML)*, p. 439–446. 43, 95, 96, 98, XIV

Hoffman, M. D. 2014a, «A problem with (and fix for) variational bayesian NMF», dans *Proc. of IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, p. 527–531. 108

Hoffman, M. D. 2014b, «Stochastic structured mean-field variational inference», *arXiv preprint arXiv :1404.4114*. 137, 165

Hoyer, P. O. 2002, «Non-negative sparse coding», dans *Proc. of the IEEE Workshop on Neural Networks for Signal Processing*, p. 557–565. 14

Hoyer, P. O. 2004, «Non-negative matrix factorization with sparseness constraints», *The Journal of Machine Learning Research*, vol. 5, p. 1457–1469. 20

Hsu, C.-L. et J.-S. R. Jang. 2010, «On the improvement of singing voice separation for monaural recordings using the mir-1k dataset», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n° 2, p. 310–319. 28

Huang, G., S. Song, J. N. D. Gupta et C. Wu. 2014a, «Semi-supervised and unsupervised extreme learning machines», *IEEE Transactions on Cybernetics*, vol. 44, n° 12, p. 2405–2417. 166

Huang, P.-S., S. D. Chen, P. Smaragdis et M. Hasegawa-Johnson. 2012, «Singing-voice separation from monaural recordings using robust principal component analysis», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 57–60. 28, 30, 70, 71

Huang, P.-S., M. Kim, M. Hasegawa-Johnson et P. Smaragdis. 2014b, «Deep learning for monaural speech separation», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 1562–1566. 14, 139, 149

Huang, P.-S., M. Kim, M. Hasegawa-Johnson et P. Smaragdis. 2014c, «Singing-voice separation from monaural recordings using deep recurrent neural networks», *Proc. of International Symposium on Music Information Retrieval (ISMIR)*. 14, 139

Huang, Y., M. Slaney, M. L. Seltzer et Y. Gong. 2014d, «Towards better performance with heterogeneous training data in acoustic modeling using deep neural networks», dans *Proc. of Interspeech*, p. 845–849. 166

Huber, R. et B. Kollmeier. 2006, «PEMO-Q—A New Method for Objective Audio Quality Assessment Using a Model of Auditory Perception», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, n° 6, p. 1902–1911. 24

Hurvich, C. M. et C.-L. Tsai. 1993, «A corrected akaike information criterion for vector autoregressive model selection», *Journal of time series analysis*, vol. 14, n° 3, p. 271–279. 36

Hyvarinen, A. 1999, «Fast and robust fixed-point algorithms for independent component analysis», *IEEE Transactions on Neural Networks*, vol. 10, n° 3, p. 626–634. 14

Hyvärinen, A., J. Karhunen et E. Oja. 2004, *Independent component analysis*, vol. 46, John Wiley & Sons. 12

Itakura, F. et S. Saito. 1968, «Analysis synthesis telephony based on the maximum likelihood method», dans *Proc. of the International Congress on Acoustics*, vol. 17, p. C17–C20. 17

Jacobs, R. A. 1988, «Increased rates of convergence through learning rate adaptation», *IEEE Transactions on Neural Networks*, vol. 1, n° 4, p. 295–307. 148

Jaureguiberry, X., P. Leveau, S. Maller et J. J. Burred. 2011, «Adaptation of source-specific dictionaries in non-negative matrix factorization for source separation», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 5–8. 20

Jaureguiberry, X., G. Richard, P. Leveau, R. Hennequin et E. Vincent. 2013, «Introducing a simple fusion framework for audio source separation», dans *Proc. of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, p. 1–6. 5, 45, 48, 79

Jaureguiberry, X., E. Vincent et G. Richard. 2014a, «Multiple-order non-negative matrix factorization for speech enhancement», dans *Proc. of Interspeech*, p. 4. 6

Jaureguiberry, X., E. Vincent et G. Richard. 2014b, «Variational Bayesian model averaging for audio source separation», dans *Proc. of IEEE Statistical Signal Processing Workshop (SSP)*, p. 33–36. 5, 6

Jaureguiberry, X., E. Vincent et G. Richard. 2015, «Fusion methods for audio source separation», Submitted to *IEEE Transactions on Audio, Speech, and Language Processing*. 5, 6, 81

Joder, C., F. Weninger, D. Virette et B. Schuller. 2013, «A comparative study on sparsity penalties for nmf-based speech separation : Beyond lp-norms», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 858–862. 20

Jolliffe, I. 2002, *Principal component analysis*, Wiley Online Library. 12

Jørgensen, B. 1982, *Statistical properties of the generalized inverse Gaussian distribution*, Springer. 101

Jourjine, A., S. Rickard et O. Yilmaz. 2000, «Blind separation of disjoint orthogonal signals : Demixing N sources from 2 mixtures», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 5, p. 2985–2988. 13, 27, 44

Kashino, K., K. Nakadai, T. Kinoshita et H. Tanaka. 1995, «Application of Bayesian probability network to music scene analysis», *Computational auditory scene analysis*, vol. 1, nº 998, p. 1–15. 13

Kass, R. E. et A. E. Raftery. 1995, «Bayes factors», *Journal of the American Statistical Association*, vol. 90, nº 430, p. 773–795. 34

Katahira, K., K. Watanabe et M. Okada. 2008, «Deterministic annealing variant of variational Bayes method», *Journal of Physics : Conference Series*, vol. 95, nº 1. 119

Kinoshita, T., S. Sakai et H. Tanaka. 1999, «Musical sound source identification based on frequency component adaptation», dans *Proc. of IJCAI worshop on CASA*, p. 18–24. 13

Kittler, J., M. Hatef, R. P. W. Duin et J. Matas. 1998, «On combining classifiers», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, nº 3, p. 226–239. 49

Klatt, D. H. et L. C. Klatt. 1990, «Analysis, synthesis, and perception of voice quality variations among female and male talkers», *Journal of the Acoustical Society of America*, vol. 87, nº 2, p. 820–857. 27, 29, 57, 71

Kompass, R. 2007, «A generalized divergence measure for nonnegative matrix factorization», *Neural computation*, vol. 19, nº 3, p. 780–791. 18, 164

Koren, Y. «The BellKor solution to the Netflix Grand Prize», http://www.netflixprize.com/assets/GrandPrize2009_BPC_BellKor.pdf. 39

Kullback, S. et R. A. Leibler. 1951, «On information and sufficiency», *The annals of mathematical statistics*, p. 79–86. 17

Kuncheva, L. I., J. C. Bezdek et R. P. Duin. 2001, «Decision templates for multiple classifier fusion : an experimental comparison», *Pattern recognition*, vol. 34, nº 2, p. 299–314. 38

Le Roux, J. et J. R. Hershey. 2012, «Indirect model-based speech enhancement», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 4045–4048. 45

Le Roux, J., S. Watanabe et J. R. Hershey. 2013, «Ensemble learning for speech enhancement», dans *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, p. 1–4. 44, 45

Lebarbier, E. et T. Mary-Huard. 2006, «Une introduction au critère BIC : Fondements théoriques et interprétation», *Journal de la Société française de statistique*, vol. 147, nº 1, p. 39–57. 35

LeCun, Y., B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard et L. D. Jackel. 1989, «Backpropagation applied to handwritten zip code recognition», *Neural computation*, vol. 1, nº 4, p. 541–551. 146

Lee, D. D. et H. S. Seung. 1999, «Learning the parts of objects by non-negative matrix factorization», *Nature*, vol. 401, n° 6755, p. 788–791. 15, 17, 18, 22

Lee, D. D. et H. S. Seung. 2001, «Algorithms for non-negative matrix factorization», *Advances in Neural Information Processing Systems*, vol. 13, p. 556–562. 28, 93

Li, J., D. Yu, J.-T. Huang et Y. Gong. 2012, «Improving wideband speech recognition using mixed-bandwidth training data in CD-DNN-HMM», dans *Proc. of SLT*, p. 131–136. 149

Li, Y. et D. L. Wang. 2007, «Separation of singing voice from music accompaniment for monaural recordings», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, n° 4, p. 1475–1487. 28

Li, Y. et D. L. Wang. 2009, «On the optimality of ideal binary time–frequency masks», *Speech Communication*, vol. 51, n° 3, p. 230–239. 49

Lin, C.-J. 2007, «Projected gradient methods for nonnegative matrix factorization», *Neural computation*, vol. 19, n° 10, p. 2756–2779. 18

Liutkus, A., D. Fitzgerald, Z. Rafii, B. Pardo et L. Daudet. 2014, «Kernel additive models for source separation», *IEEE Transactions on Signal Processing*, vol. 62, n° 16, p. 4298–4310. 28, 31, 32, 69, 70, 71

Liutkus, A., Z. Rafii, R. Badeau, B. Pardo et G. Richard. 2012, «Adaptive filtering for music/voice separation exploiting the repeating musical structure», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 53–56. 27

Loizou, P. C. 2013, *Speech enhancement : theory and practice*, CRC press. 25

Maas, A. L., Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen et A. Y. Ng. 2012, «Recurrent neural networks for noise reduction in robust ASR», dans *Proc. of Interspeech*, p. 22–25. 139

MacKay, D. J. C. 1995, «Probable networks and plausible predictions-a review of practical Bayesian methods for supervised neural networks», *Network : Computation in Neural Systems*, vol. 6, n° 3, p. 469–505. 43

Mandel, M. I., R. J. Weiss et D. P. W. Ellis. 2010, «Model-based expectation-maximization source separation and localization», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n° 2, p. 382–394. 13

Mathieu, B., S. Essid, T. Fillon, J. Prado et G. Richard. 2010, «YAAFE, an easy to use and efficient audio feature extraction software», dans *Proc. of International Symposium on Music Information Retrieval (ISMIR)*, p. 441–446. 139

Miao, Y., F. Metze et S. Rawat. 2013, «Deep maxout networks for low-resource speech recognition», dans *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, p. 398–403. 147

Minka, T. P. 2002, «Bayesian model averaging is not model combination», http://research.microsoft.com/en-us/um/people/minka/papers/minka-bma-isnt-mc.pdf. 40

Mohammadiha, N., J. Taghia et A. Leijon. 2012, «Single channel speech enhancement using Bayesian NMF with recursive temporal updates of prior distributions», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 4561–4564. 26, 53

Monteith, K., J. L. Carroll, K. Seppi et T. Martinez. 2011, «Turning Bayesian model averaging into Bayesian model combination», dans *Proc. of IEEE Int. Joint Conference on Neural Networks*, p. 2657–2663. 41

Moreno, P. J., B. Raj et R. M. Stern. 1996, «A vector Taylor series approach for environment-independent speech recognition», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, p. 733–736. 45

Moritz, N., M. R. Schädler, K. Adiloğlu, B. T. Meyer, T. Jürgens, T. Gerkmann, B. Kollmeier, S. Doclo et S. Goetze. 2013, «Noise robust distant automatic speech recognition utilizing NMF based source separation and auditory feature extraction», dans *Proc. of CHiME*, p. 1–6. 26, 53

Mowlaee, P., R. Saeidi, M. G. Christensen, Z.-H. Tan, T. Kinnunen, P. Franti et S. H. Jensen. 2012, «A joint approach for single-channel speaker identification and speech separation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, n° 9, p. 2586–2601. 26

Nair, V. et G. E. Hinton. 2010, «Rectified linear units improve restricted Boltzmann machines», dans *Proc. of International Conference on Machine Learning (ICML)*, p. 807–814. 146

Nakatani, T. 2002, *Computational auditory scene analysis based on residue-driven architecture and its application to mixed speech recognition*, thèse de doctorat, Kyoto University. 13

Narayanan, A. et D. L. Wang. 2013, «Ideal ratio mask estimation using deep neural networks for robust speech recognition», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 7092–7096. 14, 139, 149

Neeser, F. D. et J. L. Massey. 1993, «Proper complex random processes with applications to information theory», *IEEE Transactions on Information Theory*, vol. 39, n° 4, p. 1293–1302. 21

Niedermayer, B. 2008, «Non-negative matrix division for the automatic transcription of polyphonic music.», dans *Proc. of International Symposium on Music Information Retrieval (ISMIR)*, p. 544–549. 20

Ochiai, K., H. Kameoka et S. Sagayama. 2012, «Explicit beat structure modeling for non-negative matrix factorization-based multipitch analysis», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 133–136. 20

Ono, N., Z. Koldovský, S. Miyabe et N. Ito. 2013, «The 2013 signal separation evaluation campaign», dans *Proc. of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, p. 1–6. 42

Ozerov, A. et C. Févotte. 2010, «Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n° 3, p. 550–563. 15, 21, 22, 29, 44

Ozerov, A., P. Philippe, F. Bimbot et R. Gribonval. 2007, «Adaptation of Bayesian models for single-channel source separation and its application to voice/music separation in popular songs», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, n° 5, p. 1564–1578. 44

Ozerov, A., E. Vincent et F. Bimbot. 2012, «A general flexible framework for the handling of prior information in audio source separation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, nº 4, p. 1118–1133. 22, 26, 27, 55, 94

Parry, R. M. et I. Essa. 2006, «Estimating the spatial position of spectral components in audio», dans *Proc. of International Conference on Independent Component Analysis and Blind Signal Separation*, Springer, p. 666–673. 21

Parvaix, M. et L. Girin. 2011, «Informed source separation of linear instantaneous under-determined audio mixtures by source index embedding», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, nº 6, p. 1721–1733. 19

Plumbley, M. D., T. Blumensath, L. Daudet, R. Gribonval et M. E. Davies. 2010, «Sparse representations in audio and music : from coding to source separation», *Proc. of the IEEE*, vol. 98, nº 6, p. 995–1005. 13

Plumbley, M. D. et E. Oja. 2004, «A "nonnegative PCA" algorithm for independent component analysis», *IEEE Transactions on Neural Networks*, vol. 15, nº 1, p. 66–76. 14

Rabiner, L. R. et B.-H. Juang. 1993, *Fundamentals of speech recognition*, vol. 14, PTR Prentice Hall Englewood Cliffs. 139

Rafii, Z. et B. Pardo. 2011, «A simple music/voice separation method based on the extraction of the repeating musical structure», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 221–224. 31

Rafii, Z. et B. Pardo. 2012, «Music/voice separation using the similarity matrix», dans *Proc. of International Symposium on Music Information Retrieval (ISMIR)*, p. 583–588. 27, 30, 70, 71

Raftery, A. E., D. Madigan et J. A. Hoeting. 1997, «Bayesian model averaging for linear regression models», *Journal of the American Statistical Association*, vol. 92, nº 437, p. 179–191. 40

Raj, B., R. Singh et T. Virtanen. 2011, «Phoneme-dependent NMF for speech enhancement in monaural mixtures», dans *Proc. of Interspeech*, p. 1217–1220. 26, 53

Raphael, C. 2008, «A classifier-based approach to score-guided source separation of musical audio», *Computer Music Journal*, vol. 32, nº 1, p. 51–59. 48

Reddy, A. M. et B. Raj. 2007, «Soft mask methods for single-channel speaker separation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, nº 6, p. 1766–1776. 49

Refaeilzadeh, P., L. Tang et H. Liu. 2009, «Cross-validation», dans *Encyclopedia of database systems*, Springer, p. 532–538. 33

Rickard, S. et O. Yilmaz. 2002, «On the approximate W-disjoint orthogonality of speech», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, p. I–529–I–532. 13, 27

Rissanen, J. 1978, «Modeling by shortest data description», *Automatica*, vol. 14, nº 5, p. 465–471. 36

Rix, A. W., J. G. Beerends, M. P. Hollier et A. P. Hekstra. 2001, «Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, p. 749–752. 24

Roman, N., D. L. Wang et G. J. Brown. 2003, «Speech segregation based on sound localization», *The Journal of the Acoustical Society of America*, vol. 114, n° 4, p. 2236–2252. 13

Rosenblatt, F. 1958, «The perceptron : a probabilistic model for information storage and organization in the brain», *Psychological review*, vol. 65, n° 6, p. 386–408. 140

Rumelhart, D. E., G. E. Hinton et R. J. Williams. 1986, «Learning representations by back-propagating errors», *Nature*, vol. 323, p. 533–536. 141, 146

Sakuraba, Y. et H. G. Okuno. 2003, «Note recognition of polyphonic music by using timbre similarity and direction proximity», dans *Proc. of the International Computer Music Conference*, p. 167–170. 13

Salaün, Y., E. Vincent, N. Bertin, N. Souviraà-Labastie, X. Jaureguiberry, D. T. Tran et F. Bimbot. 2014, «The Flexible Audio Source Separation Toolbox version 2.0», dans *ICASSP*. 55

Saul, L. K. et M. I. Jordan. 1996, «Exploiting tractable substructures in intractable networks», *Advances in neural information processing systems*, p. 486–492. 107

Schapire, R. E. 1990, «The strength of weak learnability», *Machine learning*, vol. 5, n° 2, p. 197–227. 38

Schmidt, M. N., O. Winther et L. K. Hansen. 2009, «Bayesian non-negative matrix factorization», dans *Prof. of International Conefrence on Independent Component Analysis and Signal Separation*, Springer, p. 540–547. 42

Schwarz, G. 1978, «Estimating the dimension of a model», *The annals of statistics*, vol. 6, n° 2, p. 461–464. 35

Seltzer, M. L., D. Yu et Y. Wang. 2013, «An investigation of deep neural networks for noise robust speech recognition», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 7398–7402. 149

Shafer, G. 1976, *A mathematical theory of evidence*, Princeton University Press. 37

Shashanka, M., B. Raj et P. Smaragdis. 2008, «Sparse overcomplete latent variable decomposition of counts data», dans *Advances in neural information processing systems*, p. 1313–1320. 20

Shinozaki, T. et S. Watanabe. 2015, «Structure discovery of deep neural network based on evolutionary algorithms», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 164

Slaney, M. et R. F. Lyon. 1990, «A perceptual pitch detector», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 357–360. 13

Smaragdis, P. 2001, *Redundancy reduction for computational audition, a unifying approach*, thèse de doctorat, Massachusetts Institute of Technology. 14

Smaragdis, P. 2007, «Convolutive speech bases and their application to supervised speech separation», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, n° 1, p. 1–12. 42

Smaragdis, P. et J. C. Brown. 2003, «Non-negative matrix factorization for polyphonic music transcription», dans *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, p. 177–180. 14

Smaragdis, P., B. Raj et M. Shashanka. 2009, «Missing data imputation for spectral audio signals», dans *Proc. of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, p. 1–6. 15, 19

Socher, R., J. Pennington, E. H. Huang, A. Y. Ng et C. D. Manning. 2011, «Semi-supervised recursive autoencoders for predicting sentiment distributions», dans *Proc. of the Conference on Empirical Methods in Natural Language Processing*, p. 151–161. 151

Sprechmann, P., J. Bruna et Y. LeCun. 2015, «Audio source separation with discriminative scattering networks», dans *Proc. of International Conference on Learning Representations (ICLR)*. 164

Srinivasan, S., J. Samuelsson et W. B. Kleijn. 2006, «Codebook driven short-term predictor parameter estimation for speech enhancement», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, n° 1, p. 163–176. 26

Starck, J.-L., Y. Moudden, J. Bobin, M. Elad et D. L. Donoho. 2005, «Morphological component analysis», dans *Optics & Photonics*, p. 59 140Q–1–59 140Q–15. 14

Stoica, P. et Y. Selen. 2004, «Model-order selection : a review of information criterion rules», *Signal Processing Magazine*, vol. 21, n° 4, p. 36–47. 34, 35, 36

Stone, C. J. 1977, «Consistent nonparametric regression», *The annals of statistics*, p. 595–620. 31

Stone, J. V., J. Porrill, C. Buchel et K. Friston. 1999, «Spatial, temporal and spatiotemporal independent component analysis of fMRI data», dans *Proc. of Leeds Statistical Research Workshop*, p. 23–28. 14

Sturmel, N., A. Liutkus, J. Pinel, L. Girin, S. Marchand, G. Richard, R. Badeau et L. Daudet. 2012, «Linear mixing models for active listening of music productions in realistic studio conditions», dans *Audio Engineering Society Convention*. 10, 11

Swietojanski, P., J. Li et J.-T. Huang. 2014, «Investigation of maxout networks for speech recognition», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 7649–7653. 147

Tan, V. Y. F. et C. Févotte. 2009, «Automatic relevance determination in nonnegative matrix factorization», dans *Proc. of Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS)*. URL https://hal.inria.fr/inria-00369376. 43

Tan, V. Y. F. et C. Févotte. 2013, «Automatic relevance determination in nonnegative matrix factorization with the β-divergence», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, n° 7, p. 1592–1605. 42, 43

Tax, D. M. J. et R. P. W. Duin. 2002, «Using two-class classifiers for multiclass classification», dans *Proc. of 16th IEEE International Conference on Pattern Recognition*, vol. 2, p. 124–127. 49

Theis, F. J. 2006, «Towards a general independent subspace analysis», dans *Advances in Neural Information Processing Systems*, p. 1361–1368. 14

Thiede, T., W. C. Treurniet, R. Bitto, C. Schmidmer, T. Sporer, J. G. Beerends et C. Colomes. 2000, «PEAQ-The ITU standard for objective measurement of perceived audio quality», *Journal of the Audio Engineering Society*, vol. 48, nº 1/2, p. 3–29. 24

Tipping, M. E. 2001, «Sparse Bayesian learning and the relevance vector machine», *The Journal of Machine Learning Research*, vol. 1, p. 211–244. 43

Vega-Pons, S. et J. Ruiz-Shulcloper. 2011, «A survey of clustering ensemble algorithms», *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, nº 03, p. 337–372. 37

Vincent, E. 2006, «Musical source separation using time-frequency source priors», *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, nº 1, p. 91–98. 12, 15

Vincent, E. 2012, «Improved perceptual metrics for the evaluation of audio source separation», dans *Proc. of Int. Conf. on Latent Variable Analysis and Signal Separation (LVA/ICA)*, Springer, p. 430–437. 24, 164

Vincent, E., S. Arberet et R. Gribonval. 2009, «Underdetermined instantaneous audio source separation via local Gaussian modeling», dans *Prof. of International Conefrence on Independent Component Analysis and Signal Separation*, Springer, p. 775–782. 44

Vincent, E., J. Barker, S. Watanabe, J. Le Roux, F. Nesta et M. Matassoni. 2013a, «The second 'CHiME' speech separation and recognition challenge : An overview of challenge systems and outcomes», dans *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, p. 162–167. 26, 53

Vincent, E., J. Barker, S. Watanabe, J. Le Roux, F. Nesta et M. Matassoni. 2013b, «The second 'CHiME' speech separation and recognition challenge : datasets, tasks and baselines», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 126–130. 42, 53

Vincent, E., R. Gribonval et C. Févotte. 2006, «Performance measurement in blind audio source separation», *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, nº 4, p. 1462–1469. 24, 50, 164

Vincent, E., M. G. Jafari, S. A. Abdallah, M. D. Plumbley et M. E. Davies. 2010a, «Probabilistic modeling paradigms for audio source separation», *Machine Audition : Principles, Algorithms and Systems*, p. 162–185. 11

Vincent, E., H. Sawada, P. Bofill, S. Makino et J. P. Rosca. 2007, «First stereo audio source separation evaluation campaign : data, algorithms and results», dans *Proc. of Int. Conf. on Independent Component Analysis and Blind Signal Separation (ICA)*, Springer, p. 552–559. 24

Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio et P.-A. Manzagol. 2010b, «Stacked denoising autoencoders : Learning useful representations in a deep network with a local denoising criterion», *The Journal of Machine Learning Research*, p. 3371–3408. 146

Virtanen, T. 2003, «Sound source separation using sparse coding with temporal continuity objective», dans *Proc. of the International Computer Music Conference*, vol. 3, p. 231–234. 14

Virtanen, T. 2007, «Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria», *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, n° 3, p. 1066–1074. 20

Virtanen, T., A. T. Cemgil et S. Godsill. 2008a, «Bayesian extensions to non-negative matrix factorisation for audio signal modelling», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 1825–1828. 20, 93

Virtanen, T., A. Mesaros et M. Ryynänen. 2008b, «Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music», dans *Proc. of Interspeech*, p. 17–22. 28

Viste, H. et G. Evangelista. 2003, «On the use of spatial cues to improve binaural source separation», dans *Proc. of Conf. on Digital Audio Effects (DAFx)*, p. 209–213. 13

Wan, L., M. Zeiler, S. Zhang, Y. LeCun et R. Fergus. 2013, «Regularization of neural networks using dropconnect», dans *Proc. of International Conference on Machine Learning (ICML)*, p. 1058–1066. 146

Wang, A. 2006, «The Shazam music recognition service», *Communications of the ACM*, vol. 49, n° 8, p. 44–48. 3

Wang, W. et X. Zou. 2008, «Non-negative matrix factorization based on projected nonlinear conjugate gradient algorithm», dans *Proc. of ICA Research Network International Workshop*, p. 5–8. 18

Weninger, F., J. Le Roux, J. R. Hershey et B. Schuller. 2014, «Discriminatively trained recurrent neural networks for single-channel speech separation», dans *Proc. of IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, p. 577–581. 14, 149, 160, 161, 164

Wiener, N. 1949, *Extrapolation, interpolation, and smoothing of stationary time series*, MIT press. 23

Winther, I. et K. B. Petersen. 2007, «Bayesian independent component analysis : Variational methods and non-negative decompositions», *Digital Signal Processing*, vol. 17, n° 5, p. 858–872. 42

Wolpert, D. H. 1992, «Stacked generalization», *Neural networks*, vol. 5, n° 2, p. 241–259. 39

Wright, S. J. et J. Nocedal. 1999, *Numerical optimization*, Springer. 82

Xu, L., A. Krzyzak et C. Y. Suen. 1992, «Methods of combining multiple classifiers and their applications to handwriting recognition», *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, n° 3, p. 418–435. 38, 49

Zdunek, R. et A. Cichocki. 2007, «Nonnegative matrix factorization with constrained second-order optimization», *Signal Processing*, vol. 87, n° 8, p. 1904–1916. 18

Zeiler, M. D. 2012, «ADADELTA : an adaptive learning rate method», *arXiv preprint arXiv :1212.5701*. 148

Zeiler, M. D., M. Ranzato, R. Monga, M. Mao, K. Yang, Q. V. Le, P. Nguyen, A. Senior, V. Vanhoucke, J. Dean et al.. 2013, «On rectified linear units for speech processing», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 3517–3521. 147, 153

Zhang, X., J. Trmal, D. Povey et S. Khudanpur. 2014, «Improving deep neural network acoustic models using generalized maxout networks», dans *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 215–219. 147

Zhang, Y. et Y. Fang. 2007, «A NMF algorithm for blind separation of uncorrelated signals», dans *IEEE International Conference on Wavelet Analysis and Pattern Recognition*, vol. 3, p. 999–1003. 20

Zimmermann, H.-J. 2001, *Fuzzy set theory—and its applications*, Springer Science & Business Media. 37

Annexe A

Représentations temps-fréquence

Pour la plupart des méthodes de séparation de sources que nous avons évoquées dans ce travail, il convient dans un premier temps de choisir la représentation temps-fréquence qui sera employée. Deux familles principales de représentations existent : les *représentations temps-fréquence linéaires* et les *représentations temps-fréquence quadratiques* [Hlawatsch et Boudreaux-Bartels, 1992]. Nous en donnons un aperçu ci-après.

A.1 Représentations temps-fréquence linéaires

Les représentations temps-fréquence linéaires sont certainement les plus employées et la célèbre *transformée de Fourier à court-terme* (TFCT) est très souvent préférée dans les applications audio. Le principe d'une représentation linéaire consiste à représentater un signal $\mathbf{x}(t)$ à I canaux par un vecteur $\mathbf{x}_{fn}$ de I valeurs complexes en chaque point temps-fréquence (f, n). Les coefficients de la TFCT $\mathbf{X} = \{\mathbf{x}_{fn}\}_{n=1..N}^{f=1..F}$ sont obtenus selon

$$\forall f, n, \quad \mathbf{x}_{fn} = \sum_{l=1}^{L} \mathbf{x}(l + (n-1)h) w(l) \, \mathrm{e}^{-i2\pi(f-1)(l-1)/L} \tag{A.1}$$

où $\mathbf{w} = (w(1), ..., w(l), ..., w(L))^\intercal$ est une fenêtre d'analyse de longueur L et h est le pas séparant deux fenêtres adjacentes ($L - h$ est donc la longueur de recouvrement de ces deux fenêtres). La TFCT a pour principal avantage d'avoir été très étudiée, d'être rapide à calculer et d'être facilement interprétable. Son module et son module au carré forment respectivement les spectrogrammes d'amplitude $|\mathbf{X}| = \{|\mathbf{x}_{fn}|\}_{n=1..N}^{f=1..F}$ et de puissance $|\mathbf{X}|^{\cdot 2} = \{|\mathbf{x}_{fn}|^{\cdot 2}\}_{n=1..N}^{f=1..F}$.

Certains de ses défauts sont également bien connus. L'un d'entre eux est l'emploi d'une échelle de fréquence linéaire qui induit une résolution fréquentielle identique entre basses et hautes fréquences, alors qu'en audio l'essentiel de l'énergie se situe dans les basses fréquences. Pour y remédier, il est possible de considérer plutôt une échelle de fréquence non-linéaire. C'est le cas notamment de la représentation ERB (pour *Equivalent Rectangular Bandwidth*) qui emploie l'échelle ERB [Fastl et Zwicker, 2007] modélisant la perception de l'oreille humaine. La fréquence f_{ERB} sur l'échelle ERB correspondant à la fréquence f_{Hz} sur l'échelle linéaire est obtenue par

$$f_{\mathsf{ERB}} = 9.26 \log(0.00437 f_{\mathsf{Hz}} + 1). \tag{A.2}$$

Ainsi, pour un même nombre de bandes de fréquence F, la représentation temps-fréquence ERB présente une meilleure résolution dans les basses fréquences que la TFCT, mais une résolution plus faible dans les hautes fréquences.

En pratique, la transformée ERB est implémentée à l'aide d'un banc de filtres. Les F filtres sont des fenêtres de Hann modulés par une exponentielle complexe [Duong et al., 2010]. Les

fréquences centrales des filtres sont linéairement espacées sur l'échelle ERB entre 0 Hz et $f_s/2$. La largeur de la fenêtre est choisie de sorte que le lobe principal du filtre soit quatre fois plus large que la distance entre les fréquences centrales des deux filtres adjacents.

A.2 Représentations temps-fréquence quadratiques

Plutôt que de représenter le signal comme un vecteur $\mathbf{x}_{fn}$ de coefficients complexes, les représentations quadratiques représentent un signal multicanal par une matrice de covariance en chaque point temps-fréquence. Ainsi, pour un signal $\mathbf{x}(t)$ formé de I canaux, chaque point temps-fréquence est décrit par une matrice de taille $I \times I$

$$\hat{\mathbf{R}}_{\mathbf{x}_{fn}} = \hat{\mathbb{E}}[\mathbf{x}_{fn}\mathbf{x}_{fn}^H] \tag{A.3}$$

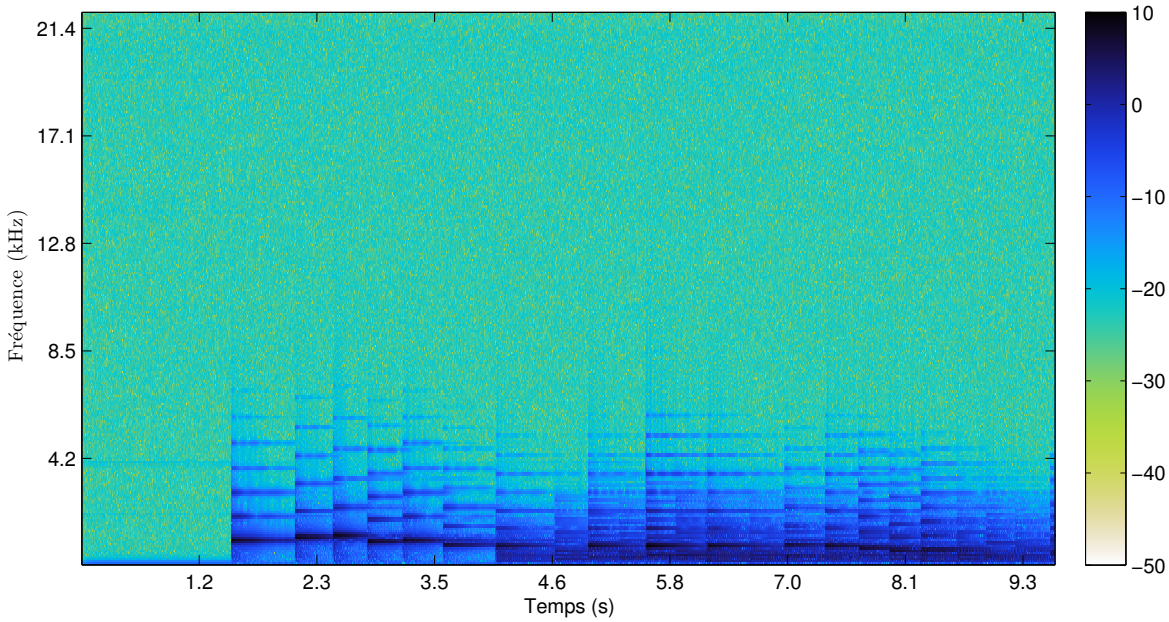
où $\hat{\mathbb{E}}[.]$ représente une espérance empirique calculée par exemple par moyennage des valeurs des points temps-fréquence voisins. Dans ce cas, le voisinage du point temps-fréquence (f, n) peut être spécifié au moyen d'une fenêtre bi-dimensionnelle $\mathbf{W} = \{w_{f'n'}\}$ et la covariance au point temps-fréquence (f, n) est calculée selon

$$\hat{\mathbf{R}}_{\mathbf{x}_{fn}} = \sum_{f',n'} w^2_{f'n'}\mathbf{x}_{f'n'}\mathbf{x}^H_{f'n'} \tag{A.4}$$

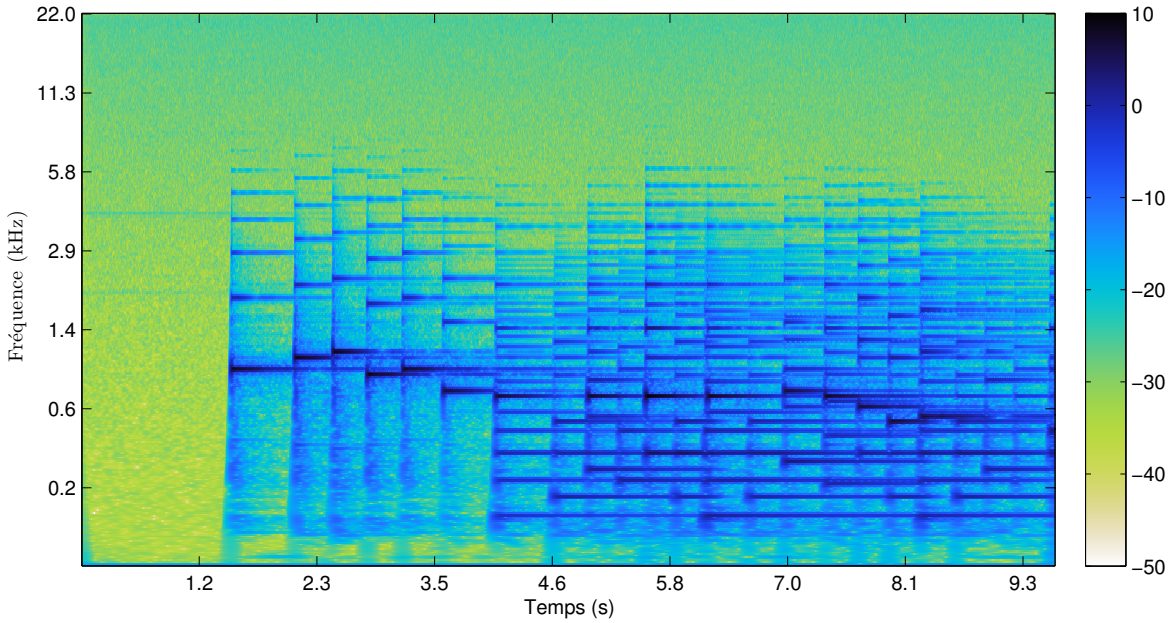
La transformée QERB (pour *Quadratic Equivalent Rectangular Bandwidth*) [Duong et al., 2010] que nous avons privilégiée dans notre étude fait partie de cette famille de représentations. La covariance est obtenue en intégrant chaque signal $\mathbf{x}^f(t)$ en sortie du banc de filtres ERB à l'aide de fenêtres sinusoïdales $\mathbf{w}$ de longueur L. La covariance au point temps-fréquence (f, n) est donc donnée par

$$\hat{\mathbf{R}}^{\text{ERB}}_{\mathbf{x}_{fn}} = w^2(l) \sum_{l=1}^{L} \mathbf{x}^f(l + (n-1)h)\, \mathbf{x}^{fH}(l + (n-1)h). \tag{A.5}$$

Dans nos expériences, nous avons choisi empiriquement $F = 350$ bandes de fréquence pour la représentation QERB. La figure A.1 présente le spectrogramme de log-puissance obtenu par TFCT classique (soit $\log(\mathbf{x}_{fn}\mathbf{x}^H_{fn})$) et le spectrogramme de log-puissance QERB (soit $\log \hat{\mathbf{R}}^{\text{ERB}}_{\mathbf{x}_{fn}}$), pour un extrait monophonique de fréquence d'échantillonnage $f_s = 44.1$ kHz composé de quelques notes de piano, tout deux calculés pour une fenêtre d'analyse sinusoïdale de $L = 512$ échantillons et pour un recouvrement de 50%. Notons bien que le signal considéré étant monocanal, le spectrogramme de log-puissance QERB est équivalent au spectrogramme de log-puissance obtenu par transformée linéaire avec échelle ERB. Ces figures illustrent donc le gain de résolution dans les basses fréquences obtenus grâce à l'échelle ERB.



(a) Spectrogramme de log-puissance (TFCT)



(b) Spectrogramme de log-puissance QERB

FIGURE A.1 – Spectrogramme de log-puissance en échelle linéaire (TFCT) et spectrogramme de log-puissance QERB.

Annexe B

Détails sur l'initialisation des modèles EF pour le rehaussement de la parole

Nous détaillons ici la méthode d'initialisation de la partie filtre du modèle excitation-filtre pour l'apprentissage d'un modèle de voix sur le corpus *CHiME*. Pour rappel, dans ce cas, la variance de la source de parole est exprimée par

$$v_{1,fn} = \left(\sum_{k=1}^{K_1^{\text{ex}}} w_{1,fk}^{\text{ex}} h_{1,kn}^{\text{ex}} \right) \left(\sum_{k=1}^{K_1^{\text{ft}}} \sum_{k'=1}^{K_1'^{\text{ft}}} b_{1,fk'}^{\text{ft}} u_{1,k'k}^{\text{ft}} h_{1,kn}^{\text{ft}} \right). \tag{B.1}$$

Comme pour le modèle NMF, l'observation considérée ici est la concaténation $x(t)$ des 500 extraits de voix seule de l'ensemble d'apprentissage, pour un locuteur donné. Le calcul de l'initialisation des matrices de la partie filtre suit les étapes suivantes :

1. **Calcul des MFCCs :** Pour un locuteur donné, on calcule C coefficients MFCC pour chaque trame n de l'observation ainsi que la transformée temps-fréquence choisie de l'observation. Nous calculons alors le spectrogramme de puissance de l'observation $|\mathbf{X}|^{.2} = \{|x_{fn}|^{.2}\}$ de taille $F \times N$ et formons la matrice des coefficients MFCC notée $\mathbf{M}_{\text{MFCC}}$ de taille $C \times N$ en concaténant les vecteurs de coefficients MFCC de chaque trame.
2. **Inversion des MFCCs :** En inversant les MFCC de chacune des trames, on obtient un spectrogramme de puissance $|\mathbf{X}|_{(\text{ft})}^{.2}$ qui est découplé de l'excitation. On peut également en déduire le spectrogramme de puissance de l'excitation $|\mathbf{X}|_{(\text{ex})}^{.2} = |\mathbf{X}|^{.2} ./ |\mathbf{X}|_{(\text{ft})}^{.2}$, en divisant terme-à-terme le spectrogramme de puissance de l'observation par le spectrogramme de puissance de la partie filtre.
3. **Clustering des MFCCs :** Les MFCCs sont ensuite groupés en $K_1'^{\text{ft}}$ *clusters* par quantification vectorielle. Chaque trame n est donc associée à un cluster. On procède alors séparément pour chacun des $K_1'^{\text{ft}}$ clusters.
4. **Calcul des initialisations de la partie filtre :** Pour chaque cluster k, le spectrogramme $|\mathbf{X}|_{(\text{ft}),k}^{.2}$ de la partie filtre ne contenant que les N_k trames associées à ce cluster est décomposé par NMF comme le produit $\mathbf{B}_1^{\text{ft}} \mathbf{U}_{1,k}^{\text{ft}} \mathbf{H}_{1,k}^{\text{ft}}$ où $\mathbf{B}_1^{\text{ft}}$ est fixé, de taille $F \times K_1'^{\text{ft}}$ et composé de $K_1'^{\text{ft}} = 230$ spectres à bande étroite, $\mathbf{U}_{1,k}^{\text{ft}}$ est de taille $K_1^{\text{ft}} \times 1$ et $\mathbf{H}_{1,k}^{\text{ft}}$ est de taille $1 \times N_k$.

Finalement, la matrice complète $\mathbf{U}_1^{\text{ft}}$ est initialisée comme la concaténation des K_1^{ft} vecteurs $\mathbf{U}_{1,k}^{\text{ft}}$ ainsi appris. La matrice $\mathbf{H}_1^{\text{ft}}$ est elle réinitialisée avec des 1 et l'apprentissage du modèle EF complet de la voix du locuteur considéré est alors effectué comme expliqué dans la partie 3.3.4.

Annexe C

Descriptif du corpus *ccMixter*

Nous présentons ci-après un tableau décrivant succinctement les 49 titres qui composent le corpus *ccMixter*. Pour chacun, nous précisons le nom de l'artiste, le nom du titre, le nombre d'extraits de 20 à 30 secondes (colonne **Nb**) ainsi qu'un rapide descriptif de la source de voix et de la source d'accompagnement musical. Nous rappelons que tous les morceaux sont stéréophoniques et échantillonés à 44.1 kHz. La première colonne indique de plus à quel groupe appartient chaque titre, sachant que l'index de groupe attribué correspond au groupe dans lequel l'exemple considéré fait partie de l'ensemble de test. Ce tableau nous permet notamment de remarquer que, même si le même artiste peut être crédité sur plusieurs titres dans différents groupes, cela n'implique pas forcément que les voix ou accompagnements sont similaires. Ils peuvent même être très variés.

Groupe	Artiste	Titre	Nb	Descriptifs de la voix	Descriptif de l'accompagnement
1	Javolenus	You don't get in touch anymore	8	Voix d'homme, peu de réverbération	Guitare folk
1	Kamihamiha	No peace for the middle east	6	Voix d'homme, plutôt parlée	Percussions électroniques, basse électronique saturée, guitares saturées
1	Mr Yesterday	King Richard's blues	5	Voix d'homme, peu de réverbération	Guitare, basse, batterie, orgue jazz
1	Stellarartwars	A forest	6	Voix d'homme, peu de réverbération	Guitare électrique, guitare folk, basse, batterie, synthétiseur
1	Stellarartwars	Finally found	7	Voix d'homme rapée, beaucoup de réverbération	Synthétiseurs, beaucoup d'effet de réverbération
1	Stellarartwars	Square bidness	8	Voix d'homme parlée, parfois doublée	Synthétiseur, basse, batterie
1	Tmray	Forget it	3	Voix d'homme, saturée, avec réverbération	Guitares électriques, percussions
1	Tmray	Roulette	8	Voix d'homme, parfois saturée, parfois doublée	Guitare électrique saturée, guitare électrique claire, basse, batterie
1	VJ Memes	What child is this	6	Voix d'homme, peu de réverbération	Piano, guitare folk, orgue jazz
1	Von Korf	Marktversagen	7	Voix d'homme, peu de réverbération	Guitare électrique saturée, basse, batterie
2	Alex Beroza	To be sensitive	7	Voix de femme avec effet de réverbération	Musique électronique : claviers, percussions
2	Doxent	Alice in the city	7	Voix de femme, peu de réverbération	Piano, ensemble de cordes synthétique
2	Stellarartwars	Amy Winehouse Blues	8	Voix d'homme, légèrement saturée	Piano électronique, batterie, guitare électrique, saxophone synthétique
2	Stellarartwars	Black swan	6	Voix de femme, peu de réverbération	Synthétiseurs, batterie, basse
2	Stellarartwars	Dollar	6	Voix d'homme, peu de réverbération	Synthétiseurs, basse électrique, guitare électrique claire

Groupe	Artiste	Titre	Nb	Descriptifs de la voix	Descriptif de l'accompagnement
2	Stellarartwars	I will follow	7	Voix de femme, peu de réverbération	Piano électronique, basse électronique, guitare électrique, batterie
2	Stellarartwars	Lucky boy	6	Voix d'homme, peu de réverbération	Batterie, synthétiseurs, basse
2	Stellarartwars	To be sensitive	6	Voix de femme, peu de réverbération	Piano, basse,
2	Unreal dm	Californian winter	7	Voix de femme, légèrement saturée	Guitares électriques, synthétiseur, basse, batterie
2	Unreal dm	Just beginning	7	Voix de femme, peu de réverbération	Piano électronique, basse, batterie
3	Casimps1	Bingo	3	Chœur d'enfants	Piano et batterie
3	Casimps1	Itsy Bitsy Spider	2	Chœur d'enfants	Piano, guitare folk, batterie
3	Copperhead	Need you baby	8	Voix d'homme, peu de réverbération	Reggae : clavier, guitare électrique, basse, batterie
3	Geertveneklaas	Blue boy	7	Voix d'homme, peu de réverbération	Orchestre : vents, cordes, pas de percussions
3	Nikmusik	Little white lies	6	Voix d'homme, parfois doublée, avec réverbération	Guitare, basse, batterie, synthétiseur
3	Stellarartwars	Breakdown boy	7	Voix de femme, peu de réverbération	Synthétiseurs, batterie électronique, basse
3	Stellarartwars	Elements	8	Voix d'homme, peu de réverbération, parlée	Synthétiseurs, batterie
3	Stellarartwars	Human race	5	Voix de femme, parfois beaucoup de réverbération	Synthétiseurs, orgue électronique, batterie
3	Stellarartwars	Occupy walk	6	Voix d'homme, peu de réverbération	Batterie, orgues électroniques
3	Stellarartwars	Pheromones	6	Voix de femme, peu de réverbération	Synthétiseurs, percussions
4	Casimps1	Twinkle twinkle little star	2	Chœur d'enfants	Basse, synthétiseur, batterie

Groupe	Artiste	Titre	Nb	Descriptifs de la voix	Descriptif de l'accompagnement
4	Per	Orange	7	Voix d'homme doublée, beaucoup de réverbération	Musique électronique : percussions et sons synthétiques
4	Stellarartwars	Love liberation	6	Voix d'homme, peu de réverbération	Batterie, synthétiseurs, piano
4	Stellarartwars	Music	5	Voix de femme, beaucoup d'effet	Basse, percussions, synthétiseurs
4	Stellarartwars	On a silent night	8	Voix d'homme, peu de réverbération	Basse, batterie, guitare électrique
4	Stellarartwars	Orange	7	Voix d'homme, peu de réverbération	Basse, synthétiseurs, batterie électronique, beaucoup d'effets
4	Stellarartwars	Perfectly 4 u	3	Voix d'homme, peu de réverbération	Basse, synthétiseurs, batterie électronique
4	Stellarartwars	Sweet seduction	8	Voix d'homme	Basse, batterie, percussion, synthétiseur
4	Stellarartwars	Winter glow	8	Voix de femme, peu de réverbération	Basse électronique, piano électronique, synthétiseur, batterie
4	Unreal dm	Big john	4	Voix de femme, peu de réverbération, quelques chœurs	Piano, guitare électrique claire, batterie, basse
5	Casimps1	Could be	3	Voix de femme sans effets, et quelques chœurs	Basse, guitare folk, batterie
5	Javolenus	King Henry	8	Voix d'homme, peu de réverbération, parfois doublée	Guitare folk
5	Kamihamiha	A spatial voyage from A to B	6	Voix d'homme, parfois saturée, plutôt parlée	Guitares saturées, batterie, synthétiseurs
5	Kamihamiha	Trash the disco	6	Voix d'homme, souvent doublée, plutôt parlée	Basse électronique, percussion électronique, synthétiseurs
5	Mindmapthat	Not so happy holidays	7	Voix de femme, peu de réverbération	Piano, basse, batterie
5	Stellarartwars	Bird	7	Voix de femme, peu de réverbération, beaucoup d'effet	Musique électronique : batterie, basse, synthétiseur

Groupe	Artiste	Titre	Nb	Descriptifs de la voix	Descriptif de l'accompagnement
5	Stellarartwars	Emma	6	Voix d'homme, peu de réverbération	Synthétiseurs, batterie, basse électronique
5	Stellarartwars	Go time	8	Voix d'homme parlée, parfois doublée, peu de réverbération	Batterie électronique, synthétiseur, piano, basse électronique
5	Unreal dm	Jerusalem	6	Voix d'homme, peu saturée	Synthétiseur, guitare électrique, guitare folk, batterie, basse

TABLEAU C.1 – Descriptif succinct du corpus *ccMixter*.

Annexe D

Calcul des gradients pour l'optimisation

Nous précisions dans cette partie les différents calculs de gradients des fonctions de coût dont nous avons proposé l'optimisation grâce à l'outil d'optimisation de *MATLAB*.

D.1 Fusion statique invariante par maximisation du SDR

Dans le cas de la fusion statique invariante, la fonction de coût à minimiser par rapport au vecteur de coefficients de fusion $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_m, ..., \alpha_M)^\mathsf{T}$ s'écrit

$$\varphi(\boldsymbol{\alpha}) = \sum_l \log_{10}\left(c^{(l)} + \boldsymbol{\alpha}^\mathsf{T}\,\widetilde{\mathbf{G}}^{(l)}\,\boldsymbol{\alpha} - 2\,\tilde{\mathbf{d}}^{(l)\mathsf{T}}\,\boldsymbol{\alpha}\right). \tag{D.1}$$

où les matrices de Gram $\widetilde{\mathbf{G}}^{(l)}$, les vecteurs $\tilde{\mathbf{d}}^{(l)}$ et les scalaires $c^{(l)}$ sont définis comme aux équations (4.4), (4.5) et (4.6). Nous pouvons constater que cette fonction peut être réécrite comme la somme de termes relatifs à chaque exemple l, selon

$$\varphi(\boldsymbol{\alpha}) = \sum_l \log_{10}\varphi^{(l)}(\boldsymbol{\alpha}) \tag{D.2}$$

avec

$$\varphi^{(l)}(\boldsymbol{\alpha}) = c^{(l)} + \boldsymbol{\alpha}^\mathsf{T}\,\widetilde{\mathbf{G}}^{(l)}\,\boldsymbol{\alpha} - 2\,\tilde{\mathbf{d}}^{(l)\mathsf{T}}\,\boldsymbol{\alpha}. \tag{D.3}$$

Le gradient de la fonction de coût s'écrit

$$\mathbf{grad}(\varphi) = \left(\frac{\partial\varphi(\boldsymbol{\alpha})}{\partial\alpha_1}, ..., \frac{\partial\varphi(\boldsymbol{\alpha})}{\partial\alpha_m}, ..., \frac{\partial\varphi(\boldsymbol{\alpha})}{\partial\alpha_M}\right)^\mathsf{T}. \tag{D.4}$$

Quel que soit m, le terme $\frac{\partial\varphi(\boldsymbol{\alpha})}{\partial\alpha_m}$ peut s'écrire

$$\frac{\partial\varphi(\boldsymbol{\alpha})}{\partial\alpha_m} = \sum_l \frac{\partial\log_{10}\varphi^{(l)}(\boldsymbol{\alpha})}{\partial\alpha_m} = \sum_l \frac{1}{\varphi^{(l)}(\boldsymbol{\alpha})}\frac{\partial\varphi^{(l)}(\boldsymbol{\alpha})}{\partial\alpha_m}. \tag{D.5}$$

Il ne reste donc plus qu'à calculer la dérivée partielle de $\varphi^{(l)}(\boldsymbol{\alpha})$ par rapport à α_m. Rappelant que $\widetilde{\mathbf{G}}^{(l)} = \left\{\tilde{g}^{(l)}_{m_1 m_2}\right\}$ et que $\tilde{\mathbf{d}}^{(l)} = \left\{\tilde{d}^{(l)}_m\right\}$, cette dérivée s'écrit

$$\frac{\partial\varphi^{(l)}(\boldsymbol{\alpha})}{\partial\alpha_m} = \sum_{m_1}^{M}(\tilde{g}^{(l)}_{m_1 m} + \tilde{g}^{(l)}_{m m_1})\alpha_{m_1} - 2d_m. \tag{D.6}$$

Finalement, en notant que la matrice de Gram $\widetilde{\mathbf{G}}^{(l)}$ est symétrique, le gradient total $\mathbf{grad}(\varphi)$ peut s'écrire sous forme matricielle

$$\mathbf{grad}(\varphi) = \sum_l \frac{1}{\varphi^{(l)}(\boldsymbol{\alpha})} \mathbf{grad}(\varphi^{(l)}) \tag{D.7}$$

avec

$$\mathbf{grad}(\varphi^{(l)}) = 2\left(\widetilde{\mathbf{G}}^{(l)}\boldsymbol{\alpha} - \tilde{\mathbf{d}}^{(l)}\right). \tag{D.8}$$

D.2 Fusion statique variant en fréquence

Pour le cas de la fusion statique variant en fréquence, la maximisation du SDR est menée indépendamment sur chaque bande de fréquence f. Il s'agit alors de minimiser la fonction de coût

$$\varphi_f(\boldsymbol{\alpha}_f) = \sum_l \log_{10}\left(c_f^{(l)} + \boldsymbol{\alpha}_f^{\mathsf{T}}\,\widetilde{\mathbf{G}}_f^{(l)}\,\boldsymbol{\alpha}_f - 2\,\tilde{\mathbf{d}}_f^{(l)\mathsf{T}}\,\boldsymbol{\alpha}_f\right) \tag{D.9}$$

par rapport au vecteur de coefficients $\boldsymbol{\alpha}_f$ relatif à la bande de fréquence f considérée. Nous remarquons donc que la seule différence avec la fonction de coût dans le cas invariant (D.1) réside dans les termes $c_f^{(l)}$, $\tilde{\mathbf{d}}_f^{(l)}$ et $\widetilde{\mathbf{G}}_f^{(l)}$ qui sont cette fois calculés sur une bande de fréquence f donnée.

Le calcul du gradient de la fonction de coût est donc inchangé par rapport au cas invariant et s'écrit donc

$$\mathbf{grad}(\varphi_f) = \sum_l \frac{1}{\varphi_f^{(l)}(\boldsymbol{\alpha})} \mathbf{grad}(\varphi_f^{(l)}) \tag{D.10}$$

avec

$$\varphi_f(\boldsymbol{\alpha}) = \sum_l \log_{10} \varphi_f^{(l)}(\boldsymbol{\alpha}_f) \tag{D.11}$$

et

$$\mathbf{grad}(\varphi_f^{(l)}) = 2\left(\widetilde{\mathbf{G}}_f^{(l)}\boldsymbol{\alpha}_f - \tilde{\mathbf{d}}_f^{(l)}\right). \tag{D.12}$$

D.3 Fusion adaptative VB invariante

Dans le cas de la fusion adaptative VB invariante (avec paramètre β contrôlant l'entropie), le cas de la minimisation de l'EQM n'étant plus formulé comme un programme quadratique, il convient de calculer le gradient de la fonction de coût, au même titre que pour le cas de la maximisation du SDR.

D.3.1 Minimisation de l'EQM

Afin de minimiser l'EQM sur l'ensemble d'apprentissage, il convient de minimiser la fonction de coût suivante, par rapport aux probabilités *a priori* $\boldsymbol{\pi} = (\pi_1, ..., \pi_m, ..., \pi_M)^{\mathsf{T}}$ et au paramètre de contrôle de l'entropie β :

$$\varphi_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta) = \sum_l \varphi_{\text{VB-EQM}}^{(l)}(\boldsymbol{\pi}, \beta) \tag{D.13}$$

avec

$$\varphi_{\text{VB-EQM}}^{(l)}(\boldsymbol{\pi}, \beta) = \left[c^{(l)} + \frac{1}{\delta^{(l)\,2}}\left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right)^{\mathsf{T}} \widetilde{\mathbf{G}}^{(l)} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right) - \frac{2}{\delta^{(l)}}\,\tilde{\mathbf{d}}^{(l)\mathsf{T}} \left(\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta}\right)\right].$$

Nous noterons que, contrairement à la fusion statique invariante, nous avons ici explicitement formulé la contrainte de normalisation des coefficients par le biais du terme

$$\delta^{(l)} = \sum_{m=1}^{M} \pi_m \, e^{\mathcal{L}_m^{(l)}/\beta} \tag{D.14}$$

qui dépend donc des paramètres $\boldsymbol{\pi}$ et β à optimiser. Toutefois, en posant $\boldsymbol{\alpha} = (\boldsymbol{\pi} \circ \mathrm{e}^{\boldsymbol{\mathcal{L}}^{(l)}/\beta})/\delta^{(l)}$, nous pouvons remarquer que la fonction $\varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)$ est équivalente à la fonction $\varphi^{(l)}(\boldsymbol{\alpha})$ définie à l'équation (D.3).

Ainsi, le gradient de la fonction de coût $\varphi_{\text{VB-EQM}}$ peut être décomposé comme suit :

$$\mathbf{grad}(\varphi_{\text{VB-EQM}}) = \sum_{l} \mathbf{grad}(\varphi^{(l)}_{\text{VB-EQM}}) \tag{D.15}$$

avec

$$\mathbf{grad}(\varphi^{(l)}_{\text{VB-EQM}}) = \left(\frac{\partial \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)}{\partial \pi_1}, ..., \frac{\partial \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)}{\partial \pi_m}, \right.$$
$$\left. ..., \frac{\partial \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)}{\partial \pi_M}, \frac{\partial \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)}{\partial \beta} \right)^{\mathsf{T}}.$$

Les termes $\frac{\partial \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)}{\partial \pi_m}$ peuvent donc être calculés à partir des calculs déjà menés dans le cas statique tel que

$$\frac{\partial \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)}{\partial \pi_m} = \frac{\partial \varphi^{(l)}(\boldsymbol{\alpha})}{\partial \alpha_m} \frac{\partial \alpha_m}{\partial \pi_m}. \tag{D.16}$$

Le premier terme ayant déjà été calculé à l'équation (D.6) pour le cas statique, il ne reste plus qu'à calculer

$$\frac{\partial \alpha_m}{\partial \pi_m} = \frac{\partial}{\partial \pi_m} \left(\frac{\pi_m \, \mathrm{e}^{\mathcal{L}_m^{(l)}/\beta}}{\delta^{(l)}} \right) = \frac{\mathrm{e}^{\mathcal{L}_m^{(l)}/\beta}}{\delta^{(l)}} \left(1 - \pi_m \, \mathrm{e}^{\mathcal{L}_m^{(l)}/\beta} \right) = \frac{\mathrm{e}^{\mathcal{L}_m^{(l)}/\beta}}{\delta^{(l)}} \left(1 - \alpha_m \right). \tag{D.17}$$

Enfin, de la même manière, le terme $\frac{\partial \varphi^{(l)}(\boldsymbol{\pi}, \beta)}{\partial \beta}$ peut être décomposé en deux termes tel que

$$\frac{\partial \varphi^{(l)}(\boldsymbol{\pi}, \beta)}{\partial \beta} = \sum_{m=1}^{M} \frac{\partial \varphi^{(l)}(\boldsymbol{\alpha})}{\partial \alpha_m} \frac{\partial \alpha_m}{\partial \beta}. \tag{D.18}$$

dont seul $\frac{\partial \alpha_m}{\partial \beta}$ reste à calculer. Ce dernier terme vaut

$$\frac{\partial \alpha_m}{\partial \beta} = \frac{\alpha_m}{\beta^2} \left(\sum_{m_1} \mathcal{L}_{m_1} \alpha_{m_1} - \mathcal{L}_m \right). \tag{D.19}$$

D.3.2 Maximisation du SDR

Afin de maximiser le SDR, il convient cette fois de minimiser la fonction de coût

$$\varphi_{\text{VB-SDR}}(\boldsymbol{\pi}, \beta) = \sum_{l} \log_{10} \varphi^{(l)}_{\text{VB-SDR}}(\boldsymbol{\pi}, \beta). \tag{D.20}$$

Son gradient peut s'écrire sous la forme

$$\mathbf{grad}(\varphi_{\text{VB-SDR}}) = \sum_l \frac{1}{\varphi^{(l)}_{\text{VB-SDR}}(\boldsymbol{\pi}, \beta)} \mathbf{grad}(\varphi^{(l)}_{\text{VB-SDR}}). \tag{D.21}$$

En remarquant alors que $\varphi^{(l)}_{\text{VB-SDR}}(\boldsymbol{\pi}, \beta) = \varphi^{(l)}_{\text{VB-EQM}}(\boldsymbol{\pi}, \beta)$, l'expression du gradient peut être simplement déduite des calculs menés pour le cas de la minimisation de l'EQM.

D.4 Fusion adaptative VB variant en fréquence

Enfin, pour la fusion adaptative VB variant en fréquence, il suffit de remarquer que les fonctions de coût $\varphi^{(l)}_{\text{VB-EQM},f}$ et $\varphi^{(l)}_{\text{VB-SDR},f}$ sont les mêmes que dans le cas invariant, à ceci prêt que les termes $c^{(l)}$, $\tilde{\mathbf{d}}^{(l)}$, $\widetilde{\mathbf{G}}^{(l)}$ et $\mathcal{L}^{(l)}$ sont à remplacer par leurs équivalents $c^{(l)}_f$, $\tilde{\mathbf{d}}^{(l)}_f$, $\widetilde{\mathbf{G}}^{(l)}_f$ et $\mathcal{L}^{(l)}_f$ calculés sur la bande de fréquence f considérée.

Annexe E

Inférence variationnelle Bayésienne pour la NMF

Nous proposons ici d'introduire quelques détails de calcul qui nous ont permis de dériver les règles de mises à jour pour l'inférence variationnelle bayésienne de la NMF. Nous considérons ici le cas de la NMF à ordre multiple, introduite dans le chapitre 5.3, étant donné que la NMF à ordre unique peut être vue comme un cas particulier avec $m = 1$.

Pour rappel, dans ce cas, l'ensemble des paramètres s'écrit

$$\mathbf{Z} = \{\mathbf{S}, \mathbf{W}, \mathbf{H}, \mathbf{K}\} \tag{E.1}$$

avec $\mathbf{K} = \{K_j\}$, $\mathbf{W} = \left\{\mathbf{W}_{j1}, ..., \mathbf{W}_{jm}, ..., \mathbf{W}_{jM_j}\right\}_{j=1..J}$ et $\mathbf{H} = \left\{\mathbf{H}_{j1}, ..., \mathbf{H}_{jm}, ..., \mathbf{H}_{jM_j}\right\}_{j=1..J}$. Leur probabilité conjointe est donnée par :

$$p(\mathbf{X}, \mathbf{Z}) \;=\; p(\mathbf{X}|\mathbf{S})\, p(\mathbf{S}|\mathbf{W}, \mathbf{H})\, p(\mathbf{W}|\mathbf{K})\, p(\mathbf{H}|\mathbf{K})\, p(\mathbf{K}). \tag{E.2}$$

E.1 Maximisation de la borne inférieure par rapport aux variables auxiliaires

Nous détaillons ici le calcul des paramètres $\omega^*_{jm,fn}$ et $\phi^*_{jm,fn,k}$ maximisant la borne inférieure paramétrique de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega})$ déjà définie à l'équation (5.17) et qui peut être réécrite

$$\mathcal{B}[q](\mathbf{\Omega}) = \mathbb{E}_{\mathbf{Z}}[\log f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega})] - \mathbb{E}_{\mathbf{Z}}[q(\mathbf{Z})]. \tag{E.3}$$

Pour rappel, le terme $\log f(\mathbf{X}, \mathbf{Z}, \mathbf{\Omega})$ vise à approximer la log-probabilité conjointe $\log p(\mathbf{X}, \mathbf{Z})$. En particulier, le terme $\mathbb{E}_{\mathbf{Z}}[\log p(\mathbf{S}|\mathbf{W}, \mathbf{H})]$ pose problème puisque dans son expression

$$\begin{aligned} \mathbb{E}_{\mathbf{Z}}\left[\log p(\mathbf{S}|\mathbf{W}, \mathbf{H})\right] = \quad & -JFN \log \pi - \textstyle\sum_{j,fn} \sum_{m=1}^{M_j} q(K_{jm}) \mathbb{E}_{\mathbf{Z}}\left[\log v_{jm,fn}\right] \\ & - \textstyle\sum_{j,fn} \sum_{m=1}^{M_j} q(K_{jm}) \mathbb{E}_{\mathbf{Z}}\left[|s_{j,fn}|^2\right] \mathbb{E}_{\mathbf{Z}}[v^{-1}_{jm,fn}] \end{aligned} \tag{E.4}$$

où $v_{jm,fn} = \sum_{k=1}^{K_{jm}} w_{jm,fk} h_{jm,kn}$, les termes $\mathbb{E}_{\mathbf{Z}}\left[\log v_{jm,fn}\right]$ et $\mathbb{E}_{\mathbf{Z}}[v^{-1}_{jm,fn}]$ n'ont pas de solution analytique.

Comme cela a été montré dans la partie 5.1.2 pour le cas de la NMF à ordre unique, et d'après [Adiloğlu et Vincent, 2012; Hoffman et al., 2010], la fonction $v_{jm,fn} \to -\log(v_{jm,fn})$ étant convexe, elle peut être approchée par son développement en série de Taylor au premier ordre autour d'un point positif $\omega_{jm,fn}$ quelconque de sorte que

$$-\log v_{jm,fn} \geq -\log \omega_{jm,fn} + 1 - \frac{1}{\omega_{jm,fn}} v_{jm,fn}. \tag{E.5}$$

Pour le second terme, la fonction $v_{jm,fn} \rightarrow -v_{jm,fn}^{-1}$ étant concave, alors $\forall \phi_{jm,fn,k} \geq 0$ tel que $\sum_{k=1}^{K_{jm}} \phi_{jm,fn,k} = 1$, nous avons :

$$-\left(\sum_{k=1}^{K_{jm}} w_{jm,fk} h_{jm,kn}\right)^{-1} \geq -\sum_{k=1}^{K_{jm}} \phi_{jm,fn,k}^2 w_{jm,fk}^{-1} h_{jm,kn}^{-1}. \tag{E.6}$$

En utilisant les inégalités (E.5) et (E.6), nous pouvons alors formuler une borne inférieure de $\log p(\mathbf{S}|\mathbf{W},\mathbf{H})$ telle que :

$$\log p(\mathbf{S}|\mathbf{W},\mathbf{H}) \geq \log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega},\boldsymbol{\phi}) \tag{E.7}$$

avec

$$\begin{aligned} \log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega},\boldsymbol{\phi}) = \quad & -JFN\log\pi \\ & -\textstyle\sum_{j,fn}\sum_{m=1}^{M_j} q(K_{jm})\left(-\log\omega_{jm,fn} + 1 - \frac{1}{\omega_{jm,fn}} v_{jm,fn}\right) \\ & -\textstyle\sum_{j,fn}|s_{j,fn}|^2 \sum_{m=1}^{M_j} q(K_{jm}) \sum_{k=1}^{K_{jm}} \phi_{jm,fn,k}^2 w_{jm,fk}^{-1} h_{jm,kn}^{-1}. \end{aligned} \tag{E.8}$$

où $\boldsymbol{\omega} = \{\omega_{jm,fn}\}$ et $\boldsymbol{\phi} = \{\phi_{jm,fn,k}\}$ désignent l'ensemble des paramètres auxiliaires définissant la borne inférieure.

Afin de maximiser la borne inférieure $\mathcal{B}[q](\boldsymbol{\Omega})$, il suffit donc de trouver les paramètres $\omega_{jm,fn}^*$ et $\phi_{jm,fn,k}^*$ pour lesquels la dérivée du terme problématique $\mathbb{E}_{\mathbf{Z}}[\log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega},\boldsymbol{\phi})]$ s'annule.

La dérivée par rapport à $\omega_{jm,fn}$ s'écrivant

$$\begin{aligned} \frac{\partial \mathcal{B}[q](\boldsymbol{\Omega})}{\partial \omega_{jm,fn}} &= \frac{\partial \mathbb{E}_{\mathbf{Z}}[\log g(\mathbf{S}|\mathbf{W},\mathbf{H},\boldsymbol{\omega},\boldsymbol{\phi})]}{\partial \omega_{jm,fn}} \\ &= -\frac{1}{\omega_{jm,fn}} + \frac{1}{\omega_{jm,fn}^2}\mathbb{E}_{\mathbf{Z}}[v_{jm,fn}], \end{aligned}$$

celle-ci s'annule pour

$$\omega_{jm,fn}^* = \mathbb{E}_{\mathbf{Z}}[v_{jm,fn}]. \tag{E.9}$$

Pour le calcul de $\phi_{jm,fn,k}^*$, nous devons tenir compte de la contrainte $\sum_{k=1}^{K_{jm}} \phi_{jm,fn,k} = 1$. Pour cela, nous introduisons le multiplicateur de Lagrange $\lambda_{jm,fn}$ et définissons le lagrangien de la borne inférieure par

$$\mathcal{B}'[q](\boldsymbol{\Omega}) = \mathcal{B}[q](\boldsymbol{\Omega}) + \sum_{j,fn}\sum_{m=1}^{M_j} \lambda_{jm,fn}\left(\sum_{k=1}^{K_{jm}} \phi_{jm,fn,k} - 1\right). \tag{E.10}$$

En dérivant par rapport aux paramètres $\phi_{jm,fn,k}$ et aux multiplicateurs $\lambda_{jm,fn}$, nous obtenons le système d'équations suivant :

$$\frac{\partial \mathcal{B}'[q](\boldsymbol{\Omega})}{\partial \phi_{jm,fn,k}} = -\,\mathbb{E}_{\mathbf{Z}}\left[|s_{j,fn}|^2\right]\, q(K_{jm}) \sum_{k=1}^{K_{jm}} \phi_{jm,fn,k}^2\, \mathbb{E}_{\mathbf{Z}}[v_{jm,fn}^{-1}] \tag{E.11}$$

$$\frac{\partial \mathcal{B}'[q](\boldsymbol{\Omega})}{\partial \lambda_{jm,fn}} = \left(\sum_{k=1}^{K_{jm}} \phi_{jm,fn,k}\right) - 1. \tag{E.12}$$

Ces dérivées s'annulent alors pour

$$\phi^*_{jm,fn,k} = \frac{1}{C_{jm,fn}} \mathbb{E}_{\mathbf{Z}}\left[w^{-1}_{jm,fk}\right]^{-1} \mathbb{E}_{\mathbf{Z}}\left[h^{-1}_{jm,kn}\right]^{-1} \tag{E.13}$$

où $C_{jm,fn}$ désigne la constante de normalisation

$$C_{jm,fn} = \sum_{k=1}^{K_{jm}} \mathbb{E}_{\mathbf{Z}}\left[w^{-1}_{jm,fk}\right]^{-1} \mathbb{E}_{\mathbf{Z}}\left[h^{-1}_{jm,kn}\right]^{-1}. \tag{E.14}$$

E.2 Calcul des distributions variationnelles

Nous souhaitons ci-après justifier le calcul des distributions variationnelles maximisant l'énergie libre. Ces distributions sont obtenues en dérivant l'énergie libre $\mathcal{L}[q]$ (ou en pratique dans notre cas, l'approximation de l'énergie libre $\mathcal{B}[q](\mathbf{\Omega}^*)$) par rapport à chaque distribution variationnelle. Pour simplifier notre démonstration, nous omettrons ici les indices de source j, de trame n et de bande de fréquence f de sorte que l'ensemble des paramètres $\mathbf{Z}$ n'est composé que d'une variable de sources $\mathbf{S}$, de deux variables de NMF $\mathbf{W}$ et $\mathbf{H}$ ainsi que d'une variable $\mathbf{K}$ relative au nombre de composantes. Nous supposerons donc que la probabilité jointe s'écrit simplement

$$p(\mathbf{X}, \mathbf{Z}) = p(\mathbf{X}|\mathbf{S})p(\mathbf{S}|\mathbf{W}, \mathbf{H})p(\mathbf{W}|\mathbf{K})p(\mathbf{H}|\mathbf{K}) \tag{E.15}$$

et la distribution variationnelle se factorise en

$$q(\mathbf{Z}) = q_{\mathbf{S}}(\mathbf{S})q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})q_{\mathbf{K}}(\mathbf{K}). \tag{E.16}$$

Contrairement au cas étudié dans [Bishop, 2006], nous conservons ici certains conditionnements dans la définition de la distribution variationnelle. De ce fait, les solutions classiques telles que définies à l'équation 5.15 ne sont pas directement applicables et il nous faut dériver pour chaque cas l'expression de la distribution variationnelle maximisant l'énergie libre

$$\mathcal{L}[q] = \int q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} d\mathbf{Z}. \tag{E.17}$$

E.2.1 Distribution variationnelle des sources

Le calcul de la distribution variationnelle des sources $q_{\mathbf{S}}(\mathbf{S})$ se trouve être équivalent au cas traditionnel sans conditionnement. Nous détaillons quand même son calcul, afin d'illustrer les différences avec le calcul des autres distributions variationnelles.

La distribution variationnelle optimale des sources est obtenue en dérivant l'énergie libre $\mathcal{L}[q]$ par rapport à la distribution variationnelle des sources $q_{\mathbf{S}}$, en supposant que toutes les autres distributions variationnelles $q_{\mathbf{W}}$, $q_{\mathbf{H}}$ et $q_{\mathbf{K}}$ sont constantes. De ce fait, l'énergie libre peut être réécrite en ne conservant que les termes dépendant de la distribution variationnelle des sources $q_{\mathbf{S}}$ telle que

$$\begin{aligned}\mathcal{L}[q] = &\int q_{\mathbf{S}}(\mathbf{S}) \left\{ \iiint q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})q_{\mathbf{K}}(\mathbf{K}) \log p(\mathbf{X}, \mathbf{Z}) d\mathbf{W} d\mathbf{H} d\mathbf{K} \right\} d\mathbf{S} \\ &- \int q_{\mathbf{S}}(\mathbf{S}) \log q_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} + \text{cte}.\end{aligned}$$

En posant

$$\begin{aligned}\log \tilde{p}(\mathbf{X}, \mathbf{S}) &= \iiint q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) q_{\mathbf{K}}(\mathbf{K}) \log p(\mathbf{X}, \mathbf{Z}) d\mathbf{W} d\mathbf{H} d\mathbf{K} + \text{cte} \\ &= \mathbb{E}_{\mathbf{Z}\setminus\mathbf{S}}[\log p(\mathbf{X}, \mathbf{Z})] + \text{cte},\end{aligned}$$

l'énergie libre peut se réécrire

$$\mathcal{L}[q] = \int q_{\mathbf{S}}(\mathbf{S}) \log \frac{\tilde{p}(\mathbf{X}, \mathbf{S})}{q_{\mathbf{S}}(\mathbf{S})} d\mathbf{S}. \tag{E.18}$$

Rappelant alors que la distribution variationnelle $q_{\mathbf{S}}$ doit vérifier $\int q_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} = 1$, la distribution variationnelle optimale $q^*_{\mathbf{S}}$ est obtenue en dérivant le lagrangien de l'énergie libre

$$\mathcal{L}'[q] = \mathcal{L}[q] + \lambda \left(\int q_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} - 1 \right), \tag{E.19}$$

où λ est le multiplicateur de Lagrange. L'équation d'Euler-Lagrange nous dit alors que pour qu'une distribution $q^*_{\mathbf{S}}(\mathbf{S})$ soit un maximum local de $\mathcal{L}'[q]$, il faut que la dérivée de l'intégrande soit égale à 0 pour tout $\mathbf{S}$:

$$\forall \mathbf{S}, \quad \log \frac{\tilde{p}(\mathbf{X}, \mathbf{S})}{q^*_{\mathbf{S}}(\mathbf{S})} + \lambda - 1 = 0 \tag{E.20}$$

Cette équation nous donne alors l'expression de la distribution variationnelle optimale en fonction de λ

$$q^*_{\mathbf{S}}(\mathbf{S}) = \mathrm{e}^{\lambda - 1} \, \tilde{p}(\mathbf{X}, \mathbf{S}). \tag{E.21}$$

Étant donné que $\int \tilde{p}(\mathbf{X}, \mathbf{S}) = 1$, la contrainte $\int q^*_{\mathbf{S}}(\mathbf{S}) d\mathbf{S} = 1$ nous permet alors de conclure que $\lambda^* = 1$ et que

$$q^*_{\mathbf{S}}(\mathbf{S}) = \tilde{p}(\mathbf{X}, \mathbf{S}), \tag{E.22}$$

soit

$$\log q^*_{\mathbf{S}}(\mathbf{S}) = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{S}}[\log p(\mathbf{X}, \mathbf{Z})] + \text{cte}. \tag{E.23}$$

E.2.2 Distribution variationnelle des paramètres de NMF

Étant donné que les distributions variationnelles des paramètres de NMF sont conditionnées au nombre de composantes $\mathbf{K}$, le calcul des distributions variationnelles maximisant l'énergie libre est légèrement modifié, bien qu'il suive le même principe.

Prenons pour exemple le calcul de la distribution variationnelle optimale $q^*_{\mathbf{W}}(\mathbf{W}|\mathbf{K})$. Comme pour la distribution variationnelle des sources, nous supposons que toutes les distributions variationnelles autres que $q_{\mathbf{W}}$ sont fixées. De ce fait, l'expression de l'énergie libre peut se réécrire, en ne gardant que les termes dépendant de la distribution $q_{\mathbf{W}}$,

$$\begin{aligned}\mathcal{L}[q] = &\iint q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) q_{\mathbf{K}}(\mathbf{K}) \left\{ \iint q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) q_{\mathbf{S}}(\mathbf{S}) \log p(\mathbf{X}, \mathbf{Z}) d\mathbf{S} d\mathbf{H} \right\} d\mathbf{K} d\mathbf{W} \\ &- \iint q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) q_{\mathbf{K}}(\mathbf{K}) \log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) d\mathbf{K} d\mathbf{W} + \text{cte}.\end{aligned}$$

Nous posons cette fois

$$\begin{aligned}\log \tilde{p}(\mathbf{X}, \mathbf{W}, \mathbf{K}) &= \iint q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) q_{\mathbf{S}}(\mathbf{S}) \log p(\mathbf{X}, \mathbf{Z}) d\mathbf{S} d\mathbf{H} + \text{cte} \\ &= \mathbb{E}_{\mathbf{Z}\setminus\{\mathbf{W},\mathbf{K}\}}[\log p(\mathbf{X}, \mathbf{Z})] + \text{cte}\end{aligned}$$

afin de simplifier l'écriture de l'énergie libre en

$$\mathcal{L}[q] = \iint q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) q_{\mathbf{K}}(\mathbf{K}) \log \frac{\tilde{p}(\mathbf{X}, \mathbf{W}, \mathbf{K})}{q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})} d\mathbf{K} d\mathbf{W} + \text{cte}. \tag{E.24}$$

Nous procédons ensuite comme pour la distribution variationnelle des sources, en écrivant le lagrangien de l'énergie libre par rapport à la distribution $q_{\mathbf{W}}$. Étant donnée la contrainte de normalisation $\forall \mathbf{K}, \int q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) d\mathbf{W} = 1$, le lagrangien s'écrit cette fois

$$\mathcal{L}'[q] = \mathcal{L}[q] + \int \lambda_{\mathbf{K}} \left(\int q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) d\mathbf{W} - 1 \right) d\mathbf{K}. \tag{E.25}$$

L'équation d'Euler-Lagrange nous donne alors la condition :

$$\forall \mathbf{K}, \forall \mathbf{W}, \quad \log \frac{\tilde{p}(\mathbf{X}, \mathbf{W}, \mathbf{K})}{q^*_{\mathbf{W}}(\mathbf{W}|\mathbf{K})} + \lambda_{\mathbf{K}} - 1 = 0. \tag{E.26}$$

Puisque $\tilde{p}(\mathbf{X}, \mathbf{W}, \mathbf{K})$ est normalisée, nous pouvons déduire que $\forall \mathbf{K}, \lambda_{\mathbf{K}} = 1$ et

$$\forall \mathbf{K}, \quad \log q^*_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) = \log \tilde{p}(\mathbf{X}, \mathbf{W}, \mathbf{K}) = \mathbb{E}_{\mathbf{Z} \setminus \{\mathbf{W}, \mathbf{K}\}}[\log p(\mathbf{X}, \mathbf{Z})] + \text{cte}. \tag{E.27}$$

La distribution variationnelle optimale $q^*_{\mathbf{H}}(\mathbf{H}|\mathbf{K})$ peut être obtenue de la même façon.

E.2.3 Distribution variationnelle du nombre de composantes

Enfin, nous procédons à nouveau de la même façon pour déterminer la distribution variationnelle $q^*_{\mathbf{K}}(\mathbf{K})$ qui maximise l'énergie libre. En supposant que toutes les autres distributions variationnelles sont constantes, nous pouvons réécrire l'énergie libre en ne gardant que les termes dépendant de $q_{\mathbf{K}}$: telle que

$$\begin{aligned}
\mathcal{L}[q] = & \int q_{\mathbf{K}}(\mathbf{K}) \left\{ \iiint q_{\mathbf{S}}(\mathbf{S}) q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) \log p(\mathbf{X}, \mathbf{Z} d\mathbf{S} d\mathbf{W} d\mathbf{H} \right\} d\mathbf{K} \\
& - \iint q_{\mathbf{K}}(\mathbf{K}) q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) \log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) d\mathbf{W} d\mathbf{K} \\
& - \iint q_{\mathbf{K}}(\mathbf{K}) q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) \log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) d\mathbf{H} d\mathbf{K} \\
& - \int q_{\mathbf{K}}(\mathbf{K}) \log q_{\mathbf{K}}(\mathbf{K}) d\mathbf{K} + \text{cte}.
\end{aligned}$$

En posant

$$\begin{aligned}
\log \tilde{p}(\mathbf{X}, \mathbf{K}) &= \iiint q_{\mathbf{S}}(\mathbf{S}) q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) \log p(\mathbf{X}, \mathbf{Z}) d\mathbf{S} d\mathbf{W} d\mathbf{H} + \text{cte} \\
&= \mathbb{E}_{\mathbf{Z} \setminus \mathbf{K}}[\log p(\mathbf{X}, \mathbf{Z})] + \text{cte},
\end{aligned}$$

l'énergie libre peut alors se réécrire

$$\mathcal{L}[q] = \int q_{\mathbf{K}}(\mathbf{K}) \left(\log \frac{\tilde{p}(\mathbf{X}, \mathbf{K})}{q_{\mathbf{K}}(\mathbf{K})} - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] \right) d\mathbf{K} + \text{cte}. \tag{E.28}$$

avec

$$\mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] = \int q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) \log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K}) d\mathbf{W}$$

et

$$\mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] = \int q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) \log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K}) d\mathbf{H}.$$

La contrainte de normalisation $\int q(\mathbf{K})d\mathbf{K} = 1$ nous oblige à utiliser un multiplicateur de Lagrange λ et le lagrangien s'écrit donc

$$\mathcal{L}'[q] = \mathcal{L}[q] + \lambda \left(\int q(\mathbf{K})d\mathbf{K} - 1 \right). \tag{E.29}$$

En invoquant l'équation d'Euler-Lagrange, nous savons que la distribution optimale $q^*_{\mathbf{K}}(\mathbf{K})$ satisfait :

$$\forall \mathbf{K}, \quad \log \frac{\tilde{p}(\mathbf{X}, \mathbf{K})}{q^*_{\mathbf{K}}(\mathbf{K})} - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \lambda - 1 = 0. \tag{E.30}$$

Nous en déduisons la valeur du multiplicateur de Lagrange

$$\lambda^* = 1 - \log \int \tilde{p}(\mathbf{X}, \mathbf{K}) \, \mathrm{e}^{-\mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})]} \, d\mathbf{K}$$

et la distribution variationnelle optimale

$$\begin{aligned} \log q^*_{\mathbf{K}}(\mathbf{K}) &= \log \tilde{p}(\mathbf{X}, \mathbf{K}) - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \text{cte} \\ &= \mathbb{E}_{\mathbf{Z}\setminus\mathbf{K}}[\log p(\mathbf{X}, \mathbf{Z})] - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \text{cte}. \end{aligned}$$

E.3 Justification du paramètre de contrôle de l'entropie

Afin de rendre la fusion effective, nous avons proposé dans la partie 5.4.2 d'introduire un paramètre $\beta \geq 1$ de sorte que la probabilité *a posteriori* du nombre de composantes K_{1m} soit donnée par

$$q(K_{1m}) \propto \mathrm{e}^{\mathcal{L}_m/\beta} \tag{E.31}$$

où $\mathcal{L}_m$ est l'équivalent de l'énergie libre d'un modèle de NMF à ordre unique ayant pour nombre de composantes $K_1 = K_{1m}$. Nous proposons ici de montrer que cette proposition revient à pénaliser l'entropie de la distribution *a posteriori* du nombre de composantes $q_{\mathbf{K}}(\mathbf{K})$ dans l'expression de l'énergie libre. Pour cela, nous reprendrons les notations utilisées dans la partie E.2 précédente pour le calcul des distributions variationnelles maximisant l'énergie libre.

L'entropie de la distribution *a posteriori* du nombre de composantes s'écrit

$$\mathbb{E}_{\mathbf{K}}[\log q_{\mathbf{K}}(\mathbf{K})] = -\int q_{\mathbf{K}}(\mathbf{K}) \log q_{\mathbf{K}}(\mathbf{K}) d\mathbf{K}. \tag{E.32}$$

Nous allons à présent dériver l'expression de la distribution variationnelle $q^*_{\mathbf{K}}(\mathbf{K})$ maximisant l'énergie libre $\mathcal{L}_\beta[q]$, qui diffère de l'énergie libre $\mathcal{L}[q]$ par l'ajout du terme β pénalisant l'entropie (E.32) de la distribution variationnelle $q_{\mathbf{K}}(\mathbf{K})$. Nous supposerons donc que toutes les autres distributions variationnelles sont constantes, de sorte que l'énergie libre peut s'écrire, similairement à l'équation (E.28),

$$\begin{aligned} \mathcal{L}_\beta[q] = \int q_{\mathbf{K}}(\mathbf{K}) \Bigg(& \log \tilde{p}(\mathbf{X}, \mathbf{K}) - \beta \log q_{\mathbf{K}}(\mathbf{K}) \\ & - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] \Bigg) d\mathbf{K} + \text{cte}. \end{aligned}$$

Nous noterons que la seule différence entre cette équation et l'équation (E.28) originale réside bien dans l'ajout du terme β qui pénalise l'entropie (E.32).

Procédons alors à la dérivation du lagrangien de l'énergie libre $\mathcal{L}_\beta[q]$ par rapport à $q_{\mathbf{K}}$ et par rapport au multiplicateur de Lagrange assurant que $\int q_{\mathbf{K}}(\mathbf{K})d\mathbf{K} = 1$, comme nous l'avons fait dans la partie E.2. Comme auparavant, l'équation d'Euler-Lagrange nous dit que la distribution variationnelle optimale $q^*_{\mathbf{K}}(\mathbf{K})$ vérifie pour tout $\mathbf{K}$:

$$\log \tilde{p}(\mathbf{X},\mathbf{K}) - \beta \log q^*_{\mathbf{K}}(\mathbf{K}) - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \lambda - \beta = 0. \quad \text{(E.33)}$$

En réarrangeant ces termes et en en prenant l'exponentielle, nous pouvons exprimer la distribution variationnelle $q_{\mathbf{K}}(\mathbf{K})$ selon

$$q^*_{\mathbf{K}}(\mathbf{K}) = \mathrm{e}^{(\lambda - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] - \beta)/\beta}\, \tilde{p}(\mathbf{X},\mathbf{K})^{1/\beta}. \quad \text{(E.34)}$$

Finalement, étant donné que $\int q^*_{\mathbf{K}}(\mathbf{K})d\mathbf{K} = 1$, le multiplicateur de Lagrange optimal λ^* s'écrit

$$\lambda^* = \beta \left(1 - \log \int \tilde{p}(\mathbf{X},\mathbf{K})^{1/\beta}\, \mathrm{e}^{-\mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})]}\, d\mathbf{K}\right). \quad \text{(E.35)}$$

En injectant cette expression dans l'équation (E.34), nous obtenons alors l'expression de la distribution variationnelle maximisant l'énergie libre $\mathcal{L}_\beta[q]$

$$\begin{aligned}\log q^*_{\mathbf{K}}(\mathbf{K}) &= \frac{1}{\beta}\left(\log \tilde{p}(\mathbf{X},\mathbf{K}) - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \text{cte}\right) \\ &= \frac{1}{\beta}\left(\mathbb{E}_{\mathbf{Z}\setminus\mathbf{K}}[\log p(\mathbf{X},\mathbf{Z})] - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \text{cte}\right).\end{aligned}$$

Enfin, en posant $\mathcal{L}_m = \mathbb{E}_{\mathbf{Z}\setminus\mathbf{K}}[\log p(\mathbf{X},\mathbf{Z})] - \mathbb{E}_{\mathbf{W}}[\log q_{\mathbf{W}}(\mathbf{W}|\mathbf{K})] - \mathbb{E}_{\mathbf{H}}[\log q_{\mathbf{H}}(\mathbf{H}|\mathbf{K})] + \text{cte}$ et en notant que cette quantité est similaire à l'énergie libre d'un modèle NMF à ordre simple, nous avons montré qu'en pénalisant l'entropie de la distribution variationnelle $q_{\mathbf{K}}(\mathbf{K})$ dans l'expression de l'énergie libre du modèle NMF à ordre multiple, la distribution variationnelle optimale du nombre de composantes prenait la forme

$$q^*_{\mathbf{K}}(\mathbf{K}) \propto \mathrm{e}^{\mathcal{L}_m/\beta}. \quad \text{(E.36)}$$

Remerciements

N'ayant réussi qu'à balbutier quelques mots le jour de ma soutenance, j'espère ici remercier du mieux que je peux celles et ceux qui m'ont permis, chacun à leur manière, de mener à bien ce travail.

⁂

En tout premier lieu, je remercie très chaleureusement mes directeurs de thèse, Emmanuel Vincent et Gaël Richard, pour leur confiance, leur soutien indéfectible et leurs encouragements maintes fois renouvelés. J'ai apprécié travailler à leurs côtés et j'ai beaucoup appris, tant sur le plan technique que sur le plan humain. Merci Gaël pour m'avoir permis de continuer cette thèse malgré un départ chaotique. Merci Emmanuel d'avoir accepté de rejoindre cette aventure en cours de route et de t'y être investi pleinement.

Je n'oublie bien sûr pas Pierre Leveau sans qui ce travail n'aurait pas vu le jour. Merci de m'avoir accompagné pendant un an. Merci également pour cette chance que tu m'as offerte de travailler pendant plus de deux ans chez Audionamix.

⁂

Je remercie sincèrement les membres du jury, Laurent Girin, Jérôme Idier, Jean-Luc Zarader, Pierre Leveau et Jonathan Le Roux, pour l'intérêt qu'ils ont porté à ce travail. Nos échanges, vos remarques, et plus généralement le regard extérieur que vous avez posé sur mon travail, m'ont permis d'apprendre plus encore, et ce même après la fin de la rédaction de cette thèse.

⁂

Mille mercis aux membres du groupe AAO pour leur bonne humeur générale, leur sympathie, les nombreux services rendus et les encouragements.

Merci aux permanents : Slim Essid, Bertrand David, Roland Badeau, Alexandre Gramfort et Yves Grenier. Tous ont su, à un moment ou à un autre, m'éclairer de leur expertise avec patience et pédagogie.

Merci aux anciens qui m'ont accueilli : Mounira, Angélique, Nicolas, François, Antoine, Thomas et Sébastien. Depuis le début de ma scolarité, j'ai toujours été inspiré par les « grands », ceux des classes supérieures. Vous n'avez pas dérogé à cette règle. Bien au contraire.

Merci aux « petits » nouveaux : Clément, Paul, Simon D., Romain, Hequn, Floriane, Simon L., Victor, Arthur, Reda, Tom, Yousra et Mainak. Les nombreux moments passés à vos côtés, sur la terrasse ou chez Jean Monnet, m'ont aidé à tenir bon, et ce de façon inestimable.

Merci à ceux qui ont partagé de très près ces trois années (et quelques mois) : Anne-Claire et Aymeric. Comme on dit chez moi : *atxik* !

Merci aussi aux collègues de passage : Karan, Alexis, Teon et Rachel. Merci à mes co-bureaux, Davide, Liying, Eric, Manoj et Chirag, pour avoir supporté mon impétuosité face aux aléas informatiques.

Merci à mes collègues de Nancy : Dung, Yann, Nathan, Motaz, Arseniy et Cyrine.

Je tiens également à remercier très sincèrement pour leur soutien logistique, toujours aimable : Marie-Laure Chauveaux, Laurence Zelmar, Fabrice Planche et Ariel Vives.

Je garde de vous tous un souvenir mémorable et une reconnaissance sincère. J'en veux pour preuve mes anciens collègues de chez Audionamix, que j'aimerais ici également remercier pour le plaisir que j'ai eu à travailler avec eux, ou plus simplement, pour leur amitié : Simon, Sergio, Juan Jo, Guillaume V., Marilyn, Romain, Guillaume M. et bien sûr Telma.

Mentions spéciales : à Aymeric pour son amitié, son oreille attentive, toujours à l'écoute dans les moments de doutes, et son altruisme exemplaire ; à Paul pour toutes les gentillesses échangées, pour m'avoir accueilli quand j'étais sans logis et pour ses frères ; à Clément pour avoir partagé son savoir-faire en levures de tout genre.

⁂

Malgré mon humeur changeante, malgré mon indisponibilité chronique, je ne peux que me réjouir que mes amis qui étaient là avant la thèse le soient encore aujourd'hui ! Je vous remercie donc tous : Alex, Willou, Loulou, Jeannot, Florette, Gotwi, Doux, Mawi, Laulau, Ben, Pierrot, Nono, Mumu, Marco, Hache et Clountch. Sachez que vous pouvez aussi compter sur moi.

Merci aussi à Émilie, Alice, Julie et Anne.

Merci également à Dong-Jun, pour son amitié et pour l'amour de la musique qu'il n'a de cesse de partager.

⁂

Pour conclure, je vais inévitablement manquer de mots pour dire toute la reconnaissance et toute l'affection que j'ai pour ma famille. Tous vos témoignages de soutien m'ont particulièrement touché. J'ai toujours su que je pouvais compter sur vous et vous n'avez jamais manqué les occasions pour me le rappeler. Je saisis donc cette chance pour vous dire un simple mais très franc merci. À mes parents, mes sœurs Ana et Maialen et mon frère Gillen. À ma marraine Odile. À mes oncles et tantes, Renée et Philippe, Bena et Michel, Margaita et Jean-Claude, Beatrix et Jean-Pierre. À mes cousins et cousines, Miren, Elorri, Guillaume et Françis. À Thibaud, Sylvain, Maddalen et Txema. À Pauline.

Mes pensées les plus tendres vont enfin à mes grand-mères, Amañi et Mamie, à qui je dédie ce mémoire.

FUSION POUR LA SÉPARATION DE SOURCES AUDIO

Xabier JAUREGUIBERRY

RESUME : La séparation aveugle de sources audio dans le cas sous-déterminé est un problème mathématique complexe dont il est aujourd'hui possible d'obtenir une solution satisfaisante pour certaines applications industrielles, à condition de sélectionner la méthode la plus adaptée au problème posé et de savoir paramétrer celle-ci soigneusement. Afin d'automatiser cette étape de sélection déterminante, nous proposons dans cette thèse de recourir au principe de fusion, très populaire dans le domaine de la classification mais encore peu exploité en séparation de sources. L'idée est simple : il s'agit, pour un problème donné, de sélectionner plusieurs méthodes de résolution plutôt qu'une seule et de les combiner afin d'en améliorer la solution.

Pour cela, nous introduisons un cadre général de fusion qui consiste à formuler l'estimée d'une source comme la combinaison de plusieurs estimées de cette même source données par différents algorithmes de séparation, chaque estimée étant pondérée par un coefficient de fusion. Ces coefficients peuvent notamment être appris sur un ensemble d'apprentissage représentatif du problème posé par minimisation d'une fonction de coût liée à l'objectif de séparation. Pour aller plus loin, nous proposons également deux approches permettant d'adapter les coefficients de fusion au signal à séparer. La première formule la fusion de modèles de factorisation en matrices non-négatives (NMF) dans un cadre bayésien, à la manière du moyennage bayésien de modèles. La deuxième exploite la puissance d'apprentissage des réseaux de neurones profonds afin de déterminer des coefficients de fusion variant en temps.

Toutes ces approches ont été évaluées sur deux corpus distincts : l'un dédié au rehaussement de la parole, l'autre dédié à l'extraction de voix chantée. Quelle que soit l'approche considérée, nos résultats montrent l'intérêt systématique de la fusion par rapport à la simple sélection, la fusion adaptative par réseau de neurones se révélant être la plus performante.

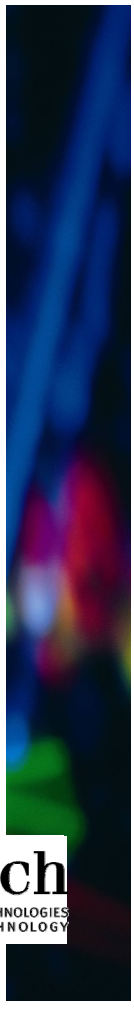
MOTS-CLEFS: fusion de modèles, aggrégation de modèles, combinaison de modèles, sélection de modèles, séparation de sources audio, rehaussement de la parole, extraction de voix chantée, factorisation en matrices non-négatives, NMF, inférence variationnelle bayésienne, moyennage bayésien de modèles, apprentissage profond, réseaux de neurones profonds

ABSTRACT: Underdetermined blind source separation is a complex mathematical problem that can be satisfyingly resolved for some practical applications, providing that the right separation method has been selected and carefully tuned. In order to automate this selection process, we propose in this thesis to resort to the principle of fusion which has been widely used in the related field of classification yet is still marginally exploited in source separation. Fusion consists in combining several methods to solve a given problem instead of selecting a unique one.

To do so, we introduce a general fusion framework in which a source estimate is expressed as a linear combination of estimates of this same source given by different separation algorithms, each source estimate being weighted by a fusion coefficient. For a given task, fusion coefficients can then be learned on a representative training dataset by minimizing a cost function related to the separation objective. To go further, we also propose two ways to adapt the fusion coefficients to the mixture to be separated. The first one expresses the fusion of several non-negative matrix factorization (NMF) models in a Bayesian fashion similar to Bayesian model averaging. The second one aims at learning time-varying fusion coefficients thanks to deep neural networks.

All proposed methods have been evaluated on two distinct corpora. The first one is dedicated to speech enhancement while the other deals with singing voice extraction. Experimental results show that fusion always outperform simple selection in all considered cases, best results being obtained by adaptive time-varying fusion with neural networks.

KEY-WORDS: model fusion, model aggregation, model combination, model selection, audio source separation, speech enhancement, singing voice extraction, non-negative matrix factorization, NMF, variational Bayesian inference, Bayesian model averaging, deep learning, deep neural networks



TELECOM ParisTech

EDITE

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY