Processus gaussiens pour la séparation de sources et le codage informé

Antoine Liutkus

2012-ENST-069

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal et Images »

présentée et soutenue publiquement par

Antoine LIUTKUS

le 27 novembre 2012

Processus gaussiens pour la séparation de sources et le codage informé

Directeur de thèse : **Roland BADEAU**
Co-encadrement de la thèse : **Gaël RICHARD**

Jury

M. Laurent GIRIN, Professeur, Grenoble INP, Université J. Fourier	Président
M. Jérôme IDIER, Professeur, IRCCyN École Centrale de Nantes	Rapporteur
M. Christian JUTTEN, Directeur de recherches, Grenoble INP, Université J. Fourier	Rapporteur
M. Taylan CEMGIL, Professeur, Bogazici University, Istambul	Examinateur
M. Emmanuel VINCENT, Chargé de recherches, INRIA Rennes	Examinateur
M. Roland BADEAU, Maître de conférences, Télécom ParisTech	Directeur de thèse
M. Gaël RICHARD, Professeur, Télécom ParisTech	Directeur de thèse

TELECOM ParisTech
école de l'Institut Télécom - membre de ParisTech

THÈSE

Résumé

La séparation de sources est la tâche qui consiste à récupérer plusieurs signaux dont on observe un ou plusieurs mélanges. Dans le cas des signaux audio, elle consiste à récupérer la piste jouée par chacun des instruments à partir de l'observation du morceau mixé. Ce problème est particulièrement difficile du fait du grand nombre d'inconnues qu'il laisse à estimer, bien plus nombreuses que le nombre d'observations disponibles. De manière à rendre la séparation possible, toute information supplémentaire connue sur les sources ou le mélange doit pouvoir être prise en compte pour résoudre l'indétermination du problème.

Dans cette thèse, je propose un formalisme général permettant d'inclure un grand nombre de connaissances dans les problèmes de séparation de sources. Dans ce formalisme, une source est modélisée comme la réalisation d'un processus gaussien, entendu comme un espace non paramétrique de fonctions. La séparation de mélanges se fait alors grâce à des techniques classiques de régression. L'approche a de nombreux intérêts. Tout d'abord, elle généralise une grande partie des méthodes actuelles et permet leur extension à des domaines de définition quelconques. Ensuite, elle permet la prise en compte immédiate de nombreux a prioris sur les sources, tels que la continuité, la périodicité, la stationnarité ou le timbre. De plus, les sources estimées minimisent l'erreur quadratique moyenne. Enfin, les différents paramètres du modèle peuvent être estimés automatiquement grâce à de puissantes approches probabilistes.

Ce cadre théorique de la séparation de processus gaussiens est ensuite appliqué à la séparation informée de sources audio. Dans cette configuration, on considère que la séparation peut être assistée par la connaissance supplémentaire d'une information annexe sur les signaux. Dans le cas de la musique, de nombreuses informations annexes peuvent être envisagées telles que la tonalité, le tempo, une partition, etc. On verra que le formalisme gaussien proposé permet de prendre en compte naturellement de telles connaissances. Je concentrerai mon attention sur un cas précis de séparation informée, pour lequel l'information annexe a été calculée en amont de la séparation, lors d'une phase préliminaire où à la fois le mélange et les sources sont disponibles. En pratique cela se produit en studio lorsque les mélanges ont été obtenus à partir des sources enregistrées auprès des musiciens. L'intérêt de l'approche est de permettre de récupérer de bonnes sources estimées à partir de la seule connaissance des mélanges et de l'information annexe. Pour peu que celle-ci puisse se coder efficacement, cela rend possible des applications comme le karaoké ou le traitement séparé des différentes sources à un coût réduit en débit.

Il nous est apparu que le problème de la séparation informée s'apparente fortement à un problème de codage multicanal. Dans cette perspective, le codeur a accès à la fois aux sources et aux mélanges, tandis que le décodeur n'a accès qu'aux mélanges et à l'information annexe générée par le codeur. En sortie du décodeur, on obtient les sources séparées. Cette analogie permet de placer la séparation informée dans un cadre théorique plus global où elle devient un problème de codage particulier et bénéficie à ce titre des résultats classiques de la théorie du codage, qui permettent d'optimiser efficacement les performances.

à Raphaël Blouet

Du moment qu'une de leurs idées est vraie, elle n'est plus leur bien propre. Elle appartient en commun à tous les amants de la vérité.

Saint Augustin

Remerciements

Avant toute chose, je voudrais remercier mes parents pour leur affection constante et sans faille ainsi que pour m'avoir toujours soutenu en toute circonstance.

Je tiens également à remercier mes directeurs de thèse Roland Badeau et Gaël Richard de m'avoir offert l'opportunité de réaliser ce doctorat à Télécom ParisTech. Leur compétence, leur implication, leur bonne humeur et la confiance dont ils ont fait preuve à mon égard ont fait de ces trois années passées au laboratoire LTCI une expérience extraordinaire, tant sur le plan professionnel que personnel. Je remercie tous les membres du groupe AAO, à la fois pour la qualité des échanges scientifiques que j'ai eus avec eux, mais aussi pour leur constante bonne humeur et pour l'ambiance exceptionnelle dans laquelle se sont déroulées ces années de doctorat. Rémi, Thomas, Manuel, Benoit, Angélique, Mounira, François, Romain, Nicolas, Cyril, Aymeric, Sébastien, Roland, Gaël, Slim, Bertrand, Yves : vous voir au bureau a souvent été une motivation plus forte que les processus gaussiens pour me lever le matin [1].

Mon doctorat s'est déroulé dans le cadre du projet DReaM, financé par l'Agence Nationale de la Recherche. Ce projet a été l'occasion d'une collaboration scientifique remarquable entre plusieurs équipes de recherche. En particulier, je remercie Laurent Girin pour son enthousiasme au sujet de la séparation informée et pour m'avoir plusieurs fois invité à travailler avec lui au GIPSA-lab. Chacune de ces visites donna une impulsion considérable à mon travail. La présentation de la séparation informée qu'on trouvera dans ce document est le fruit de trois années de maturation en commun et je souligne ici le rôle qu'y ont joué Laurent Girin, Sylvain Marchand, Laurent Daudet, Nicolas Stürmel, Stanislaw Gorlow et Dominique Fourer.

J'exprime ici ma profonde reconnaissance à Alexey Ozerov pour l'intérêt tout particulier qu'il a porté depuis le début à mes travaux sur la séparation informée, pour ses idées sur le codage de sources informé et pour avoir pris la peine de relire ce mémoire. De nombreuses idées exprimées ici sont le fruit de notre travail en commun. *The interchange often consisted of my describing a problem, his generation of possible avenues of solution, and then my going off to work for a few weeks to understand his suggestions and work them through* (Robert M. Gray, *Entropy and Information Theory*).

Je remercie Patrick Devriendt, Sébastien Maizy et Maria Trocan pour m'avoir donné l'opportunité d'effectuer des enseignements à l'ESME Sudria et à l'ISEP. En plus du plaisir que j'ai eu à nos conversations, ce fut pour moi une expérience très agréable que de découvrir en parallèle des joies de la recherche celles, non moins enrichissantes, de l'enseignement.

Sur un plan plus personnel, je remercie mes sœurs Aurélia, Marie et Mélanie pour leur présence, leur soutien et pour la fierté que j'ai à voir grandir mes sept neveux et nièces. Je remercie mon frère Bastien pour avoir toujours été là pour moi, pour m'avoir transmis sa joie de vivre et pour sa profonde sagesse. Je salue tous mes amis et tous les convives des repas du mercredi. Parmi eux, je remercie en particulier Matthieu, David, Lionel et Eric pour leur inestimable amitié.

Si mon doctorat a occupé mes jours et parfois mes nuits, Julie a occupé mon cœur. Je la remercie pour son amour, ses encouragements et pour les six années extraordinaires que nous avons déjà passées ensemble. Puisse rien ne pouvoir nous séparer.

1. Et Zafar et Gilles, même s'ils ne sont pas de notre lab.

Table des matières

Liste des symboles

Ensemble de définition

$\mathbb{T}$	Ensemble de définition des sources.
D	Dans les parties I et II : dimension de $\mathbb{T}$ si $\mathbb{T} = \mathbb{Z}^D$. $D = 1$ pour les séries temporelles
T	Ensemble discret $T = [t_1, \ldots, t_L] \in \mathbb{T}^L$ de L points de $\mathbb{T}$
L	Nombre d'éléments de T

Formes d'ondes sources, mélanges

I, J	Nombre de mélanges, de sources
$\tilde{s}$, $\tilde{x}$	Processus gaussiens sources et mélanges, définis sur $\mathbb{T}$, à valeurs dans $\mathbb{C}$ ou $\mathbb{R}$ (variables aléatoires)
$\tilde{s}(t, j)$	Valeur de la $j^{ème}$ source à la position t (variable aléatoire)
$\tilde{\boldsymbol{s}}(T, \cdot)$	Réalisation des processus sources sur $T \in \mathbb{T}^L$. Matrice de dimension $L \times J$ ou sa vectorisation de dimension $LJ \times 1$ selon le contexte.
$\tilde{\boldsymbol{s}}(t, \cdot)$	Réalisation de toutes les sources à la position t, vecteur de dimension $1 \times J$
$\tilde{\boldsymbol{s}}(t, j)$	Réalisation de la source j, prise à la position t

Formes d'ondes images

$\tilde{y}(\cdot, \cdot, j)$	Processus image de la source j. Défini sur $\mathbb{T} \times \mathbb{N}_I$, à valeurs dans $\mathbb{C}$ ou $\mathbb{R}$
$\tilde{y}(t, i, j)$	Image de la source j dans le $i^{ème}$ canal du mélange. Variable aléatoire
$\tilde{\boldsymbol{y}}(T, \cdot, j)$	Réalisation sur $T \in \mathbb{T}^L$ de l'image de la source j. Matrice de dimension $L \times I$

Tramage

$\mathbb{T}_0$	Domaine de définition d'une trame
$\mathcal{G}\{\tilde{s}\}$	Tramage du processus $\tilde{s}$. Variable aléatoire définie sur $\mathbb{T}_0 \times \mathbb{Z}^D$ et à valeurs dans $\mathbb{C}$ ou $\mathbb{R}$
$\mathcal{N}_\mathcal{G}, N$	Pour un ensemble $T \in \mathbb{T}^L$, ensemble des indices des trames dans lesquelles apparaît au moins un point de T. Leur nombre
$\mathcal{G}\{\tilde{\boldsymbol{s}}\}$	Tramage d'une réalisation du processus $\tilde{s}$
$\mathcal{G}^{-1}\left\{\hat{S}\right\}$	Addition-recouvrement : reconstruction d'un signal défini sur $\mathbb{T}$ à partir d'un tramage

Transformées de Fourier à court terme (TFCT)

$\mathcal{F}_D\{\tilde{s}\}$	Transformée de Fourier en dimension D d'un signal $\tilde{s}$ défini sur $\mathbb{Z}^D$ et à valeurs dans $\mathbb{C}$ ou $\mathbb{R}$. Si $\tilde{s}$ est à valeurs réelles, seuls les indices des fréquences positives sont conservés
F, $\mathbb{F}$	Nombre des indices fréquentiels non redondants d'une transformée de Fourier. Ensemble de ces indices
s	Transformées de Fourier à court terme (TFCT) des formes d'ondes des sources $\tilde{s}$. Variables aléatoires définies sur $\mathbb{F} \times \mathbb{Z}^D \times \mathbb{N}_J$ et à valeurs dans $\mathbb{C}$
x	TFCT des processus mélanges $\tilde{x}$. Variables aléatoires définies sur $\mathbb{F} \times \mathbb{Z}^D \times \mathbb{N}_I$ et à valeurs dans $\mathbb{C}$
$\boldsymbol{f}$, f	Indice fréquentiel. De dimension $D \times 1$. Sa notation particulière pour $D = 1$
$\boldsymbol{n}$, n	Indice de trame. De dimension $D \times 1$. Sa notation particulière pour $D = 1$
$s(\boldsymbol{f}, \boldsymbol{n}, j)$	Valeur de la TFCT du processus $\tilde{s}(\cdot, j)$ au point $(\boldsymbol{f}, \boldsymbol{n})$. Variable aléatoire
$s(\boldsymbol{f}, \boldsymbol{n}, \cdot)$	Valeur des TFCTs des processus sources au point $(\boldsymbol{f}, \boldsymbol{n})$. Vecteur aléatoire de dimension $J \times 1$
$y(\boldsymbol{f}, \boldsymbol{n}, \cdot, j)$	Valeur des TFCTs de l'image de la source j au point $(\boldsymbol{f}, \boldsymbol{n})$. Vecteur aléatoire de dimension $I \times 1$
$\boldsymbol{s}$, $\boldsymbol{x}$, $\boldsymbol{y}$	Réalisation des TFCT des processus sources, mélanges et images
$P(\boldsymbol{f}, \boldsymbol{n}, j)$	Densité spectrale de puissance de la source j au point $(\boldsymbol{f}, \boldsymbol{n})$
$v_s(\boldsymbol{f}, \boldsymbol{n}, j)$	Spectrogramme de puissance de la source j au point $(\boldsymbol{f}, \boldsymbol{n})$: $v_s(\boldsymbol{f}, \boldsymbol{n}, j) = \lvert s(\boldsymbol{f}, \boldsymbol{n}, j)\rvert^2$
$v_x(\boldsymbol{f}, \boldsymbol{n}, i)$	Spectrogramme de puissance du mélange i au point $(\boldsymbol{f}, \boldsymbol{n})$

Modèles de mixage

A	Matrice de mélange dans le cas linéaire instantané, de dimension $I \times J$
H	Ordre des filtres de mixage dans le cas convolutif
$a_{ij}(\cdot)$	Réponse impulsionnelle des filtres de mélange dans le cas convolutif
$A(\boldsymbol{f})$	Matrice de mélange à l'indice de fréquence $\boldsymbol{f}$ dans le cas convolutif. De dimension $I \times J$: $[A(\boldsymbol{f})]_{ij} = \mathcal{F}_D\{a_{ij}(\cdot)\}(\boldsymbol{f})$
$A_j(\boldsymbol{f})$	Vecteur de dimension $I \times 1$, $j^{ème}$ colonne de la matrice $A(\boldsymbol{f})$
$R_j(\boldsymbol{f})$	Matrice de covariance spatiale de l'image de la source j à l'indice de fréquence $\boldsymbol{f}$ dans le cas diffus. De dimension $I \times I$

Notations mathématiques

$\cdot^*$	Conjugaison complexe
$\cdot^\top$	Transposition
$\cdot^H$	Conjugaison hermitienne (transposition et conjugaison complexe)
$\propto$	Proportionnel à
$\boldsymbol{a} \cdot \boldsymbol{b}$	Produit composante par composante de $\boldsymbol{a}$ et $\boldsymbol{b}$: $[\boldsymbol{a} \cdot \boldsymbol{b}]_k = \boldsymbol{a}_k \cdot \boldsymbol{b}_k$
$\frac{\boldsymbol{a}}{\boldsymbol{b}}$	Division composante par composante de $\boldsymbol{a}$ et $\boldsymbol{b}$: $\left[\frac{\boldsymbol{a}}{\boldsymbol{b}}\right]_k = \frac{\boldsymbol{a}_k}{\boldsymbol{b}_k}$
$\mathbb{C}^{\mathbb{T}}$	Ensemble des fonctions de $\mathbb{T}$ dans $\mathbb{C}$
$\delta_{tt'}$	Symbole de Kronecker, 1 si et seulement si $t = t'$, 0 sinon.
$\mathrm{diag}(v)$	Matrice diagonale dont les entrées sont les éléments du vecteur v. De dimension $L \times L$ si L est la longueur de v
I_L	Matrice identité de dimension $L \times L$
$\mathbb{N}_J$	Ensemble $[1, \ldots, J]$ des J premiers entiers naturels non nuls

Modèles de sources, séparation informée

D	En partie IV : distorsion moyenne maximale sur les signaux sources transmis
Δ_θ	Pas de quantification pour le modèle de sources θ
K	Nombre de composantes dans un modèle NTF
Q, W, H	Paramètres non-négatifs du modèle NTF
$\mathcal{P}(\cdot \mid \theta)$	Famille paramétrique de densités spectrales de puissance, indexée par leurs paramètres de sources θ
$\mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta)$	Valeur du modèle de densité spectrale de puissance de la source j au point $(\boldsymbol{f}, \boldsymbol{n})$
θ	Paramètres de sources dans une famille paramétrique $\mathcal{P}(\cdot \mid \theta)$ de densité spectrale de puissance
$\mathcal{M}_c$, M_c	Ensemble des sources mixées de manière convolutive. Leur nombre
$\mathcal{M}_d$, M_d	Ensemble des sources mixées de manière diffuse. Leur nombre
$\mathcal{S}_p$, S_p	Ensemble des sources disponibles à l'encodage sous forme ponctuelle. Leur nombre
$\mathcal{S}_d$, S_d	Ensemble des sources disponibles à l'encodage sous forme diffuse. Leur nombre
Θ, $\bar{\Theta}$	Pour un système de séparation informée, Θ désigne l'information annexe. Pour CISS, Θ sont les paramètres de modèle. $\bar{\Theta}$ est leur version quantifiée
$U_j(f)$	Filtre de formation de voie pour la récupération sous forme ponctuelle de la source j, si elle est mixée de manière diffuse
$\mathcal{Z}_s$, Z_s	Ensemble des sources à récupérer au décodeur sous forme ponctuelle. Leur nombre
$\mathcal{Z}_y$, Z_y	Ensemble des sources à récupérer au décodeur sous forme d'images. Leur nombre
$\mathcal{Z}_\varnothing$, $Z_\varnothing$	Ensemble des sources à ne pas récupérer au décodeur. Leur nombre

Abréviations

CI	Modèle de sources par compression d'images
CISS	*Coding-Based Informed Source Separation*
DSP	Densité Spectrale de Puissance
EC	Fonction de covariance Exponentielle Carrée
kbps	kilobits par seconde
NMF	*Nonnegative Matrix Factorization*
NTF	*Nonnegative Tensor Factorization*
PGLS	Processus Gaussien Localement Stationnaire
SAOC	*Spatial Audio Object Coding*
TFCT	Transformée de Fourier à Court Terme

Chapitre 1

Introduction

1.1 Préambule

Cette thèse porte sur le problème de la séparation de sources, qui vise à récupérer différents signaux appelés *sources,* à partir de l'observation de leurs *mélanges*. C'est un sujet qui a des applications dans de nombreuses disciplines du traitement du signal et qui a des liens très forts avec le vaste domaine des *problèmes inverses*. Dans le cas de la séparation de sources, l'opération à inverser est celle du *mixage* des sources.

De nombreuses techniques de séparation de sources ont été proposées et cette problématique a attiré l'attention d'une vaste communauté de chercheurs depuis le début des années 1980. Dans ce travail, je présente un formalisme particulier pour la séparation dans lequel les sources sont modélisées comme la réalisation de processus gaussiens. Ce cadre théorique permet de caractériser les signaux à séparer d'une manière souple et naturelle. Ce faisant, il rend possible la prise en compte de nombreuses connaissances *a priori* pour la séparation et se confond avec certaines méthodes de l'état de l'art dans plusieurs cas particuliers.

Parvenir à séparer des sources de leurs mélanges a des applications importantes dans le domaine du traitement du signal audio. Dans ce contexte, une séparation des différents instruments d'un morceau de musique rend possible certaines applications populaires comme le karaoké ou la respatialisation. Cependant, il est rare de parvenir aujourd'hui à obtenir une qualité suffisante de séparation pour ces applications. En conséquence, il a récemment été proposé d'améliorer la séparation en lui fournissant des informations supplémentaires en plus des seuls mélanges. Dans cette thèse, je montre comment ce cas de *séparation informée* peut être abordé naturellement avec le formalisme gaussien proposé. J'y explicite en outre les relations étroites entre la séparation de sources informée et le codage audio multicanal.

Ainsi, mon travail a porté à la fois sur une formalisation théorique pour la séparation de sources et sur son application au codage audio informé. Dans cette introduction, je présente d'abord le domaine de la séparation de sources en section 1.2, puis celui de la séparation informée en section 1.3. Pour chacun, je dresse un rapide état de l'art avant d'en présenter les enjeux qui m'ont intéressés au cours de mon travail et dont on trouvera un développement tout au long de ce texte. Cette introduction se conclut par la présentation du plan de l'exposé.

1.2 Séparation de sources

1.2.1 Motivations

La séparation de sources [28, 115, 38] consiste à récupérer plusieurs signaux à partir de l'observation de leurs mélanges. C'est un problème qui a des applications dans plusieurs domaines, tels que le traitement du signal audio, les télécommunications, les géostatistiques ou le traitement des signaux biomédicaux.

En audio, la séparation de sources est souvent introduite en évoquant l'effet *cocktail party* [26]. Lors d'une réception, de nombreuses conversations simultanées parviennent à mes oreilles. Pourtant, je suis capable si je le souhaite de ne porter mon attention que sur l'une d'entre elles. Ce

faisant, j'ai réduit l'influence des autres dans la compréhension de ce qui m'intéresse. En tout état de cause, je n'ai accès à l'environnement sonore que par le biais de mes deux oreilles. Il a donc bien fallu que je sois capable de *séparer* cognitivement certains sons de tous ceux que j'entends. De la même manière, je peux me concentrer sur un des instruments jouant dans une chanson, en l'isolant ainsi mentalement des autres. Cette capacité, si elle pouvait être imitée par une machine, permettrait la suppression à l'envi de n'importe quelle piste d'un enregistrement audio. Par exemple, je pourrais rajouter une lourde distorsion sur la rythmique d'un morceau, pour le rendre plus écoutable, ou bien en extraire la piste vocale pour l'utiliser dans un morceau de ma composition. Il est donc naturel qu'une large communauté de chercheurs en traitement du signal audio se soit penchée sur le problème [18, 17, 45, 50, 53, 72, 155].

En télécommunications, il est fréquent de recevoir un signal qui correspond à celui qui nous intéresse, mais qui a été contaminé par l'addition de signaux parasites plus ou moins complexes [28, 38]. Il s'agit alors de séparer le signal cible de ce *mélange*. La situation est similaire en géostatistiques [32], où la grandeur étudiée est souvent captée avec une incertitude sur la position ou la valeur de la mesure. Il s'agit alors de déduire la valeur recherchée à partir de ces mesures bruitées. Comme on le verra, le formalisme utilisé pour accomplir ces tâches est le même dans tous les cas : il s'agit de *séparer* le signal utile d'un bruit. Dans le cas des géostatistiques, un problème supplémentaire de *régression* s'ajoute : celui d'extrapoler une grandeur à des coordonnées différentes de celles des mesures.

Enfin, dans certaines disciplines telles qu'en traitement des signaux biologiques, la séparation de sources est couramment utilisée dans le but de décomposer une observation comme une somme de différentes contributions. Par exemple, lors du traitement d'électroencéphalogrammes, on cherche souvent à modéliser l'observation comme une somme de différentes contributions provenant de différentes sources localisées dans le cerveau, dans le but en particulier d'éliminer l'influence importante des clignements des yeux du sujet [112].

D'une manière générale, on verra que décomposer une observation comme une somme de fonctions élémentaires peut s'avérer utile à des fins d'analyse. Cependant, il n'est pas nécessaire que ces fonctions élémentaires correspondent à des signaux émis par de réelles entités indépendantes, comme c'est le cas des différents instruments de musique jouant dans un morceau. L'objectif de la décomposition peut tout simplement être d'expliquer au mieux une observation complexe comme la somme de plusieurs *variables latentes* plus simples. Cette approche déjà ancienne a donné lieu à des travaux précurseurs en statistiques sous le nom de *modèles additifs généralisés* [97, 98].

1.2.2 Notations

Je vais introduire quelques notations utiles pour présenter plus avant le problème de la séparation de sources. Le reste de la nomenclature sera introduit plus tard en partie I lors de la présentation du formalisme gaussien proposé et s'inspire fortement des notations habituelles dans le domaine [163, 155]. J'ai de plus choisi une notation qui ressemble beaucoup au style des langages scientifiques PYTHON [1] et MATLAB [2].

Dans toute la suite de cet exposé, $\mathbb{T}$ désignera le domaine de définition des sources. Par exemple, dans le cas de séries temporelles régulièrement échantillonnées, on aura $\mathbb{T} = \mathbb{Z}$ qui correspond aux différents instants d'échantillonnage. En général, je parlerai de *position* pour désigner un élément $t \in \mathbb{T}$, bien qu'il soit entendu que dans certains cas, t se comprend plutôt comme un instant. Le choix du terme de position se justifie par le traitement du problème de séparation dans le cas général de $\mathbb{T}$ quelconque en partie II. Dans tous les cas, la notation $T \in \mathbb{T}^L$ servira toujours à désigner un ensemble $T = [t_1, \ldots, t_L]$ de L points de $\mathbb{T}$.

On supposera toujours l'existence de $J \in \mathbb{N}$ sources. Tout au long de cet exposé, ce nombre de sources sera considéré connu et je ne me préoccuperai donc pas du problème délicat de son estimation, qui fait lui-même l'objet de nombreuses études [38].

L'ensemble de nos J sources est une fonction de [3] $\mathbb{T} \times \mathbb{N}_J$ à valeurs dans $\mathbb{C}$, que je représenterai par une lettre minuscule et la notation tilde, i.e. $\tilde{s}$. Ainsi, $\tilde{s}\,(t, j)$ désigne la valeur de la $j^{\text{ème}}$ forme

1. `www.python.org`
2. `www.mathworks.com`
3. $\mathbb{N}_J$ désigne l'ensemble $[1, \ldots, J]$ des J premiers entiers naturels non nuls.

d'onde $\tilde{s}(\cdot, j)$ à la position t. Pour un ensemble $T \in \mathbb{T}^L$ de L points de $\mathbb{T}$, $\tilde{\boldsymbol{s}}(T, \cdot)$ désignera la matrice de dimension $L \times J$ regroupant les J valeurs de $\tilde{s}$ en ces L points :

$$[\tilde{\boldsymbol{s}}(T, \cdot)]_{l,j} = \tilde{s}(t_l, j).$$

Un enregistrement stéréo $\tilde{x}$ composé de L échantillons ($\mathbb{T} = [1, \ldots, L]$) est un groupe de $I = 2$ formes d'ondes de longueur L. $\tilde{x}(\cdot, i)$ désigne le $i^{\text{ème}}$ canal (gauche ou droit), tandis que $\tilde{\boldsymbol{x}}(t, \cdot)$ représente le vecteur de dimension 1×2 contenant les valeurs à gauche et à droite de ce signal à l'instant t.

De la même manière qu'il y a un nombre connu J de sources, il y a un nombre connu I de mélanges, notés $\tilde{x}$. La principale caractéristique des problèmes de séparation de sources est que les mélanges sont générés à partir des sources. Cela justifie l'objectif de récupérer les sources à partir de l'observation des mélanges. Une étape essentielle dans la résolution de ce genre de problème est donc la formalisation du processus de mélange, aussi appelé *mixage*, qui permet de passer des sources aux mélanges, dans le but de *l'inverser*. Pour se fixer les idées, on peut imaginer le modèle le plus simple qui soit, qui consiste à sommer l'ensemble des sources pour obtenir un unique mélange. Dans ce cas, on a $I = 1$ et

$$\forall t \in \mathbb{T},\ \tilde{x}(t) = \sum_{j=1}^{J} \tilde{s}(t, j). \tag{1.2.1}$$

Cependant, ce modèle simple n'est satisfaisant que dans certains cas et plus de souplesse est nécessaire pour pouvoir modéliser un grand nombre de situations. On verra en partie II que la littérature est riche de nombreux modèles de mélange dont j'ai retenu les cas linéaires instantanés, convolutifs et diffus, particulièrement utiles en traitement du signal audio [49, 50, 51, 47]. Pour l'heure, je me contenterai d'en introduire la notion centrale d'*images des sources*.

Dans un enregistrement audio stéréo ($I = 2$) composé d'un trio ($J = 3$) chant, guitare et batterie, le son produit par la *source* "chant", est monophonique. Cependant, il s'entend sur chacun des 2 haut-parleurs : il produit une *image* stéréophonique, qui en est une version spatialisée. Le *mélange* est ainsi la somme des 3 images stéréophoniques des sources.

L'image d'une source $\tilde{s}(\cdot, j)$ a les mêmes dimensions que les mélanges et est donc une fonction de $\mathbb{T} \times \mathbb{N}_I$ dans $\mathbb{C}$ notée $\tilde{y}(\cdot, \cdot, j)$. Dans ces conditions, $\tilde{y}(\cdot, i, j)$ se comprend comme la *contribution* de la source j dans le mélange i. En effet, le mélange est modélisé comme la somme des images des sources :

$$\tilde{x} = \sum_{j=1}^{J} \tilde{y}(\cdot, \cdot, j). \tag{1.2.2}$$

Un modèle de mixage est alors entendu comme formalisant le lien entre une source et ses images. Comme on le verra en partie II, certains modèles comme le linéaire instantané ou le convolutif supposent un lien déterministe entre sources et images, d'autres comme le diffus supposent un lien probabiliste.

Dans tous les cas, l'objectif de la séparation de sources devient d'estimer les sources à partir de l'observation de leur mélange. Dans certains cas, extraire des images est suffisant comme lorsque l'objectif est de supprimer la voix d'un morceau de musique. Dans d'autres cas, les sources originales sont nécessaires, comme lorsqu'une respatialisation est envisagée.

1.2.3 Approches existantes

Le problème de la séparation de sources a été abordé de nombreuses manières différentes. Dans cette section, je vais présenter certaines des grandes idées qui ont dominé le domaine.

Tout d'abord, la plupart des méthodes existantes reposent sur l'hypothèse de mélange linéaire instantané. Cela signifie qu'on suppose que les I mélanges sont des combinaisons linéaires des sources :

$$\forall t,\ \tilde{x}(t, i) = \sum_{j=1}^{J} A_{ij} \tilde{s}(t, j). \tag{1.2.3}$$

Dans l'expression 1.2.3, chaque $A_{ij} \in \mathbb{C}$ donne le gain de la source j dans le mélange i. En introduisant la matrice de mélange A, de dimension $I \times J$, l'équation 1.2.3 peut alors s'écrire :

$$\forall t,\, \tilde{\boldsymbol{x}}\,(t,\cdot) = \tilde{\boldsymbol{s}}\,(t,\cdot)\,A^{\top}. \tag{1.2.4}$$

L'expression 1.2.4 est très classique en séparation de sources [4]. Plusieurs cas de figure s'offrent à nous.

S'il y a plus de mélanges que de sources ($I \geq J$), le problème est dit déterminé ($I = J$) ou sur-déterminé ($I > J$). Dans ce cas, il est équivalent à l'obtention d'une *matrice de séparation* W de dimension $I \times J$ telle que :

$$\forall t,\, \tilde{\boldsymbol{s}}\,(t,\cdot) = \tilde{\boldsymbol{x}}\,(t,\cdot)\,W. \tag{1.2.5}$$

Si A est connue, le problème peut être résolu, puisqu'il suffit alors pour obtenir W soit de l'inverser si c'est possible, soit d'utiliser une pseudo-inverse. D'une manière générale, des techniques de *formation de voie,* aussi appelées *filtrage spatial,* permettent de déterminer W et donc de récupérer les sources dans le cas sur-déterminé, à la condition que leurs distributions spatiales soient différentes. En effet, si deux sources sont sommées à l'identique sur $I \geq 2$ mélanges, aucun filtre uniquement spatial ne pourra les séparer.

Cependant, la plupart du temps, la matrice de mélange n'est pas connue et l'enjeu de la séparation sur-déterminée est de l'estimer. Pour faire face à cette incertitude et la prendre en compte dans le modèle, l'expression 1.2.4 est augmentée par l'ajout d'un terme d'incertitude, ou de bruit $\tilde{\boldsymbol{\epsilon}}\,(t,\cdot)$, distribué selon une loi connue [37, 28] :

$$\forall t,\, \tilde{\boldsymbol{x}}\,(t,\cdot) = \tilde{\boldsymbol{s}}\,(t,\cdot)\,A^{\top} + \tilde{\boldsymbol{\epsilon}}\,(t,\cdot)\,. \tag{1.2.6}$$

L'Analyse en Composantes Indépendantes (ACI, cf [37, 28, 109, 38]) permet de fournir une matrice de séparation au prix de certaines hypothèses. Tout d'abord, les sources sont considérées *indépendantes*. Cela signifie que sans considérer les mélanges, la connaissance d'une d'entre elles ne donne d'information sur aucune des autres. Cette hypothèse, quoique discutable dans certains cas [5] est la plupart du temps très raisonnable et d'ailleurs commune à la presque totalité des techniques existantes pour la séparation de sources.

Le terme de bruit $\tilde{\epsilon}$ dans l'équation 1.2.6 peut être compris soit comme un bruit réellement ajouté aux mélanges lors du mixage, soit comme modélisant notre *incertitude* sur le mixage. Dans les deux cas, la résolution du problème mène aux mêmes équations.

La deuxième hypothèse forte faite par l'ACI est que le signal correspondant à chaque source à extraire est une séquence de réalisations indépendantes d'une même variable aléatoire. Enfin, si on suppose qu'au plus une seule des sources a une loi gaussienne, alors on démontre [37] qu'il est possible de séparer les sources à partir de leur mélange 1.2.6 à un coefficient d'amplitude et à une permutation près. Cette séparation s'effectue en pratique par la maximisation de la non Gaussianité des signaux obtenus ou bien par la minimisation de leur information mutuelle.

Un des inconvénients majeurs de l'ACI est qu'elle se transpose difficilement au cas sous-déterminé, c'est-à-dire au cas où il y a moins de mélanges disponibles que de sources ($I < J$). Dans ce cas en effet, le problème devient assez difficile. Pour reprendre l'exemple 1.2.4 du mixage linéaire instantané, connaître la matrice de mélange A ne suffit plus. Il devient nécessaire de mettre au point d'autres approches et la séparation sous-déterminée a d'abord été sentie comme un problème de *contrôle*.

Il est en effet possible de voir l'équation 1.2.6 comme caractérisant un processus d'observation des sources, qui sont elles-mêmes un processus latent, ou un *état caché*. Dans ce cadre, l'évolution des sources et des mélanges peut être modélisée par un modèle à état, classique de la discipline du contrôle optimal [119]. Les sources peuvent alors être estimées en utilisant des algorithmes adaptatifs dits de Kalman [116] pour résoudre le problème. De nombreuses études se sont focalisées

4. Nous considèrerons l'autre cas classique des mélanges *convolutifs* au chapitre 6. Ils peuvent sous certaines conditions se ramener aux mélanges linéaires instantanés dans le domaine fréquentiel.

5. Des sources audio, jouant en rythme et en harmonie, peuvent parfois difficilement être considérées comme indépendantes.

sur ce point de vue [35, 154]. Cependant, leur principale faiblesse réside dans le modèle dynamique qu'elles supposent pour les sources : il est souvent difficile de modéliser l'évolution complexe des sources au cours du temps comme une simple multiplication matricielle suivie de l'ajout d'un bruit. Pour cette raison, des alternatives non linéaires ont été proposées [204] mais impliquent encore des algorithmes d'une complexité prohibitive pour des signaux de grande taille, tels que les signaux audio. La modélisation *dynamique* de signaux sonores attire cependant de nouvelles études depuis peu [11, 14] et reste un champ prometteur de recherches.

DARMOIS a démontré en 1953 qu'il est impossible de séparer des mélanges linéaires de sources indépendantes et identiquement distribuées (i.i.d.) qui sont toutes gaussiennes [41, 37, 39]. Alors que l'ACI se concentre sur le cas de sources i.i.d. non gaussiennes, le formalisme que j'ai étudié dans mon travail suppose des signaux sources gaussiens mais non i.i.d.

Une approche importante dans le domaine de la séparation sous-déterminée depuis quelques années repose sur un modèle gaussien [18, 17, 214] des sources. Je reviendrai plus en détail sur ce modèle en partie I. Pour l'heure, je peux déjà en expliciter les idées principales. Ce modèle a été tout particulièrement étudié dans le cas de la séparation de sources audio ($\mathbb{T} = \mathbb{Z}$), et fut d'abord considéré dans le cas où il n'y a qu'un seul mélange ($I = 1$). Le mélange est supposé être la simple somme des sources comme dans l'expression 1.2.1. Les signaux sont tout d'abord transformés pour en obtenir une représentation Temps-Fréquence (TF) telle que la Transformée de Fourier à Court Terme (TFCT, [5, 6, 40, 92]). Dans ce domaine, plusieurs hypothèses sont faites.

Tout d'abord, comme dans l'ACI, les sources sont supposées indépendantes. Ensuite, tous les coefficients de la TFCT d'une source donnée sont supposés indépendants et distribués selon des lois gaussiennes complexes centrées d'une certaine variance, assimilable à la Densité Spectrale de Puissance (DSP) de la source en ces instants et ces fréquences. Si on suppose connues les DSP des sources, on montre que la séparation peut se faire très simplement par l'application trame par trame d'un filtrage de WIENER [217], qui est optimal au sens des moindres carrés. Cette technique est souvent appelée filtrage de WIENER *généralisé*.

L'enjeu essentiel des techniques basées sur ce modèle gaussien devient alors d'estimer les DSP des sources à partir de la seule observation du mélange. Un large effort de recherche de la communauté s'est opéré sur ce point précis. Dans le cadre gaussien, les DSP des sources s'ajoutent pour obtenir celle du mélange, estimée par son *spectrogramme*[6]. Le problème de la séparation de formes d'ondes est donc rapidement devenu celui de la séparation de DSP. Plusieurs constats et techniques ont guidé ces recherches.

Tout d'abord, les DSP sont des grandeurs nécessairement positives. Ensuite, il est courant pour de nombreuses applications, en particulier le traitement du signal musical, qu'elles présentent de très nombreuses redondances. En effet, les mêmes notes sont susceptibles de se reproduire à de nombreux endroits dans le morceau. Dans ces conditions, on comprend que des techniques de réduction de dimension aient été appliquées, qui portent le nom de factorisation en matrices non-négatives (NMF, en anglais) [128, 34, 194, 192, 188, 75] et qui permettent de décomposer le spectrogramme du mélange comme la superposition d'éléments simples. Cette approche donne parfois de très bons résultats, mais il s'est avéré rapidement que son principal problème réside dans l'arbitraire des composantes produites. En effet, en l'absence de contraintes supplémentaires, la NMF va chercher à expliquer au mieux le mélange comme une somme de composantes, mais ces composantes ne correspondent pas nécessairement à des sources [19]. Ainsi, plusieurs pistes de recherche ont émergé qui cherchent toutes à forcer la NMF à produire des composantes plus satisfaisantes. Cela s'est surtout fait en pénalisant toute décomposition qui ne répondait pas à certains critères.

Tout d'abord, il a été tenté d'inclure dans la NMF certaines contraintes de régularité, permettant de garantir qu'aucune des composantes ne soit activée puis désactivée de manière trop abrupte [211, 182, 215, 19, 181, 46]. Ensuite, certains modèles ont cherché à décomposer le spectrogramme du mélange comme une somme de motifs d'une certaine durée, dans le but de ne plus extraire seule-

6. Le spectrogramme est défini comme le module au carré de la TFCT. Je reviendrai sur ce point en section 2.5 page 35.

ment des spectres instantanés redondants, mais plutôt des blocs entiers [192, 183]. Ces modèles reposent sur une notion d'invariance par translation : chaque bloc est supposé se reproduire à l'identique plusieurs fois dans le mélange. Par ailleurs, il a été tenté par Durrieu *et al.* de décomposer les spectrogrammes en utilisant des modèles plus complexes permettant de forcer certaines sources à présenter une ligne mélodique. Ces approches se concentrent sur la séparation de la voix dans les enregistrements musicaux et permettent d'obtenir de bonnes performances [56, 55, 53, 163, 57].

Une autre piste de recherche a consisté à ne plus utiliser la TFCT comme transformée pour analyser les signaux, mais plutôt une autre transformée dénommée la transformée à Q constant (CQT en anglais). L'avantage de cette représentation est qu'elle présente une échelle logarithmique selon l'axe des fréquences. Ainsi, un changement de hauteur se traduit non plus comme une homothétie des spectres, mais plutôt comme une translation [101]. Des modèles invariants par translation ont pu être mis au point pour exploiter cette propriété dans le cas de la séparation et l'analyse de musique [74, 84, 193].

Ces modèles pour la séparation de sources avec un seul mélange ont été étendus dans de nombreuses directions pour traiter le cas de mélanges multicanaux.

Une partie de la discussion a donc porté sur la nature du processus de mélange considéré. Pour peu que les sources aient été distribuées dans des directions spatiales suffisamment séparées, un filtrage spatial par formation de voie [27] permet de les extraire correctement [91, 221] après estimation de ces directions. Dans le cas contraire, il devient nécessaire de modéliser la structure des sources de manière à résoudre les ambiguïtés.

En audio, un ingénieur du son cherche souvent à bien séparer les différents instruments dans l'espace stéréophonique. Ce faisant, il provoque une bonne séparation spatiale des sources, qui facilite la séparation. Si deux instruments se retrouvent au même azimut (chant et basse au centre par exemple), il devient nécessaire de pouvoir les séparer par d'autres considérations.

Des études ont alors procédé à l'extension du formalisme NMF au cas des mélanges multicanaux, en considérant d'abord le cas convolutif [155, 71, 54], puis le cas diffus [50, 163]. Il devient apparent aujourd'hui que des modèles paramétriques tels que la NMF ne suffisent pas toujours à modéliser la complexité des signaux. C'est pourquoi certaines études cherchent à les modéliser de manière plus souples. En particulier, l'idée de modéliser les DSP d'une source comme la réalisation d'un champ aléatoire a été suggérée [45, 149]. Cette direction de recherche me parait très prometteuse, mais n'en est encore qu'à ses commencements.

1.2.4 Enjeux

En cherchant dans la littérature si d'autres communautés s'intéressaient à des problèmes similaires à celui de la séparation de sources, il m'est apparu que de nombreuses techniques utilisées en géostatistiques [144, 111, 44, 32] procèdent à un type de séparation de sources spatiales appelée Krigeage. Plus particulièrement, ces méthodes séparent le signal utile d'un bruit, et effectuent également des analyses de signaux multicanaux, tels que différents types de mesures effectués dans le sol. De la même manière, certaines études en apprentissage automatique permettent de décomposer des observations en sommes de composantes latentes [178].

Dans l'exemple p.118 de [178], les auteurs utilisent les processus gaussiens pour décomposer une mesure atmosphérique en ses composantes saisonnières et moyenne. Ce faisant, ils font sans le dire de la séparation de sources. C'est ainsi que m'est venue l'idée d'utiliser les processus gaussiens pour la séparation.

C'est dans ce contexte qu'un de mes efforts a été de définir un cadre général pour la séparation de sources qui permette d'unifier un grand nombre d'approches existantes. Ceci a été rendu possible par la formulation de la séparation de sources comme un problème de régression de processus gaussiens [178], que je présenterai en détails en partie II. L'établissement de ce formalisme constitue une part importante du travail théorique effectué sur cette thèse, et a pu être utilisé largement dans le domaine applicatif que j'ai étudié, celui de la séparation informée que je vais maintenant présenter.

1.3 Séparation informée

1.3.1 Motivations

La séparation de sources en traitement du signal musical ouvre des perspectives sur de nombreuses applications. En effet, elle se traduit dans ce domaine par la capacité à *démixer* la musique, c'est-à-dire à pouvoir récupérer les pistes individuelles des instruments constituant l'enregistrement.

Si une telle opération est possible, elle permet la suppression ou l'isolation de n'importe quel instrument, une application appelée *karaoké généralisé* [10, 153]. La possibilité d'obtenir isolément l'enregistrement de chaque instrument permet en outre d'envisager des applications de remixage enthousiasmantes pour l'amateur de musique et des collaborations artistiques anachroniques.

Les avancées dans le domaine de la séparation de sources vont inévitablement poser le problème de sa compatibilité avec la notion de droits d'auteur. La réflexion a déjà beaucoup avancé grâce au développement des licences Creative Commons et à leur intégration progressive dans les institutions.

Malheureusement, si les résultats des systèmes de séparation de sources présentés plus haut en section 1.2 sont très encourageants, ils ne sont souvent pas d'une qualité suffisante pour de telles applications. Cela vient du fait que ces systèmes ne disposent que de très peu d'informations sur les sources à séparer. Au plus, ils font certaines hypothèses sur leur structure ou leurs propriétés statistiques. Très ambitieux, ce cadre de travail s'appelle l'approche *aveugle*. Certaines études récentes tendent à montrer [156, 57, 77] que les performances de la séparation augmentent significativement à partir du moment où un utilisateur peut fournir un guidage du système [7]. Ce guidage peut permettre au système de choisir entre plusieurs décompositions également probables, ou bien de forcer l'algorithme à respecter une structure connue. On parle dans ce cas de séparation de sources *supervisée*.

D'une manière générale, il a vite semblé judicieux d'aider la séparation de sources par la prise en compte de toute *information annexe* disponible sur les sources, pour en rendre les résultats meilleurs. Dans ce cas, la séparation n'est plus aveugle, elle devient *informée*. Par exemple, certaines études pionnières [18, 162] ont cherché à apprendre en amont de la séparation certains modèles de sources grâce à une base de données. Ces modèles sont alors utilisés pour la séparation. D'autres approches ont consisté à améliorer la séparation d'un morceau de musique par la connaissance d'une partition [80, 95, 101, 63, 191]. Le domaine de la séparation informée permet de tirer parti de nombreuses informations relatives aux sources pour faciliter la séparation.

Même avec l'aide d'un utilisateur ou de modèles spécifiques, les techniques évoquées plus haut ne permettent pas toujours d'obtenir des résultats d'une qualité suffisante. Cela est dû au fait que les sources à séparer peuvent ne pas correspondre exactement au modèle ou bien que le critère utilisé lors de l'apprentissage ne conduise pas à l'estimation des bons paramètres.

Une idée originale de Parvaix et Girin proposée dans [168, 166] est d'introduire un cas *fortement informé* où l'information annexe considérée est calculée sur les sources à extraire elles-mêmes. Dans ce cas, cette information peut avoir été calculée spécifiquement dans le but de produire de bons résultats de séparation.

Cette idée peut paraître très surprenante au premier abord, puisque la séparation de sources vise précisément à la récupération des signaux mélangés. Si on les suppose connus pour le calcul de l'information annexe, on peut se demander quel est le cas d'application de la technique.

Dans un contexte musical, le mélange est souvent fait en studio par un ingénieur du son qui dispose des pistes séparées. Le calcul d'une information annexe à partir des sources et du mélange peut se faire à ce moment-là, et être utilisé plus tard par un utilisateur qui n'a accès qu'au mélange.

Il est vite apparu que ce cas particulier de séparation informée s'apparente fortement à un

7. J'ai remarqué lors de mes années passées à Audionamix qu'un bon guidage de la séparation peut fournir des résultats d'une excellente qualité.

problème de codage, résumé en figure 1.1. Le traitement se considère en deux temps. Tout d'abord, une étape d'*encodage* bénéficie de la connaissance jointe des sources et des mélanges. Elle produit une information annexe, envoyée au *décodeur*. Lors du décodage, on connaît à la fois le mélange et l'information annexe, et le système produit un lot de *sources estimées*.

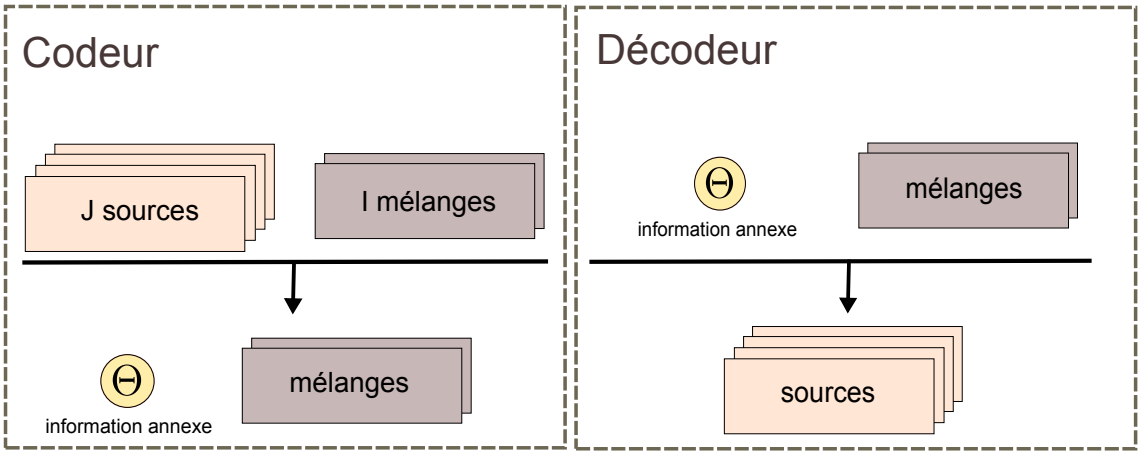

FIGURE 1.1: Schéma synthétique de la séparation informée. Lors d'une étape d'encodage, les sources et les mélanges sont disponibles. Au décodage, seuls les mélanges et l'information annexe sont disponibles.

La séparation informée permet de garantir une bonne séparation des sources, au prix de l'envoi en plus des mélanges d'une information annexe, ce qui nécessite un certain débit.

Pour distinguer ce cas fortement informé des autres approches évoquées plus haut où l'information annexe n'est pas calculée en utilisant les sources mais seulement les mélanges ou l'intervention d'un utilisateur, j'ai introduit dans cet exposé une distinction entre séparation aveugle, semi-informée et informée qui est résumée dans le tableau 1.1. Alors que la séparation aveugle, que je ne traiterai pas dans cette étude, se caractérise par la faiblesse des hypothèses qu'elle fait sur les signaux mélangés, la séparation semi-informée utilise une connaissance annexe pour restreindre le champ d'étude des signaux à séparer, de manière à améliorer les performances de la séparation. Les paramètres des modèles y sont appris sur les seuls mélanges, contrairement au cas de la séparation informée où les sources elles-mêmes peuvent être utilisées pour produire une information annexe efficace, comme représenté en figure 1.1.

L'avantage crucial de l'approche informée est qu'elle permet de produire des sources séparées qui sont systématiquement d'une très bonne qualité. Bien entendu, ce gain en performance se fait au prix de la transmission d'une information annexe en plus des mélanges. Il est donc important que cette information soit de petite taille de manière à ce que sa transmission corresponde à un coût supplémentaire en débit faible par rapport à celui utilisé pour transmettre les mélanges. En effet, sans cette propriété, autant transmettre directement les sources en les encodant avec des techniques de compression classiques. Or, c'est bien le cas : les techniques que j'ai mises au point pendant mon doctorat permettent de récupérer des sources d'une excellente qualité avec un débit de l'ordre de 2kbps [8] par source, qui est à mettre en perspective par rapport aux 128kbps souvent requis pour la transmission du mélange.

L'obtention de ces sources rend possibles toutes les applications de karaoké généralisé ou de remixage évoquées plus haut.

8. kilobits par seconde

Séparation	aveugle (ACI)	semi-informée	informée
hypothèses sur les sources	– indépendantes – identiquement distribuées et non gaussiennes	– indépendantes – nature des signaux – familles paramétriques	idem qu'en semi-informé
hypothèses sur le mixage	– déterminé ou surdéterminé ($I \geq J$) – type de mixage connu	– type de mixage connu	– type de mixage connu
Apprentissage des paramètres	sur les mélanges	sur les mélanges	sur les mélanges **et les sources**

TABLE 1.1: Nomenclature utilisée dans ce texte pour les différents types de séparation de sources.

1.3.2 Notations

Le problème de la séparation informée a été uniquement étudié dans le cas des signaux audio régulièrement échantillonnés ($\mathbb{T} = \mathbb{Z}$). Pour présenter les méthodes de l'état de l'art, il est nécessaire d'introduire les représentations Temps Fréquence (TF) des signaux. Plus de détail sur ce point sera donné en sections 3.1 page 41 et 3.3 page 52. Pour l'instant, je me contenterai de présenter brièvement la Transformée Fréquentielle à Court Terme (TFCT) $s(\cdot,\cdot)$ d'une forme d'onde $\tilde{s}(\cdot)$. Elle correspond au spectre de l'onde estimé localement autour de N positions d'analyse. Ainsi, $s(f,n)$ désigne le spectre de $\tilde{s}$ autour d'une fréquence ω_f et de l'instant d'analyse t_n. Souvent, la transformée utilisée est la transformée de Fourier mais elle peut aussi être la transformée en cosinus discrète, ou bien n'importe quel banc de filtres.

Dans le cas des signaux audio, la TFCT d'un signal $\tilde{s}$ de dimension $L \times 1$ se calcule en découpant le signal en N trames de longueur identique L_0 et régulièrement espacées, avec entre elles un certain *recouvrement*. Le spectre de chacune de ces trames est alors calculé pour obtenir la TFCT s de $\tilde{s}$, de dimension $F \times N$, où F est le nombre d'indices fréquentiels non redondants. $s(f,n)$ est alors le spectre de $\tilde{s}$ à l'index de fréquence f et autour de l'instant t_n.

Sous certaines conditions, la forme d'onde $\tilde{s}$ peut être récupérée à partir de sa TFCT s. Cette étape implique l'utilisation de techniques d'addition-recouvrement, classiques en traitement du signal audio [5, 6, 40, 92] et qu'on reverra plus tard en section 3.2.3 page 48.

Dans le cas de la TFCT de plusieurs sources, $\boldsymbol{s}(\cdot,\cdot,j)$ désignera la TFCT de la source $\tilde{s}(\cdot,j)$ et $\boldsymbol{s}(f,n,\cdot)$ est le vecteur de dimension $J \times 1$ qui regroupe les TFCT de chacune des sources au point TF (f,n) :

$$\boldsymbol{s}(f,n,\cdot) = [s(f,n,1),\ldots,s(f,n,J)]^\top .$$

Dans la suite de cet exposé, la notation v_s désigne le module au carré de la TFCT s, aussi appelé spectrogramme. Ainsi,

$$[v_s(\cdot,\cdot,j)]_{f,n} = |s(f,n,j)|^2$$

est le spectrogramme de la source j et

$$v_s(f,n,\cdot) = [v_s(f,n,1),\ldots,v_s(f,n,J)]^\top$$

l'ensemble des spectrogrammes des sources pour le point TF (f,n). On définit les spectrogrammes des mélanges $v_x(\cdot,\cdot,i)$ et $v_x(f,n,\cdot)$ de la même manière.

1.3.3 Parcimonie et inversion locale

Les premières méthodes proposées pour la séparation de sources informée par PARVAIX [168, 166, 167] reposent sur deux hypothèses principales.

La première hypothèse est que le mélange est linéaire instantané comme dans l'expression 1.2.4. Puisque la TFCT est une transformation linéaire, ce modèle de mélange a son équivalent dans le domaine fréquentiel :

$$\forall (f,n)\,,\ \boldsymbol{x}(f,n,\cdot) = A\boldsymbol{s}(f,n,\cdot)\,. \tag{1.3.1}$$

La deuxième hypothèse de ces méthodes est celle de *parcimonie* faite sur les signaux. Plus précisément, elles supposent que parmi l'ensemble des coefficients de la TFCT $\boldsymbol{s}(\cdot,\cdot,j)$ de chacune des sources $\tilde{\boldsymbol{s}}(\cdot,j)$, il n'y en a que peu qui ont une énergie significative. Cette hypothèse est souvent justifiée dans le cas des signaux audio, où l'énergie est concentrée dans des lignes harmoniques ou bien dans des impulsions rythmiques. Ceci a déjà été montré et exploité par plusieurs auteurs dans le domaine de la séparation [22, 91, 175] et a fait l'objet d'intenses recherches pour la compression audio [33, 147, 146]. Dans la figure 1.2, j'ai affiché le log-spectrogramme $\log \boldsymbol{v}(\cdot,\cdot,j)$ de deux sources, ainsi que sa distribution. Il est remarquable que seulement une faible proportion des points TF présente une énergie significative. Pour les points TF où l'énergie de la source est importante, on dit qu'elle y est *active*. Dans le cas contraire, on la dit *inactive*.

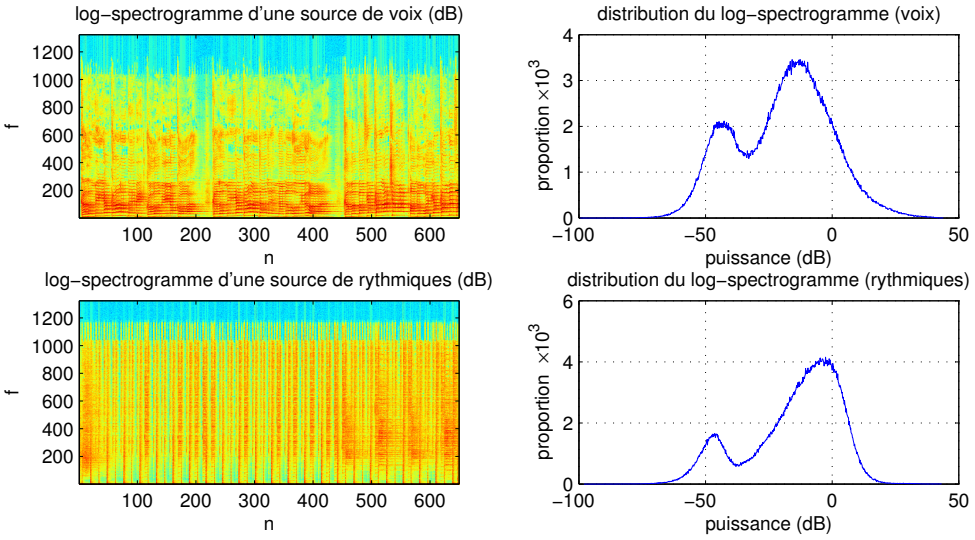

FIGURE 1.2: Affichage (gauche) et distribution (droite) des log-spectrogrammes d'une source de voix chantée (haut) et de sons percussifs (bas). Comme on peut le constater, la proportion des points qui ont une énergie importante (supérieure à 0dB par exemple) est faible, justifiant l'hypothèse courante de parcimonie.

Dans ce cadre, la technique proposée par PARVAIX fait l'hypothèse que dans chacun des points TF (f,n), il y a *au plus* I sources actives. Cette hypothèse de parcimonie faible se distingue de l'hypothèse habituelle de parcimonie forte [221] qui suppose que dans chaque point TF, seulement *une* des sources est active. Dans le cas d'un mélange audio stéréophonique ($I = 2$), cela revient à supposer que dans chaque point TF, au maximum 2 sources ont une énergie importante [9]. Les indices de ces sources actives sont rassemblés dans un ensemble *d'indices* $\mathcal{I}(f,n) \subset \mathbb{N}_J$. Les sources *inactives* en sont le complémentaire $\overline{\mathcal{I}}(f,n)$. Le nombre d'éléments de $\mathcal{I}(f,n)$, noté $\#\mathcal{I}(f,n)$ est inférieur à I :

$$\forall (f,n)\,,\ \#\mathcal{I}(f,n) \leq I \tag{1.3.2}$$

Supposons à présent qu'un codeur ait identifié $\mathcal{I}(f,n)$ et l'ait envoyé, en plus de la matrice de mélange A, au décodeur. Dans la mesure où il n'y a que peu de possibilités pour le choix de $\mathcal{I}(f,n)$, l'information annexe $\{\mathcal{I}, A\}$ à transmettre peut être codée très efficacement.

9. Ces hypothèses de parcimonie *forte* et *faible* se rejoignent dans le cas d'un seul mélange [168].

Considérons à présent le décodage, ou l'étape de séparation. Pour un point TF donné, les sources inactives $\overline{\mathcal{I}}(f,n)$ peuvent être estimées comme étant nulles :

$$\forall (f,n)\,,\ \forall j \in \overline{\mathcal{I}}(f,n)\,,\hat{s}(f,n,j) = 0. \tag{1.3.3}$$

En ce qui concerne les sources actives, l'hypothèse de parcimonie faible nous mène à un système à résoudre beaucoup plus simple. En effet, les sources inactives, d'une énergie très faible, peuvent être retirées du système d'équations 1.3.1, qui ne fait plus intervenir que les sources actives :

$$\forall (f,n)\,,\ \forall i,\ x(f,n,i) = \sum_{j \in \mathcal{I}(f,n)} A_{ij} s(f,n,j)\,. \tag{1.3.4}$$

Dans la mesure où on a supposé que $\#\mathcal{I}(f,n) \leq I$, le système d'équations 1.3.4 est surdéterminé. Puisque le décodeur connaît A, ce système peut être inversé rapidement et on obtient ainsi une estimée pour toutes les sources au point TF (f,n). Ces considérations justifient le nom de séparation par *inversion locale* donné à la méthode.

Cette technique par inversion locale a été étendue dans plusieurs directions. Tout d'abord, les indices des sources actives ont été tatouées dans les mélanges, de manière à ce que l'information annexe composée des indices $\mathcal{I}(f,n)$ n'ait pas à être transmise indépendamment des mélanges. Cela a été rendu possible par la mise au point récente par Pinel *et al.* d'une technique de tatouage audio permettant d'embarquer de manière inaudible suffisamment de données dans un signal [173, 172]. Par ailleurs, il est fréquent que les indices des sources actives soient les mêmes d'un point TF à ceux qui lui sont adjacents. Cette propriété peut être exploitée pour réduire le débit nécessaire à la transmission des $\mathcal{I}(f,n)$. Ensuite, il est intéressant de constater que dans le cas des mélanges convolutifs et sous certaines conditions (voir la section 6.1), l'équation 1.3.1 reste valide, à la seule différence que la matrice de mélange dépend de l'indice fréquentiel f considéré. La technique de séparation locale peut donc être appliquée dans ce cas [136].

1.3.4 Codage spatial

Le contexte dans lequel je me suis placé au début de mon travail de thèse était résolument axé sur la séparation de sources. Telle que je la concevais en commençant à travailler, la problématique que je devais envisager concernait avant tout la séparation et les études émanant de cette communauté. Cependant, il est vite apparu que le sujet de la séparation informée est en réalité un problème assez proche de celui du codage multicanal [10][31, 104, 61, 60, 66, 15].

En effet, la structure d'une technique de séparation informée, telle que représentée en figure 1.1, s'apparente exactement à celle d'un codeur-décodeur multicanal classique. Lors d'une première phase de codage, les sources sont analysées de manière à générer un flux de données. Ce flux de données est ensuite traité par un décodeur de manière à restituer fidèlement les sources. La correspondance avec l'état de l'art dans le domaine du codage audio multicanal devient frappante si on considère un instant la technique du codage spatial (Spatial Audio Coding, SAC [25, 103, 24, 31, 105]) telle que formalisée par le groupe de normalisation MPEG.

Le codage spatial SAC cherche à transmettre un signal audio $\tilde{s}$ de J canaux de manière compressée. Pour ce faire, sa principale idée est de réduire ce signal en un signal $\tilde{x}$ disposant d'un nombre réduit I ($I \leq J$) de canaux pour la transmission, puis de récupérer le signal initial $\tilde{s}$ lors du décodage en utilisant une méthode de *respatialisation*, paramétrée par une information annexe embarquée dans le flux de métadonnées [11] de $\tilde{x}$. Cette approche a de nombreux avantages. Tout d'abord, elle permet une réduction du débit nécessaire à la transmission de $\tilde{s}$. En effet, il est moins coûteux de transmettre $I < J$ signaux. Ensuite, elle permet l'utilisation de techniques déjà existantes pour la compression audio du signal $\tilde{x}$, telles que MPEG1-LayerIII (MP3 [146]) ou MPEG2-LayerIII (AAC [147]). Ce faisant, elle provoque une rétro-compatibilité de SAC sur

10. Je remercie Laurent Girin, Sylvain Marchand ainsi que Shuhua Zang de m'avoir mis sur la piste des codeurs spatiaux, en cours de normalisation.

11. La plupart des standards de transmission prévoient un flux de *métadonnées*, qui permet d'embarquer dans le signal transmis des données supplémentaires, dont la nature est à la discrétion de l'utilisateur. Dans le cas de la musique, ces métadonnées contiennent habituellement des informations sur le morceau de musique.

les encodeurs MPEG classiques, puisque l'utilisateur ne disposant pas d'un appareil équipé pour récupérer $\tilde{s}$ à partir de $\tilde{x}$ peut au moins lire $\tilde{x}$. Cette propriété est très appréciable dans un contexte de normalisation de standards de transmission.

Bien entendu, ces techniques nécessitent la définition d'un moyen de récupérer le signal initial $\tilde{s}$ à partir de $\tilde{x}$. Le standard SAC utilise à cette fin une décomposition par un banc de filtres suivie d'un traitement de respatialisation basé sur des critères spatiaux. Ces critères incluent la différence de phase et d'intensité entre les différentes pistes de $\tilde{s}$, qui permettent de mettre au point des filtres de respatialisation efficaces. Ces données constituent l'information annexe considérée par SAC. Plus de détail sur ces systèmes pourront être trouvés dans [25, 24, 31].

Plus récemment, le codage spatial SAC a été étendu pour devenir le codage d'objets spatiaux (Spatial Audio Object Coding, SAOC [104, 61, 60, 66, 15]). L'idée principale de SAOC demeure la même que SAC, à savoir que les différentes ondes de $\tilde{s}$ sont mélangées pour obtenir le signal transmis $\tilde{x}$. La différence entre SAC et SAOC demeure principalement conceptuelle : alors que SAC encode une *spatialisation*, SAOC cherche à encoder des *objets* sonores. En conséquence, SAOC nécessite une technique de séparation très efficace des objets pour éviter leurs interférences, alors qu'un objectif de respatialisation était moins exigeant. C'est cependant la même approche globale pour la séparation qui a été choisie lors du passage de SAC à SAOC, basée sur l'utilisation de critères de différences de phase et d'amplitude entre canaux [104].

Comme on le voit, les problématiques de SAOC, issues de la communauté du codage audio, et celles de la séparation de sources informée sont très similaires, pour ne pas dire identiques : dans les deux cas, le problème est la transmission des sources au décodeur, en passant par la transmission de signaux de mélange et d'une information annexe générée par un encodeur. Il est intéressant de constater que le même problème a donc été abordé indépendamment par deux communautés différentes. Un des objectifs de mes travaux est de montrer qu'on peut en effet voir le problème de la séparation informée comme un problème de codage, mais que des techniques à base de séparation de sources sont extrêmement efficaces dans ce but.

SAOC aborde le problème de la séparation informée avec des techniques de codage. Je l'ai initialement abordé avec des techniques de séparation. Un des objectifs de ce texte est de montrer que ces deux points de vue peuvent se confondre.

1.3.5 Enjeux

Au début de mon travail de thèse, le problème de la séparation de sources informée tel qu'introduit par Parvaix et Girin [168, 166, 169] était résolument nouveau. En parallèle, des efforts de normalisation se faisaient déjà avec SAOC [104, 61, 60] dans le sens du codage d'objets audio dans des mélanges. Le contexte était donc assez riche et les perspectives qui m'étaient ouvertes étaient nombreuses.

Tout d'abord, les méthodes existantes faisaient des hypothèses assez fortes sur les signaux sources. Si elle est raisonnable et permet d'obtenir de très bonnes performances, l'hypothèse faible de parcimonie vue en section 1.3.3 et la technique associée d'inversion locale ont l'inconvénient de provoquer la mise à 0 de nombreux coefficients TF des sources estimées, ce qui conduit à la présence lors de l'écoute d'un *bruit musical* caractéristique [86]. La suppression de ce bruit passe par l'introduction de modèles de séparation qui ne font plus d'hypothèses strictes de parcimonie, tel que le formalisme gaussien que je présenterai en partie III.

Une autre perspective ouverte au début de mon travail était l'extension de la séparation informée au cas où les mélanges ne sont plus nécessairement linéaires instantanés comme dans le modèle 1.2.4. J'ai déjà souligné que la technique par inversion locale a été étendue depuis au cas des mélanges convolutifs [134]. Cependant, de telles extensions n'étaient pas encore d'actualité au début de mon travail. SAOC, quant à lui, demeure restreint au cas des mélanges linéaires instantanés [61, 104]. Une perspective ouverte était ainsi d'étendre la séparation informée à des cas de mélanges plus complexes. On trouvera en partie II le détail des différents modèles que j'ai considérés pour le mixage, qui incluent le cas linéaire instantané, le cas convolutif et le cas diffus.

Une des limitations des approches évoquées plus haut est qu'elles contrôlent elles-même l'étape de mixage. En effet, à la fois dans SAOC et dans l'approche parcimonieuse de Parvaix, le mélange

est généré à partir des sources grâce à une matrice de mélange ou à des filtres connus. Bien qu'il soit clair que des mélanges très satisfaisants puissent être générés de cette manière, cette contrainte demeure un frein fort à l'utilisation massive de ces techniques dans l'industrie musicale. En effet, l'étape de mixage constitue un art maîtrisé par les ingénieurs du son et les mélanges obtenus par des professionnels dépassent de très loin en qualité ceux obtenus par l'amateur, et à plus forte raison ceux obtenus par un simple mélange linéaire instantané ou convolutif. D'une manière générale, il serait désirable de pouvoir formaliser le problème de la séparation informée dans le cas où les mélanges des sources sont *imposé*s. Cela permettrait de pouvoir procéder à la séparation des morceaux produits par les ingénieurs du son et d'ainsi appliquer la séparation informée aux enregistrements du commerce. De telles applications ne sont pas prises en charge par les méthodes existantes telles que SAOC (cf section 1.3.4) ou l'inversion locale (cf section 1.3.3). Nous verrons que le formalisme que j'introduis permet de considérer ce cas de figure assez naturellement.

En fonction de l'application, on peut vouloir séparer les sources ou les images. Une respatialisation de qualité nécessite des sources tandis qu'une version karaoké ne nécessite que de supprimer des images du mélange.

Une autre question ouverte au début de mon travail, liée au point précédent, était la formalisation plus claire de ce qu'est une source audio dans notre contexte. Si certains semblaient considérer que les sources sont nécessairement monophoniques, d'autres dont je faisais partie considéraient que les sources pouvaient être des signaux multicanaux. En d'autres termes, la distinction précise entre les sources et leurs images telle qu'introduite en section 1.2.2 n'était pas encore claire, bien que de nombreux travaux aient déjà traité de la question [50, 163, 47]. Dans ces conditions, le problème restait ouvert de savoir si la séparation informée devait permettre la récupération des sources ou bien de leurs images. De plus, il y a de nombreux cas de figures plus complexes envisageables, où les sources peuvent être observées sous forme d'un signal multicanal à l'encodeur, mais doivent être récupérées sous forme ponctuelle (un seul canal) au décodeur. On verra que le formalisme que je présente permet de réunir tous ces objectifs. On désignera simplement un certain nombre de sources comme devant être récupérées sous forme ponctuelle, tandis que seules les images des autres seront restituées.

Un autre problème posé par la séparation informée est celui de ses performances bornées. La plupart des techniques de séparation ne *peuvent* pas récupérer les sources originales en général avec n'importe quelle précision, même si elles sont utilisées avec les paramètres optimaux. Ce phénomène est connu comme l'existence de *bornes* dans la qualité des estimations, appelées performances oracles. Dans le cas de l'inversion locale, cette limitation est introduite par la mise à zéro de certains coefficients qui ne le sont pas réellement, même si la matrice de mélange est exactement connue. Le formalisme gaussien que j'ai initialement proposé et qu'on verra en partie III souffre lui aussi du même problème : on ne peut pas récupérer exactement les sources initiales avec ces techniques, mais plutôt la meilleure approximation qui répond à un modèle. Ce phénomène est réminiscent de ce qui se passe en audio dans le cas du codage paramétrique : quelle que soit la qualité du modèle, le signal reconstruit ne pourra qu'appartenir à une certaine classe de signaux, ce qui n'est pas forcément le cas de l'original [33].

À l'inverse, le codage multicanal *enhanced*-SAOC permet de passer outre cette limitation en envoyant tout simplement l'erreur d'estimation elle-même au décodeur en utilisant pour cela un codage de *forme d'onde* tel que AAC. De la même manière, une méthode hybride de l'inversion locale consiste à transmettre les résiduels obtenus après séparation [170]. Ce faisant, il devient garanti que l'augmentation du débit provoque nécessairement une meilleure qualité de restitution des sources. Cependant, ce codage des résiduels est effectué a posteriori et ne prend pas en compte l'étape de séparation. La question de l'optimalité d'une telle procédure était donc ouverte.

Le chaînon manquant entre la séparation informée et le codage multicanal m'a été suggéré par Alexey Ozerov comme étant la théorie du codage de source [12]. Comme on le verra au chapitre 9, la séparation informée envisagée d'un point de vue Bayésien [110] conduit à considérer la *distribution* a posteriori $p(\tilde{\boldsymbol{s}} \mid \tilde{\boldsymbol{x}}, \Theta)$ des sources $\tilde{\boldsymbol{s}}$ étant donnés les mélanges $\tilde{\boldsymbol{x}}$ et l'information

12. Le codage de source [89], issu de la théorie de l'information [186], se concentre sur le problème d'encoder le plus efficacement possible une variable aléatoire, disons $\tilde{\boldsymbol{s}}$, caractérisée par sa distribution, ou *modèle*, $p(\tilde{\boldsymbol{s}} \mid \Theta)$, dont Θ sont les paramètres. L'enjeu en est la compression efficace des données. Voir le chapitre 13.

annexe Θ. En effet, cette distribution résume ce que l'on sait des sources *après* l'obtention de l'information annexe et l'observation des mélanges.

L'approche initiale que j'ai proposée (voir chapitre 9) choisit simplement comme estimées les sources les plus probables dans $p(\tilde{\boldsymbol{s}} \mid \tilde{\boldsymbol{x}}, \Theta)$. Ce faisant, la distorsion observée au décodeur correspond à la borne de Cramér-Rao de cet estimateur. L'idée d'Ozerov sur laquelle nous avons travaillé ensemble consiste à ne plus effectuer de telle *décision* pour l'estimation, mais à plutôt utiliser la théorie du codage pour représenter les sources en utilisant cette distribution a posteriori $p(\tilde{\boldsymbol{s}} \mid \tilde{\boldsymbol{x}}, \Theta)$. Ce cas de *codage informé* [207] permet de ne plus être borné en performances, puisqu'une augmentation du débit disponible produit un meilleur codage des sources. Par ailleurs, la transmission des résiduels ne nécessite pas celle d'un modèle supplémentaire, comme c'est le cas de SAOC ou dans la méthode par inversion locale hybride. Je présenterai ces idées plus en détail en partie IV.

La séparation informée par codage consiste à encoder les sources en utilisant leur distribution a posteriori. A l'opposé, un codage du résiduel comme SAOC procède d'abord à une séparation, puis encode l'erreur d'estimation par un modèle perceptif indépendant du modèle de séparation, ce qui est sous-optimal.

1.4 Plan de l'exposé

La mise au point de méthodes efficaces pour la séparation informée et de leur cadre théorique ont été deux activités que j'ai menées en parallèle durant l'ensemble de mon travail de thèse. Ainsi, le formalisme théorique que j'ai considéré a été constamment augmenté pour pouvoir prendre en compte des scénarios d'application de plus en plus variés. Cependant, j'ai choisi pour l'organisation de ce mémoire de présenter mes contributions de manière thématique plutôt que chronologique. Ainsi on trouvera quatre parties principales dans ce travail.

La première partie concerne la définition des processus gaussiens et les nombreuses approximations qu'on peut utiliser pour rendre ces modèles efficaces pour la régression. Elle contient pour commencer une présentation en détail de ce modèle de sources dans le chapitre 2. Ensuite, je montre comment certaines approximations peuvent permettre d'utiliser les processus gaussiens pour modéliser de grandes quantités de données au chapitre 3. J'introduis alors deux modèles paramétriques de sources dans le cadre gaussien qui sont importants pour mon travail au chapitre 4 et je montre comment leurs paramètres peuvent être estimés à partir des observations.

Dans une deuxième partie, j'aborde le problème de la séparation de processus gaussiens dans sa formulation générale, qui constitue le cœur théorique de cet exposé. Au chapitre 5, je montre ainsi comment on peut séparer une somme de processus gaussiens, avant de m'attaquer aux cas plus complexes des mélanges convolutifs et diffus au chapitre 6. Ces méthodes requérant la connaissance de certains paramètres de séparation, je montre au chapitre 7 comment ils peuvent être estimés à partir des mélanges, conduisant à ce que j'ai appelé une séparation *semi-informée*. Je présente enfin au chapitre 8 deux applications de ce formalisme à des cas concrets que j'ai envisagés au cours de mon travail, illustrant ainsi l'intérêt de l'approche.

Dans une troisième partie, ces modèles gaussiens pour la séparation sont appliqués au cas informé. Au chapitre 9, je montre comment la connaissance *a priori* des sources s'intègre naturellement dans le formalisme proposé. Les sources observées permettent de choisir correctement les paramètres d'un modèle, qui sont transmis au décodeur et utilisés lors de la séparation. Aux chapitres 10 et 11, nous verrons comment ces paramètres sont estimés, ce qui revient à étudier la nature du codeur considéré. Enfin, le système complet sera évalué au chapitre 12 et comparé avec les autres méthodes existantes.

Dans une quatrième partie, le problème de la séparation informé sera abordé sous l'angle différent du codage de source. Ainsi, les méthodes de séparation informée présentées en deuxième partie seront généralisées au cas où les sources ne sont plus estimées comme la seule moyenne de leur distribution a posteriori, mais plutôt encodées en utilisant cette distribution. Dans le chapitre 13, je présente les résultats de la théorie du codage de source nécessaires à la compréhension du codage a posteriori. Au chapitre 14, j'introduis l'idée du codage a posteriori et je montre comment elle permet d'aborder efficacement la question de la séparation informée. Au chapitre 15, je montre

avec plus de détails comment cette idée peut être mise en pratique pour la réalisation d'un codeur informé opérationnel. Enfin, le chapitre 16 présente une comparaison des performances du codage informé par rapport aux variantes paramétriques de la troisième partie.

Pour finir, une partie de conclusion me permet de dresser un bilan général des résultats obtenus pour chacune des parties de mon exposé au chapitre 17, avant d'esquisser au chapitre 18 certaines questions encore ouvertes ou pistes de recherche concernées par mon travail.

Première partie

Processus gaussiens

Chapitre 2

Processus gaussiens

2.1 Motivations

Les processus gaussiens [140, 178, 185, 218] permettent de modéliser des *fonctions*. Or, la plupart des signaux qu'on peut être amené à considérer en pratique sont des cas particuliers de fonctions.

Par exemple, un signal audio peut être compris comme une fonction de $\mathbb{Z}$ dans $\mathbb{R}$ qui donne la valeur (réelle) d'une forme d'onde pour chaque instant (discret) d'échantillonnage. De la même manière, un ensemble de P séries temporelles est une fonction [1] de $\mathbb{R} \times \mathbb{N}_P$ dans $\mathbb{R}$ qui donne pour chaque couple (t, p) d'instant t et de numéro d'onde p la valeur de la $p^{\text{ème}}$ onde à l'instant t.

Il est équivalent de considérer une fonction $\boldsymbol{g}(t)$ définie sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}^J$, c'est-à-dire à valeurs vectorielles, qu'une fonction $\tilde{g}$ définie sur $\mathbb{T} \times \mathbb{N}_J$ et à valeurs dans $\mathbb{C}$. Il suffit en effet de poser $\tilde{g}(t, j) = [\boldsymbol{g}(t)]_j$.

Les signaux financiers sont un autre cas de signaux de la sorte, où P différents indices de valeurs boursières évoluent conjointement au cours du temps. On verra au chapitre 8 un exemple complexe de signal, correspondant à la position spatiale au cours du temps des articulations d'une personne qui danse. Si on considère P articulations et que la position de chacune dans l'espace $\mathbb{R}^3$ est donnée à des instants réguliers d'échantillonnage, le signal peut être compris comme une fonction définie sur $\mathbb{N} \times \mathbb{N}_P \times \mathbb{N}_3$ et à valeurs dans $\mathbb{R}$. Ainsi, $\tilde{s}(n, a, 3)$ représente l'élévation ($3^{\text{ème}}$ coordonnée) de la $a^{\text{ème}}$ articulation du danseur à l'instant n (voir figure 8.3 page 104).

En géostatistiques, un signal donnera souvent la valeur d'une grandeur physique en fonction du point de l'espace considéré, et sera donc une fonction de $\mathbb{R}^3$ dans $\mathbb{R}$. Dans certains cas, cette valeur peut dépendre du temps, auquel cas la fonction considérée sera définie sur $\mathbb{R}^4$ et à valeurs dans $\mathbb{R}$.

Les sources sont définies sur un ensemble quelconque $\mathbb{T}$. Les méthodes présentées peuvent ainsi s'appliquer pour des cas inédits de problèmes de séparation.

Dans tous les exemples précédents, l'objet principal de l'étude peut ainsi être compris comme une *fonction* qui est définie sur un espace quelconque $\mathbb{T}$ et qui prend ses valeurs dans $\mathbb{C}$. Dans tout cet exposé, $\mathbb{T}$ sera appelé le *domaine de définition* des sources. Il s'agit d'un ensemble quelconque. Ainsi, les signaux sources seront par définition des fonctions de $\mathbb{T}$ dans $\mathbb{C}$. Dans de nombreux cas, elles sont à valeurs dans $\mathbb{R}$.

Bien entendu, dans de nombreux cas pratiques, le domaine de définition $\mathbb{T}$ ne sera pas quelconque. La plupart du temps, il disposera d'une structure algébrique particulière que nous pourrons exploiter. En particulier, $\mathbb{T}$ peut être un groupe, ce qui implique que $\forall (t, t') \in \mathbb{T}^2$, $t - t' \in \mathbb{T}$. C'est par exemple le cas pour des séries temporelles ou pour des données spatiales. Comme on le verra en section 2.5, cette propriété ouvre la voie à la considération de processus stationnaires. Cependant, de telles structures ne sont pas nécessaires à l'établissement du formalisme et sauf mention contraire, le domaine de définition $\mathbb{T}$ sera supposé quelconque.

1. $\mathbb{N}_P$ est l'ensemble des P premiers entiers naturels.

Muni de la définition de l'objet mathématique à étudier, une fonction $\tilde{s}$ de $\mathbb{T}$ dans $\mathbb{C}$, le praticien est confronté à plusieurs problèmes concrets. Tout d'abord, il n'observe en général cette fonction que pour un ensemble fini $T = [t_1, \dots, t_L]$ de L positions dans $\mathbb{T}$. Il n'a donc à sa disposition la connaissance que de T et de :

$$\tilde{\boldsymbol{s}}(T) = [\tilde{s}(t_1), \dots, \tilde{s}(t_L)]^\top .$$

Son problème est alors souvent d'avoir une indication sur la valeur $\tilde{\boldsymbol{s}}(T')$ de la fonction à d'autres positions $T' \in \mathbb{T}^{L'}$. Par exemple, dans le domaine de la prospection minière, on cherche souvent à déduire la concentration du sol en pétrole à partir de relevés pris en un nombre très limité de positions (à cause du coût élevé de chaque mesure). En traitement du signal audio, on cherchera à reconstruire une forme d'onde abîmée localement par des bruits de craquement. En analyse financière, on cherchera à estimer la valeur des signaux boursiers dans l'avenir à partir de leur observation dans le passé. Tous ces problèmes portent communément les noms d'*interpolation* ou d'*extrapolation* en fonction de la localisation des points de T' par rapport à ceux de T.

Un autre problème communément rencontré par le chercheur est que ses données observées sur T présentent une incertitude, ou sont bruitées. Cela arrive en géostatistiques à cause d'incertitude sur les appareils de mesure [44, 32] ou en télécommunications lorsque le signal observé a été déformé lors de sa transmission. Dans ce cas, on considère habituellement que ce n'est pas le signal cible $\tilde{s}$ qui est observé, mais plutôt son mélange $\tilde{x}$ avec un signal parasite $\tilde{\epsilon}$:

$$\forall t \in \mathbb{T},\ \tilde{x}(t) = \tilde{s}(t) + \tilde{\epsilon}(t) . \tag{2.1.1}$$

L'objectif des traitements est souvent dans ce cas d'obtenir de bonnes estimées des valeurs de $\tilde{\boldsymbol{s}}(T')$ à partir de l'observation de $\tilde{\boldsymbol{x}}(T)$. Ces estimées peuvent être désirées aux points d'observation ($T' = T$), auquel cas on parle souvent de *débruitage* ou de *lissage*, ou bien en un ensemble T' quelconque de points. On parle souvent de régression, ou de Krigeage dans ces situations plus générales.

Tous ces objectifs ne peuvent être atteints que si un modèle est disponible pour la fonction $\tilde{s}$ étudiée, ainsi que pour le signal de bruit $\tilde{\epsilon}$ dans 2.1.1. En effet, si on ne suppose rien sur leur nature, aucune prédiction en des positions non observées ne pourra nous sembler plus plausible qu'une autre.

De nombreuses méthodes ont été proposées dans le but de modéliser la structure de la fonction $\tilde{s}$ étudiée. Une première solution est de supposer qu'elle appartient à une certaine famille connue de fonctions. Par exemple, on peut supposer que $\tilde{s}$ est linéaire, exponentielle ou trigonométrique si son domaine de définition le permet. Dans ce cas, prédire sa valeur en n'importe quel point devient équivalent à déterminer à partir des observations les paramètres qui la caractérisent, comme sa pente, ses facteurs d'amortissement ou ses coefficients trigonométriques [2]. Dans de nombreux cas, un modèle paramétrique est efficace pour modéliser un phénomène, surtout quand certaines considérations physiques viennent supporter le choix du modèle. Par exemple, l'approche paramétrique a été largement appliquée pour la modélisation de la forme d'onde glottique [67, 79, 102].

Cependant, il arrive dans certains cas que le modèle choisi ne corresponde pas aux données, ou bien même que rien ne permette de privilégier un modèle plutôt qu'un autre. Dans ces cas, l'approche paramétrique touche à ses limites et il devient essentiel d'utiliser un modèle qui permette une plus grande souplesse. Dans ce travail, je vais me concentrer sur un autre formalisme où on ne fait pas de telles hypothèses sur la nature globale de la fonction étudiée. En ce sens, on parle d'approche *non paramétrique*. Sans rentrer encore dans les détails, son principe est d'assigner une *probabilité* $p(\tilde{s} \mid \mathcal{H})$ à *toutes les fonctions* $\tilde{s}$ qui sont définies sur $\mathbb{T}$ et qui prennent leurs valeurs dans $\mathbb{C}$, c'est-à-dire à tous les éléments de $\mathbb{C}^{\mathbb{T}}$.

Bien que toutes les fonctions se voient attribuer une probabilité, certaines seront considérées comme plus plausibles. C'est le sens du lot *d'hyperparamètres* $\mathcal{H}$, qui peut par exemple mener à favoriser les fonctions *lisses*. La distinction entre la notion de paramètre et celle d'hyperparamètre est subtile. Dans l'approche non paramétrique, rien n'oblige à considérer une quelconque fonction comme *impossible*. Seule compte sa probabilité. Ainsi, un modèle non paramétrique ne contraindra pas la fonction étudiée à être une droite ou une exponentielle, caractérisée par un lot de *paramètres*

2. C'est le principe de l'interpolation Lagrangienne.

comme sa pente ou son amortissement. Par contre, toutes les fonctions possibles ne sont pas également probables et les *hyperparamètres* permettent de préciser lesquelles sont plus plausibles que d'autres, sans forcément en exclure pour autant [3].

Sans savoir si la fonction recherchée est linéaire ou exponentielle, on peut simplement s'attendre à ce qu'elle soit *dérivable*. Comment en choisir un modèle paramétrique ?

En figure 2.1 (a), j'ai illustré cette notion centrale de distribution sur des fonctions avec un exemple qui favorise les fonctions *lisses*. Ainsi, j'ai représenté 3 réalisations typiques de fonctions issues de cette distribution [4]. Conceptuellement, ce tirage est similaire à celui d'un dé. La seule différence est qu'au lieu de tirer des éléments de $\mathbb{N}_6$, on tire des éléments de $\mathbb{R}^{\mathbb{T}}$.

La grande force du modèle adopté est qu'après l'observation des données $\tilde{\boldsymbol{s}}(T)$, les probabilités de *toutes* les fonctions sont mises à jour pour devenir une probabilité a posteriori $p(\tilde{s} \mid T, \tilde{\boldsymbol{s}}(T), \mathcal{H})$ qui favorise celles en accord avec les données. Dans la figure 2.1 (b), j'ai ainsi représenté 3 réalisations de fonctions issues de cette distribution a posteriori. On peut remarquer qu'elles ont toutes la particularité d'être à la fois lisses et de passer par les points observés. Leur *moyenne* a posteriori donne une solution naturelle au problème de la régression. Du fait de l'approche probabiliste adoptée, une *incertitude* sur les estimées est en outre disponible.

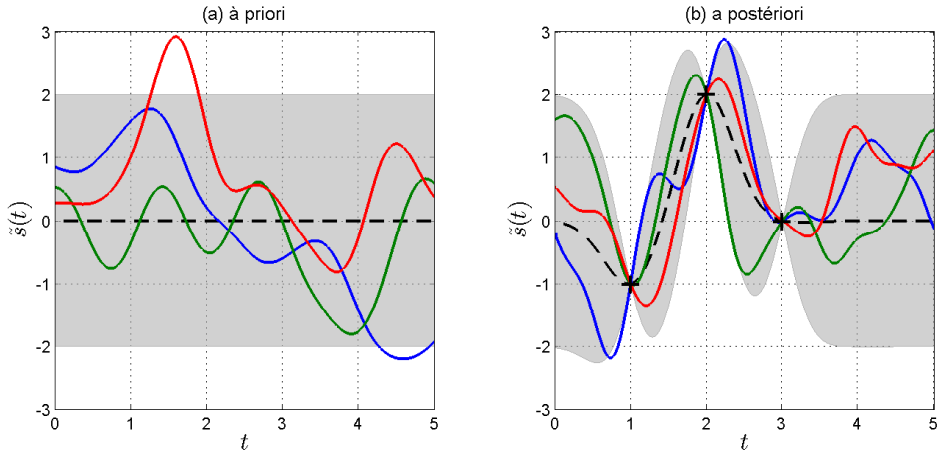

FIGURE 2.1: Réalisations de processus gaussiens en utilisant (a) une distribution *a priori* sur les fonctions et (b) la distribution a posteriori, i.e. mise à jour après observation de $L = 3$ mesures en $T = [1, 2, 3]$. Le trait en pointillé correspond à la fonction moyenne et la zone grisée correspond à 2 fois l'écart type en chacun des points t.

Les processus gaussiens assignent des probabilités à toutes les fonctions de $\mathbb{T}$ dans $\mathbb{C}$, de la même manière qu'une loi gaussienne assigne des probabilités à tous les réels.

Une telle approche probabiliste permet d'envisager tous les problèmes de régression d'une manière naturelle. La principale difficulté pour la mettre en œuvre est bien entendu de pouvoir attribuer une probabilité à chaque élément d'un ensemble aussi grand que $\mathbb{C}^{\mathbb{T}}$. C'est sur ce point que les processus gaussiens interviennent comme des distributions sur des espaces de fonctions [140, 218, 185, 178]. Pour en comprendre intuitivement le sens, on peut se représenter une fonction comme un vecteur de longueur infinie, qui donne une valeur pour chacun des points de $\mathbb{T}$. Un *processus gaussien* établit une distribution sur de telles

3. Ces subtiles différences sont discutées en détail dans [203]. Un modèle Bayésien non paramétrique n'impose pas à la fonction étudiée d'appartenir à un espace de fonctions de dimension finie. Au contraire, il peut assigner une probabilité non nulle à toutes les fonctions de l'espace. Il constitue ainsi une distribution *a priori non dégénérée* [110].

4. On trouvera en annexe A la méthode utilisée pour la synthèse de processus gaussiens.

données, de la même manière qu'une *distribution* gaussienne considère le cas de vecteurs d'une longueur finie.

2.2 Définition

2.2.1 La distribution gaussienne

Considérons un instant un scalaire $v \in \mathbb{R}$ qui n'est pas observé et dont on ne connaît donc pas la valeur. On peut quantifier notre incertitude à son sujet par le biais de sa *densité de probabilité* $p(v \mid \mathcal{H})$, où $\mathcal{H}$ représente l'ensemble de nos connaissances [5]. Pour un ensemble $T \subset \mathbb{R}$, la *probabilité* que v soit dans cet ensemble devient

$$P(v \in T \mid \mathcal{H}) = \int_{v \in T} p(v \mid \mathcal{H})\, dv.$$

Dans toute la suite et pour des raisons de clarté, je parlerai souvent abusivement de *probabilités* pour désigner des *densités de probabilités*. C'est en effet l'usage fréquent pour des variables aléatoires à valeurs scalaires.

Le choix d'une distribution particulière $p(v \mid \mathcal{H})$ en fonction de nos connaissances $\mathcal{H}$ n'est pas un problème trivial. Ce problème a même fait l'objet d'une multitude de travaux. Son enjeu est de produire une distribution $p(v \mid \mathcal{H})$ qui respecte $\mathcal{H}$ mais qui ne revienne pas non plus à faire des hypothèses supplémentaires. Dans ce travail, je considérerai la solution qui consiste à choisir $p(v \mid \mathcal{H})$ comme la distribution compatible avec $\mathcal{H}$ qui présente l'entropie maximale [110].

La question de la signification des probabilités a fait l'objet de nombreuses études. Pour ma part, je me bornerai à considérer une probabilité $p(v \mid \mathcal{H})$ comme la quantification d'une *certitude* sur la valeur de v étant donné un état $\mathcal{H}$ de connaissance [110].

On peut montrer [3, 120] que si $\mathcal{H}$ contient la moyenne $\mu = \mathbb{E}[v]$ de v ainsi que sa variance $\sigma^2 = \mathbb{E}\left[(v-\mu)^2\right]$, alors la distribution $p\left(v \mid \mu, \sigma^2\right)$ qui maximise l'entropie est la distribution gaussienne, ou *normale* :

$$p\left(v \mid \mu, \sigma^2\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(v-\mu)^2}{2\sigma^2}\right). \tag{2.2.1}$$

Bien entendu, la distribution gaussienne est très connue. Je l'ai représentée en figure 2.2. Comme on le sait, elle donne une plus grande densité de probabilité à la moyenne, tandis que sa largeur est directement liée à la variance.

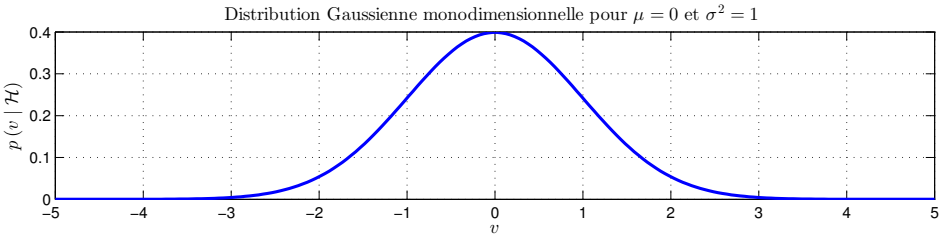

FIGURE 2.2: Distribution gaussienne scalaire

La connaissance de la moyenne et de la variance d'une grandeur étudiée est un cas assez fréquent et peut s'interpréter facilement. La moyenne indique la valeur autour de laquelle on s'attend à trouver cette grandeur, tandis que la variance quantifie notre incertitude. Le praticien dispose souvent en pratique d'une telle connaissance. En audio par exemple, les signaux sont la plupart

5. Pour des raisons de simplicité de l'exposé, je considère le cas de variables aléatoires dont la densité de probabilité existe.

du temps centrés ($\mu = 0$) tandis que leur variance est liée à leur puissance. Le cas extrême d'une variance nulle indiquerait une certitude sur la valeur $v = \mu$.

Cette manière d'introduire la distribution gaussienne permet d'en expliquer l'usage omniprésent dans les sciences quantitatives. Son succès est interprété depuis quelques décennies [110] comme le signe de la validité du critère de maximum d'entropie pour le choix d'une distribution. Si on ne connaît que la moyenne et l'écart à la moyenne d'une grandeur, alors la distribution gaussienne est la plus permissive (au sens d'une entropie maximale) qui respecte cette connaissance.

Ces considérations permettent de justifier l'usage de méthodes probabilistes sur des données déterministes, c'est-à-dire où l'incertitude n'a pas de rôle à jouer pour peu qu'on connaisse le mécanisme à l'œuvre. En effet, l'approche Bayésienne des probabilités ne nécessite pas l'intervention du hasard dans le processus étudié. Au contraire, la seule incertitude qu'elle envisage est celle de l'observateur. Ainsi, face à des signaux déterministes, ce sera le mécanisme à l'œuvre qui sera inconnu, justifiant l'usage de méthodes probabilistes.

Considérons à présent L variables scalaires $a_l \in \mathbb{R}$ dont on ignore les valeurs. Dans ce cas, le vecteur $\boldsymbol{a} = [a_1, \ldots, a_L]^\top$ est un vecteur de dimension $L \times 1$ qui représente un point dans l'espace $\mathbb{R}^L$. Pour $L = 2$, c'est un point du plan. L'approche probabiliste nous conduit à choisir une densité de probabilité *jointe* $p(\boldsymbol{a} \mid \mathcal{H}) = p(a_1, \ldots, a_L \mid \mathcal{H})$ pour décrire notre état de connaissance sur la position de ce point. Ainsi, $p(\boldsymbol{a} \mid \mathcal{H})$ indique pour chaque point de $\mathbb{R}^L$ s'il est plausible que $\boldsymbol{a}$ s'y trouve. De la même manière que dans le cas univarié, on peut montrer que si on connaît la moyenne

$$\mu_l = \mathbb{E}[a_l]$$

de chacun de ses éléments et la covariance [6]

$$k(l, l') = \mathbb{E}\left[(a_l - \mu_l)(a_{l'} - \mu_{l'})^*\right]$$

entre a_l et $a_{l'}$, alors la distribution au maximum d'entropie qui vérifie ces critères est la distribution gaussienne multivariée. Elle s'écrit :

$$p(\boldsymbol{a} \mid \boldsymbol{\mu}, K) = \frac{1}{(2\pi)^{\frac{L}{2}}\sqrt{|K|}} \exp\left(-\frac{1}{2}(\boldsymbol{a} - \boldsymbol{\mu})^\top K^{-1}(\boldsymbol{a} - \boldsymbol{\mu})\right) \tag{2.2.2}$$

où $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_L]^\top$ est le vecteur moyenne et où $[K]_{l,l'} = k(l, l')$ est la *matrice de covariance,* de dimension $L \times L$. K^{-1} désigne l'inverse de K et $|K|$ son déterminant [7]. On dit alors que $\boldsymbol{a}$ est distribué selon une loi gaussienne de moyenne $\boldsymbol{\mu}$ et de matrice de covariance K et ceci se note :

$$\boldsymbol{a} \sim \mathcal{N}(\boldsymbol{\mu}, K).$$

Cette distribution est représentée en figure 2.3 pour le cas de $L = 2$. Les lignes de niveau qu'on peut y voir indiquent que la distribution jointe $p(a, b \mid \mathcal{H})$ privilégie les couples $[a, b]$ qui se situent dans certaines régions de l'espace : tout d'abord, on voit que plus un point est proche de la moyenne 0, plus il est probable. Ensuite, la distribution favorise les couples se situant le long des *vecteurs propres* $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ et $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$ de sa matrice de covariance $K = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$.

Les distributions marginales, ou *a priori*, $p(a \mid \mathcal{H})$ et $p(b \mid \mathcal{H})$ sont obtenues en sommant la distribution jointe le long des axes :

$$p(a \mid \mathcal{H}) = \int_b p(a, b \mid \mathcal{H})\, db. \tag{2.2.3}$$

Dans le cas de la distribution gaussienne, une particularité des distributions marginales est qu'elles sont elles-mêmes gaussiennes et obtenues en sélectionnant dans le vecteur moyen $\boldsymbol{\mu}$ et la matrice de covariance K de la distribution jointe les seules lignes et colonnes faisant intervenir les variables considérées.

6. $\cdot^*$ désigne la conjugaison complexe.

7. Une autre caractérisation de $p(\boldsymbol{a} \mid \boldsymbol{\mu}, K)$ faisant intervenir la fonction génératrice est nécessaire en cas de singularité de K.

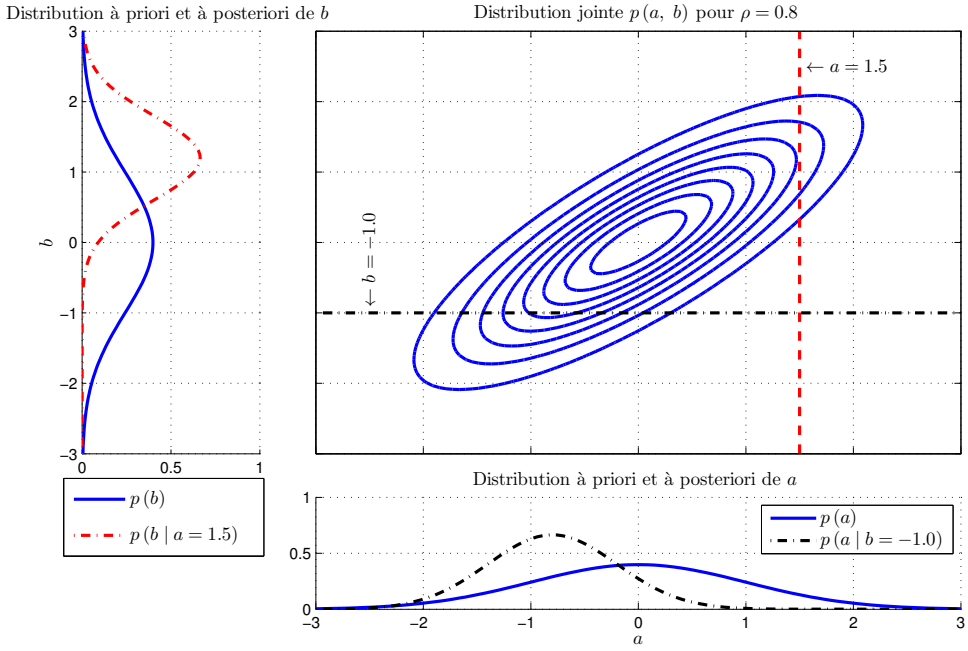

FIGURE 2.3: Distribution gaussienne bivariée de matrice de moyenne $\mu = 0$ et de covariance $K = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$. Distribution jointes, distributions marginales et conditionnelles.

Dans notre exemple, on voit ainsi que la distribution marginale de a est une gaussienne de moyenne nulle et de variance unité, ce qui correspond bien à la distribution jointe dont on n'a gardé que μ_1 et K_{11}. Cette propriété se généralise. Soient L_a et L_b deux entiers naturels et $\boldsymbol{a}$ et $\boldsymbol{b}$ deux vecteurs de $\mathbb{R}^{L_a}$ et de $\mathbb{R}^{L_b}$. Supposons que leur distribution jointe soit gaussienne :

La distribution marginale, ou *a priori*, de a est la somme de la distribution jointe sur l'axe b. Pour une distribution jointe gaussienne, elle est gaussienne.

$$\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix} \mid \mathcal{H} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{bmatrix}, \begin{bmatrix} K(\boldsymbol{a},\boldsymbol{a}) & K(\boldsymbol{a},\boldsymbol{b}) \\ K(\boldsymbol{b},\boldsymbol{a}) & K(\boldsymbol{b},\boldsymbol{b}) \end{bmatrix} \right), \tag{2.2.4}$$

où $\boldsymbol{\mu}_a$ et $\boldsymbol{\mu}_b$ sont les vecteurs moyens de $\boldsymbol{a}$ et $\boldsymbol{b}$, de dimensions $L_a \times 1$ et $L_b \times 1$ respectivement, tandis que $K(\boldsymbol{a},\boldsymbol{a})$, $K(\boldsymbol{b},\boldsymbol{a})$, $K(\boldsymbol{a},\boldsymbol{b})$ et $K(\boldsymbol{b},\boldsymbol{b})$ sont les différentes matrices de covariance de $\boldsymbol{a}$ et $\boldsymbol{b}$, de dimensions respectives $L_a \times L_a$, $L_b \times L_a$, $L_a \times L_b$ et $L_b \times L_b$. On peut montrer qu'on a alors :

$$\boldsymbol{a} \mid \mathcal{H} \sim \mathcal{N}(\boldsymbol{\mu}_a, K(\boldsymbol{a},\boldsymbol{a})) \tag{2.2.5}$$

et

$$\boldsymbol{b} \mid \mathcal{H} \sim \mathcal{N}(\boldsymbol{\mu}_b, K(\boldsymbol{b},\boldsymbol{b})). \tag{2.2.6}$$

En observant la figure 2.3, on note que la distribution $p(a, b \mid \mathcal{H})$ n'attribue pas la même valeur à b pour toutes les valeurs de a. Cela est le signe que ces deux variables sont modélisées comme étant *dépendantes*. Ainsi, la connaissance de l'une entraîne une modification de notre état de connaissance sur l'autre. En effet, si on suppose que la valeur de $a = 1.5$ est observée, la distribution jointe nous donne automatiquement une mise à jour de celle de b, appelée alors distribution *conditionnelle*, ou *a posteriori* : $p(b \mid a = 1.5, \mathcal{H})$. J'ai représenté cette distribution en trait rouge discontinu sur la figure 2.3. Elle peut se comprendre comme une *coupe* de la distribution jointe selon l'axe $a = 1.5$. En particulier, on remarque que la valeur la plus probable de b après observation de $a = 1.5$

n'est plus 0, mais une valeur bien plus proche de 1.5, ce qui est cohérent du fait de la corrélation forte supposée entre les deux variables. De la même manière, la distribution de a est modifiée par l'observation de $b = -1$ pour devenir $p(a \mid b = -1, \mathcal{H})$ représentée en trait noir discontinu sur la figure 2.3.

Comme on peut le voir sur la figure, toute coupe de la distribution gaussienne reste une distribution gaussienne. Cette propriété très forte se généralise en dimension quelconque. Si on considère à nouveau la distribution jointe 2.2.4 de $\boldsymbol{a} \in \mathbb{R}^{L_a}$ et $\boldsymbol{b} \in \mathbb{R}^{L_b}$, la distribution *a posteriori* de l'un de ces vecteurs après observation de l'autre reste une distribution gaussienne, dont les moyennes et matrice de covariance ont été mis à jour. Plus précisément, on a le résultat classique suivant [38] :

si

$$\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix} \mid \mathcal{H} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{bmatrix}, \begin{bmatrix} K(\boldsymbol{a},\boldsymbol{a}) & K(\boldsymbol{a},\boldsymbol{b}) \\ K(\boldsymbol{b},\boldsymbol{a}) & K(\boldsymbol{b},\boldsymbol{b}) \end{bmatrix} \right),$$

alors

$$\boldsymbol{a} \mid \boldsymbol{b}, \mathcal{H} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\text{post}}, K_{\text{post}}\right) \tag{2.2.7}$$

où

$$\begin{cases} \boldsymbol{\mu}_{\text{post}} &= \boldsymbol{\mu}_a + K(\boldsymbol{a},\boldsymbol{b}) K(\boldsymbol{b},\boldsymbol{b})^{-1} (\boldsymbol{b} - \boldsymbol{\mu}_b) \\ K_{\text{post}} &= K(\boldsymbol{a},\boldsymbol{a}) - K(\boldsymbol{a},\boldsymbol{b}) K(\boldsymbol{b},\boldsymbol{b})^{-1} K(\boldsymbol{b},\boldsymbol{a}) \end{cases} \tag{2.2.8}$$

sont le vecteur moyen et la matrice de covariance *a posteriori* de $\boldsymbol{a}$ après observation de $\boldsymbol{b}$.

Comme on le verra en section 14.1.2, le déterminant de la variance *a posteriori* K_{post} dans l'équation 2.2.8 est nécessairement moins grand que celui de $K(\boldsymbol{a},\boldsymbol{a})$: une observation ne peut que réduire notre incertitude. Au pire, elle ne peut que laisser notre connaissance sur $\boldsymbol{a}$ inchangée si $\boldsymbol{a}$ et $\boldsymbol{b}$ sont indépendants, auquel cas $K(\boldsymbol{a},\boldsymbol{b}) = 0$.

Les expressions 2.2.8 sont fondamentales et seront mises en œuvre de très nombreuses fois dans ce document. Elles permettent dans le cas gaussien de tirer parti de l'observation de certaines variables pour en déduire les valeurs les plus probables d'autres qui leurs sont corrélées. Cette mécanique incorpore ainsi un phénomène *d'apprentissage* : si $p(\boldsymbol{a} \mid \mathcal{H})$ représente notre état de connaissance *a priori* sur $\boldsymbol{a}$, la distribution $p(\boldsymbol{a} \mid \boldsymbol{b}, \mathcal{H})$ nous renseigne sur notre état de connaissance après observation de $\boldsymbol{b}$, c'est-à-dire *a posteriori*. La particularité de la distribution gaussienne est que la distribution *a posteriori* se calcule facilement à partir de la distribution jointe et qu'elle est gaussienne.

Démonstration. Compte tenu du rôle central de ces expressions dans cet exposé, j'en rappelle ci-dessous une démonstration rapide. Dans le but d'avoir des expressions concises, je montre ce résultat pour des vecteurs aléatoires a et b de moyenne nulle [8] et je remplace exceptionnellement les notations de type $K(\boldsymbol{a},\boldsymbol{b})$ par K_{ab}. La probabilité jointe d'une réalisation $\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix}$ est donnée par :

$$p(\boldsymbol{a},\boldsymbol{b}) \propto \exp\left(-\frac{1}{2} \begin{bmatrix} \boldsymbol{a}^H & \boldsymbol{b}^H \end{bmatrix} \begin{bmatrix} K_{aa} & K_{ab} \\ K_{ba} & K_{bb} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix} \right). \tag{2.2.9}$$

Or, on montre que [9] :

$$\begin{bmatrix} K_{aa} & K_{ab} \\ K_{ba} & K_{bb} \end{bmatrix}^{-1} = \begin{bmatrix} I_a & 0 \\ \left(-K_{bb}^{-1} K_{ba}\right) & I_b \end{bmatrix} \begin{bmatrix} \left(K_{aa} - K_{ab} K_{bb}^{-1} K_{ba}\right)^{-1} & 0 \\ 0 & K_{bb}^{-1} \end{bmatrix} \begin{bmatrix} I_a & -K_{ab} K_{bb}^{-1} \\ 0 & I_b \end{bmatrix}, \tag{2.2.10}$$

8. Les mêmes résultats sont obtenus pour des vecteurs de moyenne non nulle en remplaçant partout $\boldsymbol{a}$ par $\boldsymbol{a} - \boldsymbol{\mu}_a$ et $\boldsymbol{b}$ par $\boldsymbol{b} - \boldsymbol{\mu}_b$.

9. Le résultat 2.2.10, classique, fait intervenir le complément de Schur $K_{aa} - K_{ab} K_{bb}^{-1} K_{ba}$ de K_{bb} dans la matrice de covariance jointe.

où I_a et I_b sont des matrices identité. L'expression 2.2.10 mène par simple substitution dans 2.2.9 à :

$$p(\boldsymbol{a},\boldsymbol{b}) \propto \exp\left(-\frac{1}{2}\left(\boldsymbol{a}-K_{ab}K_{bb}^{-1}\boldsymbol{b}\right)^H\left(K_{aa}-K_{ab}K_{bb}^{-1}K_{ba}\right)^{-1}\left(\boldsymbol{a}-K_{ab}K_{bb}^{-1}\boldsymbol{b}\right)\right)\exp\left(-\frac{1}{2}\boldsymbol{b}^H K_{bb}^{-1}\boldsymbol{b}\right), \tag{2.2.11}$$

et l'expression de $p(\boldsymbol{a} \mid \boldsymbol{b})$ est obtenue directement à partir de 2.2.11 en la divisant par :

$$p(\boldsymbol{b}) \propto \exp\left(-\frac{1}{2}\boldsymbol{b}^H K_{bb}^{-1}\boldsymbol{b}\right).$$

On y reconnaît l'expression d'une densité de probabilité gaussienne dont moyenne et covariance sont données par 2.2.8. □

2.2.2 Processus gaussiens

Une *distribution* gaussienne concerne des vecteurs de longueur L finie, et donc des éléments de $\mathbb{C}^L$. Un *processus gaussien* [140, 178, 218, 185] en est l'extension directe au cas de vecteurs de dimension infinie, c'est-à-dire de *fonctions*. On peut le définir simplement [178] :

Définition 1. Un processus gaussien est une collection de variables aléatoires, dont tout ensemble fini a une distribution jointe gaussienne multivariée.

Considérons un ensemble $\mathbb{T}$ et L points $T = [t_1, \ldots, t_L]$ de cet ensemble. Une fonction $\tilde{s} \in \mathbb{C}^{\mathbb{T}}$ de $\mathbb{T}$ dans $\mathbb{C}$ est un processus gaussien si pour tout L et pour tout T

$$\tilde{\boldsymbol{s}}(T) = [\tilde{s}(t_1), \ldots, \tilde{s}(t_L)]^\top$$

est distribué selon une loi gaussienne multivariée :

$$\begin{bmatrix} \tilde{s}(t_1) \\ \vdots \\ \tilde{s}(t_L) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu(t_1) \\ \vdots \\ \mu(t_L) \end{bmatrix}, \begin{bmatrix} k(t_1,t_1) & \cdots & k(t_1,t_L) \\ \vdots & k(t_l,t_{l'}) & \vdots \\ k(t_L,t_1) & \cdots & k(t_L,t_L) \end{bmatrix}\right), \tag{2.2.12}$$

également notée

$$\tilde{\boldsymbol{s}}(T) \sim \mathcal{N}\left(\mu(T), K\left(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T)\right)\right),$$

où $\mu(t)$ désigne la moyenne du processus en $t \in \mathbb{T}$:

$$\forall t \in \mathbb{T},\ \mu(t) = \mathbb{E}[\tilde{s}(t)]$$

et $k(t,t')$ donne la covariance de $\tilde{s}(t)$ et $\tilde{s}(t')$:

$$\forall (t,t') \in \mathbb{T}\times\mathbb{T},\ k(t,t') = \mathbb{E}\left[\left(\tilde{s}(t)-\mu(t)\right)\left(\tilde{s}(t')-\mu(t')\right)^*\right].$$

On dira alors que $\tilde{s}$ est un processus gaussien de fonction moyenne $\mu : \mathbb{T} \to \mathbb{C}$ et de fonction de covariance $k : \mathbb{T}\times\mathbb{T} \to \mathbb{C}$. Ceci s'écrira :

$$\tilde{s} \sim \mathcal{PG}\left(\mu(t), k(t,t')\right).$$

Si la fonction moyenne peut être comprise simplement comme la valeur autour de laquelle on s'attend à trouver la valeur de la fonction, pour chaque point de l'espace, la fonction de covariance, plus subtile, donne la corrélation qu'on s'attend à trouver entre ses valeurs en deux points de l'espace. Par exemple, si la fonction $\tilde{s}$ est supposée *lisse*, cela signifie qu'on s'attend à une forte corrélation entre deux points qui sont proches dans $\mathbb{T}$. Dans ce cas, on pourra par exemple *choisir* une fonction de covariance exponentielle carrée (EC) [140], donnée pour $\mathbb{T} = \mathbb{R}$ par

$$k(t,t') = \sigma^2 \exp\left(-\frac{(t-t')^2}{\lambda^2}\right). \tag{2.2.13}$$

Ce choix garantira en effet que les valeurs de la fonction en deux points proches auront une forte covariance, tandis qu'elle tend vers 0 s'ils s'éloignent, provoquant leur indépendance. Ce comportement est illustré en figure 2.5.

Si on augmente la longueur caractéristique λ dans l'équation 2.2.13, il faudra augmenter la distance entre t et t' pour que $\tilde{s}(t)$ et $\tilde{s}(t')$ soient indépendants. Les réalisations typiques du processus seront alors plus lisses. D'une manière générale, le choix d'une fonction de covariance conditionne la géométrie des réalisations du processus gaussien correspondant [3]. En section 2.3, je présenterai certaines catégories et exemples usuels de fonctions de covariance.

Comme on le voit, la fonction $\tilde{s}$ étudiée n'est pas supposée appartenir à un espace paramétré de dimension finie. En effet, la définition même du processus ne considère aucune réalisation comme réellement *impossible*. Tout au plus, elle pourra être considérée comme extrêmement *improbable*. En ce sens, on n'a pas fait de modélisation paramétrique. Par contre, il va de soi qu'on ne considère pas toutes les réalisations comme également probables. C'est le sens des fonctions de moyenne et de covariance de permettre d'en préférer certaines aux autres. Si ces moyennes et covariances sont dépendantes de la valeur de certaines grandeurs, comme k l'est de σ^2 et λ dans l'équation 2.2.13, alors on parle volontiers d'*hyperparamètres* [218, 178], de manière à insister sur le fait que ces grandeurs n'imposent pas de restriction sur la nature de $\tilde{s}$, comme le font des *paramètres* classiques. Dans cet exposé, on notera θ l'ensemble des hyperparamètres d'un modèle, et la fonction de covariance sera alors notée $k(t, t' \mid \theta)$ ou simplement $k(t, t')$ quand le contexte est clair.

L'utilisation de processus gaussiens dans des problèmes d'ingénierie remonte au moins aux travaux de Wiener dans le domaine du traitement des séries temporelles stationnaires [217] et à Matheron [144] dans le domaine des géostatistiques, sous le nom de Krigeage, en hommage aux travaux précurseurs menés par l'ingénieur en prospection minière Krige [126]. Cependant, ils ne furent popularisés comme des modèles probabilistes non paramétriques puissants pour l'apprentissage automatique que bien plus tard [140, 218], lorsque leur lien avec les réseaux de neurones fut établi [219]. C'est alors qu'ils bénéficièrent de l'attention constante d'une large communauté de chercheurs [218, 185, 178, 140], intéressés par de telles *méthodes à noyaux* [184].

La caractérisation des fonctions de covariance comme des fonctions définies positives [1] (cf section 2.3 page suivante), et donc des produits scalaires, a donné lieu à de multiples rapprochements entre les processus gaussiens et les espaces Hilbertiens à noyaux reproduisants (*Reproducing Kernel Hilbert Spaces*, ou RKHS en anglais), qui sont des objets mathématiques introduits dans les années 1940 par Aronszajn [8, 9], puis étudiés par de nombreux mathématiciens [20] et qui permettent de modéliser des espaces Hilbertiens de fonctions. Certains chercheurs se sont attachés à étudier la géométrie de tels espaces [3].

2.2.3 Régression

Dans les cas pratiques évoqués en section 2.1 on dispose de l'observation du signal étudié $\tilde{s}$ en un nombre fini L de points $T = [t_1, \ldots, t_L]$ de $\mathbb{T}$. Cette observation peut de plus être bruitée, conduisant à ne pas observer le signal recherché $\tilde{s}$ directement, mais plutôt un autre signal $\tilde{x}$ qui lui est corrélé. L'objectif est de déduire de ces observations la valeur du signal $\tilde{s}$ en un ensemble quelconque $T' = [t'_1, \ldots, t'_{L'}]$ de positions dans $\mathbb{T}$. Je vais présenter la manière classique d'aborder ce problème avec des processus gaussiens. Elle forme la base de l'utilisation de ces processus pour la régression [218, 178, 185].

– Tout d'abord, le signal observé $\tilde{x}$ est supposé être la somme du signal utile $\tilde{s}$ avec un bruit additif $\tilde{\epsilon}$:

$$\forall t \in \mathbb{T},\ \tilde{x}(t) = \tilde{s}(t) + \tilde{\epsilon}(t).$$

L'approche consiste à supposer que $\tilde{s}$ et $\tilde{\epsilon}$ sont deux processus gaussiens indépendants. Le signal $\tilde{x}$, étant la somme de deux processus gaussiens, est lui-même un processus gaussien de fonctions de moyenne et de covariance :

$$\begin{cases} \mu_{\tilde{x}} &= \mu_{\tilde{s}} + \mu_{\tilde{\epsilon}} \\ k_{\tilde{x}} &= k_{\tilde{s}} + k_{\tilde{\epsilon}} \end{cases} \tag{2.2.14}$$

– Ensuite, on choisit une fonction de moyenne et de covariance pour le signal utile ainsi que pour le signal de bruit. Généralement, ce dernier est supposé être de moyenne nulle et de fonction de covariance *blanche* :
$$k_{\tilde{\epsilon}}(t,t') = \sigma^2 \delta_{tt'} \tag{2.2.15}$$
où $\delta_{tt'}$ est le symbole de Kronecker, qui vaut 1 si et seulement si $t = t'$ et 0 sinon. En d'autres termes, les valeurs de $\tilde{\epsilon}$ en deux points non identiques de $\mathbb{T}$ sont indépendantes : $\tilde{\epsilon}$ est supposé *blanc*. L'observation $\tilde{x}$ correspond ainsi bien au signal $\tilde{s}$ *bruité*. Quant au signal utile $\tilde{s}$, on lui choisit des fonctions de moyenne $\mu_{\tilde{s}}$ et de covariance $k_{\tilde{s}}$ qui correspondent à une connaissance *a priori*.
– C'est alors qu'on écrit la distribution jointe de $\tilde{\boldsymbol{s}}(T')$ et de $\tilde{\boldsymbol{x}}(T)$:
$$\begin{bmatrix} \tilde{\boldsymbol{s}}(T') \\ \tilde{\boldsymbol{x}}(T) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_{\tilde{s}}(T') \\ \boldsymbol{\mu}_{\tilde{x}}(T) \end{bmatrix}, \begin{bmatrix} K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{s}}(T')) & K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{x}}(T)) \\ K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{s}}(T')) & K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T)) \end{bmatrix} \right)$$
où chacune des sous matrices de covariance s'exprime très simplement en fonction de $k_{\tilde{s}}$ et $k_{\tilde{\epsilon}}$. Par exemple, du fait de l'indépendance de $\tilde{s}$ et $\tilde{\epsilon}$, on a :
$$[K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{s}}(T'))]_{l,l'} = k_{\tilde{s}}(t_l, t'_{l'})$$
et
$$K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T)) = K(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T)) + K(\tilde{\boldsymbol{\epsilon}}(T), \tilde{\boldsymbol{\epsilon}}(T))$$
– Enfin, on utilise les résultats 2.2.8 pour calculer la distribution de $\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{x}}(T)$:
$$\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{x}}(T) \sim \mathcal{N}\left(\boldsymbol{\mu}_{\text{post}}, K_{\text{post}}\right)$$
avec
$$\begin{cases} \boldsymbol{\mu}_{\text{post}} &= \boldsymbol{\mu}_{\tilde{s}}(T') + K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{x}}(T)) K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))^{-1} (\tilde{\boldsymbol{x}}(T) - \boldsymbol{\mu}_{\tilde{x}}(T)) \\ K_{\text{post}} &= K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{s}}(T')) - K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{x}}(T)) K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))^{-1} K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{s}}(T')) \end{cases}. \tag{2.2.16}$$
Cette distribution *a posteriori* nous donne à la fois une indication sur les valeurs les plus probables de $\tilde{s}$ en chacun des points de T' par le biais de sa moyenne, mais également une indication précieuse sur notre incertitude par le biais de sa matrice de covariance *a posteriori*. Chaque élément de la diagonale de K_{post} nous donne la variance de notre estimée en un point de T', et représente donc notre incertitude pour la valeur de $\tilde{s}$ autour de la moyenne estimée.

Bien entendu, nos estimées dépendront du modèle choisi pour $\tilde{s}$ et $\tilde{\epsilon}$. Par exemple, si σ est choisi nul, cela signifie que l'observation n'est pas considérée comme bruitée et que notre estimée passera nécessairement par les points observés. On est alors dans une situation d'*interpolation*. On retrouve cet exemple en figure 2.1. Au contraire, on peut choisir σ plus ou moins grand pour rendre compte d'une observation incertaine et également choisir différents hyperparamètres pour $k_{\tilde{s}}$. En figure 2.4, j'ai représenté les distributions *a posteriori* obtenues en utilisant la covariance exponentielle carrée 2.2.13 pour $k_{\tilde{s}}$, dont j'ai fait varier les hyperparamètres. On voit que les distributions *a posteriori* correspondantes sont différentes. J'ai de plus représenté quelques tirages de réalisations *a posteriori*, qui ont toutes la particularité de respecter les observations, tout en correspondant au modèle choisi.

Dans le domaine des géostatistiques, ces différents traitements portent le nom de Krigeage *ordinaire* [44, 32]. On verra en section 2.4 qu'il existe des techniques pour déterminer de manière automatique la valeur des hyperparamètres de $\tilde{s}$.

2.3 Fonctions de covariance

Une fonction de covariance est une fonction de $\mathbb{T} \times \mathbb{T}$ dans $\mathbb{C}$. Il s'agit donc d'une manière générale d'un type particulier de *noyau* [184] qui donne la covariance $k(t,t')$ des valeurs de la fonction étudiée en deux points de l'espace. Comme on l'a vu en section 2.2.2, elle est l'élément

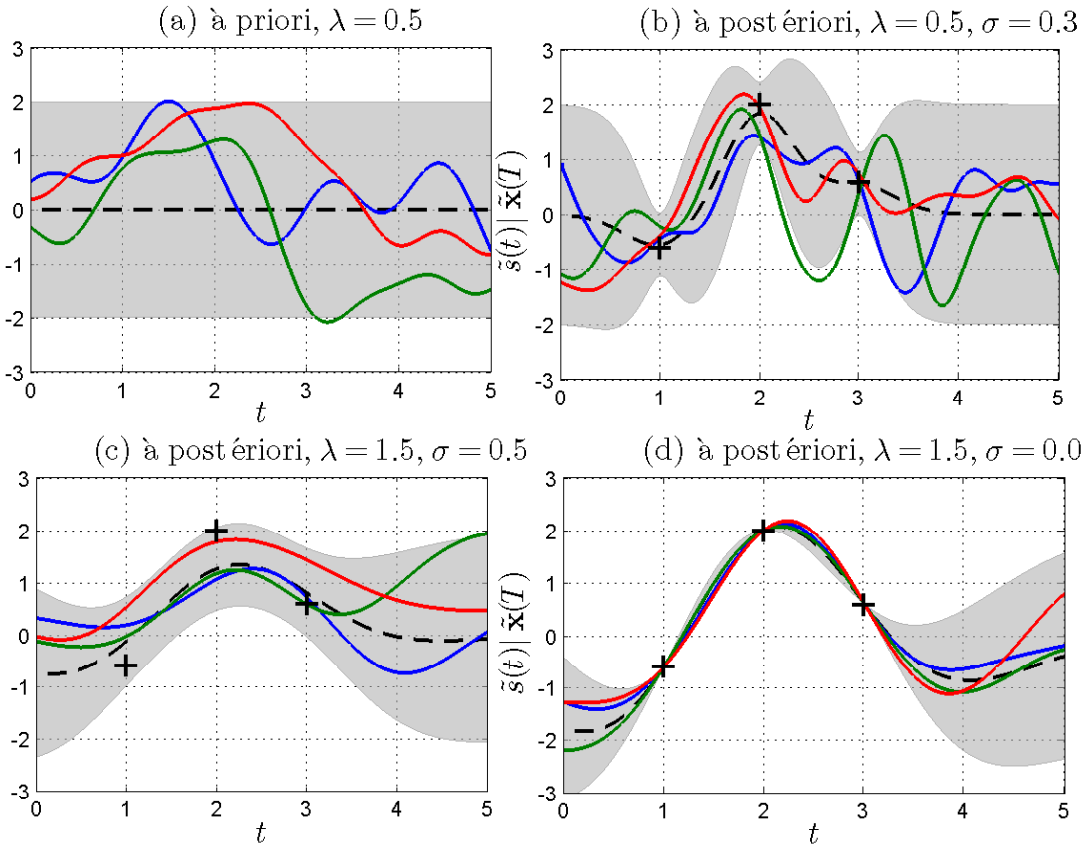

FIGURE 2.4: Régression avec des processus gaussiens. En (a), quelques réalisation du processus gaussien choisi. En (b,c,d), distribution *a posteriori* de $\tilde{\boldsymbol{s}}(T')$ pour différentes valeurs d'hyperparamètres. Les trois observations $\tilde{\boldsymbol{x}}(T)$ sont les points marqués d'une croix.

clé dans le choix d'un modèle de processus gaussien. Souvent en effet, la fonction moyenne μ est choisie nulle après centrage des données. Comme souligné dans [178], la fonction de covariance peut se comprendre intuitivement comme donnant la similarité qu'on s'attend à observer dans les valeurs de la fonction étudiée en deux points quelconques. Ainsi comprise, il est cohérent qu'une fonction de covariance telle que l'exponentielle 2.2.13 favorise les fonctions lisses. En effet, si deux points t et t' sont proches dans $\mathbb{T} = \mathbb{R}$, les valeurs de $\tilde{s}(t)$ et $\tilde{s}(t')$ seront proches également du fait de leur forte corrélation, ce qui correspond précisément à une notion de continuité. Pour être plus précis, le choix de la fonction de covariance d'un processus gaussien conditionne la géométrie de presque toutes ses réalisations [20, 3].

Dans cette section, je ne pourrai pas décrire l'ensemble de la très riche littérature qui porte sur les différents types de fonctions de covariance possibles et sur leur connexion avec les objets mathématiques puissants que sont les espaces Hilbertiens à noyau reproduisant. Je me contenterai de décrire les résultats principaux et les exemples qui m'ont été le plus utiles dans mon travail. En particulier, je ne ferai qu'évoquer rapidement le cas des domaines de définition $\mathbb{T}$ qui ne sont pas des espaces euclidiens $\mathbb{T} \nsubseteq \mathbb{R}^D$. Pour plus de détails sur les fonctions de covariance, on peut se référer avec profit à [1, 140, 178, 20].

2.3.1 Caractérisation

Une fonction de covariance permet comme en 2.2.12 de construire la matrice de covariance des valeurs du processus en un ensemble quelconque T de positions de l'espace. Or, une matrice de covariance K est nécessairement définie positive, c'est-à-dire que toutes ses valeurs propres sont

des réels positifs (ou nuls). En effet, si tel n'était pas le cas, il serait possible de mettre en évidence un ensemble $T \in \mathbb{T}^L$ ainsi qu'une combinaison linéaire des éléments de $\tilde{\boldsymbol{s}}(T)$ qui a une variance négative, ce qui est impossible [10]. C'est cette contrainte d'avoir une variance positive pour toute combinaison linéaire des éléments de $\tilde{\boldsymbol{s}}(T)$ pour tout T qui impose à k de ne produire que des matrices de covariance définies positives, pour tout T.

Une fonction de covariance k doit générer une matrice de covariance K correcte pour tout $T \in \mathbb{T}^L$, c'est-à-dire dont les valeurs propres sont positives ou nulles. On dit que k doit être *définie positive*.

De telles fonctions existent et portent le nom de fonctions *définies positives*. Elles ont fait l'objet d'une attention importante de la part de mathématiciens dès les débuts du XX$^{\text{ème}}$ siècle au sujet de l'étude des espaces Hilbertiens à noyaux reproduisants [8, 9, 20]. Sans rentrer dans des précisions au-delà de la portée de cet exposé, on peut comprendre les RKHS comme des espaces de fonctions qui offrent la même souplesse que les espaces Euclidiens classiques, c'est-à-dire qu'ils sont équipés d'un produit scalaire et d'une norme. Par ailleurs, ces espaces sont définis par une propriété supplémentaire de *reproduction* qui permet d'exprimer la valeur prise par une des fonctions de l'espace comme dépendant de ses valeurs ailleurs. Cette propriété induit une certaine forme de régularité dans les fonctions étudiées. On montre alors qu'il y a bijection entre l'ensemble des fonctions définies positives et l'ensemble des RKHS [20]. On montre également qu'il y a une bijection entre les fonctions de covariance des processus gaussiens et les fonctions définies positives [20, 1, 178].

Ainsi, toute fonction définie positive sur $\mathbb{T} \times \mathbb{T}$ peut être utilisée comme une fonction de covariance valide pour un processus gaussien. Dans toute la suite de cette section, je vais considérer le cas spécial $\mathbb{T} \subset \mathbb{R}^D$ parce qu'il est utile dans les applications que j'ai considérées. Cependant, le formalisme étudié reste valide dans d'autres espaces où des fonctions définies positives existent, comme l'ensemble des arbres [124, 199], des chaines de caractères [139], etc.

2.3.2 Opérations sur les fonctions de covariance

La question centrale posée par le praticien dans le choix d'une fonction de covariance est l'interprétation de ses hyperparamètres et la géométrie qu'elle impose aux réalisations du processus gaussien. Ce sont en effet sur la base de ces critères que s'effectue son choix. Cependant, pour être valide, une fonction de covariance doit être définie positive et cette propriété peut sembler moins intuitive. Comment choisir une fonction de covariance expressive qui soit valide ?

Plusieurs propriétés des fonctions de covariance viennent apporter une solution à ce problème. Il est possible d'utiliser des fonctions de covariance connues et de les combiner pour obtenir des fonctions de covariance valides. En particulier, pour deux fonctions de covariance k_1 et k_2 de $\mathbb{T} \times \mathbb{T} \to \mathbb{C}$, on peut montrer [1, 184, 178] que leur somme

$$(t, t') \mapsto k_1(t, t') + k_2(t, t')$$

est une fonction de covariance. Cette propriété indique que la somme de processus gaussiens indépendants est elle-même un processus gaussien, de même que toute combinaison linéaire de processus gaussiens. De la même manière, leur produit

$$(t, t') \mapsto k_1(t, t')\, k_2(t, t'),$$

est aussi une fonction de covariance. Si $\mathbb{T} = \mathbb{T}_1 \times \mathbb{T}_2$ est le produit tensoriel de deux ensemble, comme $\mathbb{T} = \mathbb{R} \times \mathbb{N}_P$ évoqué en section 2.1, alors si k_1 et k_2 sont des fonctions de covariance sur $\mathbb{T}_1 \times \mathbb{T}_1$ et $\mathbb{T}_2 \times \mathbb{T}_2$ respectivement, alors

$$((t_1, t_2), (t_1', t_2')) \mapsto k_1(t_1, t_1') + k_2(t_2, t_2')$$

ainsi que

$$((t_1, t_2), (t_1', t_2')) \mapsto k_1(t_1, t_1')\, k_2(t_2, t_2') \tag{2.3.1}$$

10. Pour mettre en évidence une telle combinaison linéaire, il suffit d'utiliser la décomposition en valeurs propres de K.

sont des fonctions de covariance. On dit souvent de la fonction de covariance 2.3.1 qu'elle est *séparable* en k_1 et k_2.

Un autre procédé utilisé fréquemment pour obtenir de nouvelles fonctions de covariance est d'en utiliser des connues dans un espace de propriétés (*features* en anglais) $\mathbb{T}_f$ obtenu à partir de $\mathbb{T}$ par une fonction de *lien* $u : t \in \mathbb{T} \mapsto u(t) \in \mathbb{T}_f$. Ainsi, si k est une fonction de covariance valide dans $\mathbb{T}_f$,

$$(t, t') \in \mathbb{T} \mapsto k(u(t), u(t'))$$

est une fonction de covariance valide dans $\mathbb{T}$. Un tel procédé porte en anglais le nom de *warping*. Il permet d'effectuer la régression dans un espace de caractéristiques, au lieu de l'espace initial $\mathbb{T}$. On verra en section 2.3.4 un tel exemple de construction d'une fonction de covariance.

2.3.3 Covariance exponentielle carrée

Le cas particulier 2.2.13 de la covariance exponentielle carrée (EC) pour $\mathbb{T} \subset \mathbb{R}$ se généralise très bien pour $\mathbb{T} \subset \mathbb{R}^D$ en choisissant :

$$k_{EC}(t, t' \mid M, \sigma) = \sigma^2 \exp\left(-\frac{(t-t')^\top M (t-t')}{2}\right) \tag{2.3.2}$$

où σ^2 donne l'énergie du processus et où M est une matrice de dimension $D \times D$ semi-définie positive dont les vecteurs propres donnent les directions principales de covariance et dont les valeurs propres sont les inverses des longueurs caractéristiques le long de ces directions. On peut montrer que l'équation 2.3.2 définit une matrice de covariance valide pour tout D.

La fonction de covariance exponentielle carrée produit des réalisations infiniment dérivables et donc très lisses. Ses paramètres permettent de régler les échelles auxquelles se font les variations.

Cette formulation 2.3.2 permet de considérer que la fonction n'a pas un comportement identique dans toutes les directions de $\mathbb{T}$. Dans certaines applications, la longueur caractéristique peut très bien être assez grande dans une direction de $\mathbb{T}$, mais petite dans une autre. Par exemple, en météorologie, on ne traite pas les longitudes et latitudes de la même manière que l'élévation ou le temps. Cette propriété est répercutée par le choix d'une fonction de covariance k qui n'est pas *isotrope*, c'est-à-dire qui n'attribue pas la même valeur à tous les couples (t, t') séparés par la même distance $|t - t'|$. Cela se fait en choisissant les *directions caractéristiques* comme les vecteurs propres de M et les longueurs caractéristiques comme l'inverse des valeurs propres correspondantes. Si les vecteurs propres de M forment une base de $\mathbb{T}$ et que ses valeurs propres sont égales, on a une fonction de covariance isotrope.

On ne connaît pas nécessairement la valeur de M qui va mener à un meilleur modèle. Par un apprentissage en utilisant les techniques que j'évoquerai plus loin en section 2.4, on peut éliminer de M les directions de $\mathbb{T}$ dont les longueurs caractéristiques sont trop grandes, parce que ces directions n'influent que très peu sur les variations de la fonction étudiée. Cette approche porte en anglais le nom d'Automatic Relevance Determination (ARD, [178]) et a été introduite pour réduire la dimension du domaine de définition, ce qui revient au même que de faire de la sélection de caractéristiques, une activité fréquente en apprentissage automatique [94].

En figure 2.5 (a, c), j'ai représenté des tirages de processus gaussiens qui utilisent 2.3.2 comme fonction de covariance pour $\mathbb{T} = \mathbb{R}$ et $\mathbb{T} = \mathbb{R}^2$. Pour $D = 2$, j'ai choisi $M = \frac{1}{\lambda^2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. On voit que plus la longueur caractéristique λ est grande, plus les réalisations sont lisses, ce qui est cohérent avec son interprétation comme la distance nécessaire à parcourir dans $\mathbb{T}$ pour que la valeur de la fonction change significativement.

2.3.4 Covariance pseudo-périodique

La fonction de covariance périodique a été introduite dans [141] pour le cas $\mathbb{T} = \mathbb{R}$. Elle est obtenue en utilisant la technique de *warping* présentée plus haut en section 2.3.2 sur la fonction

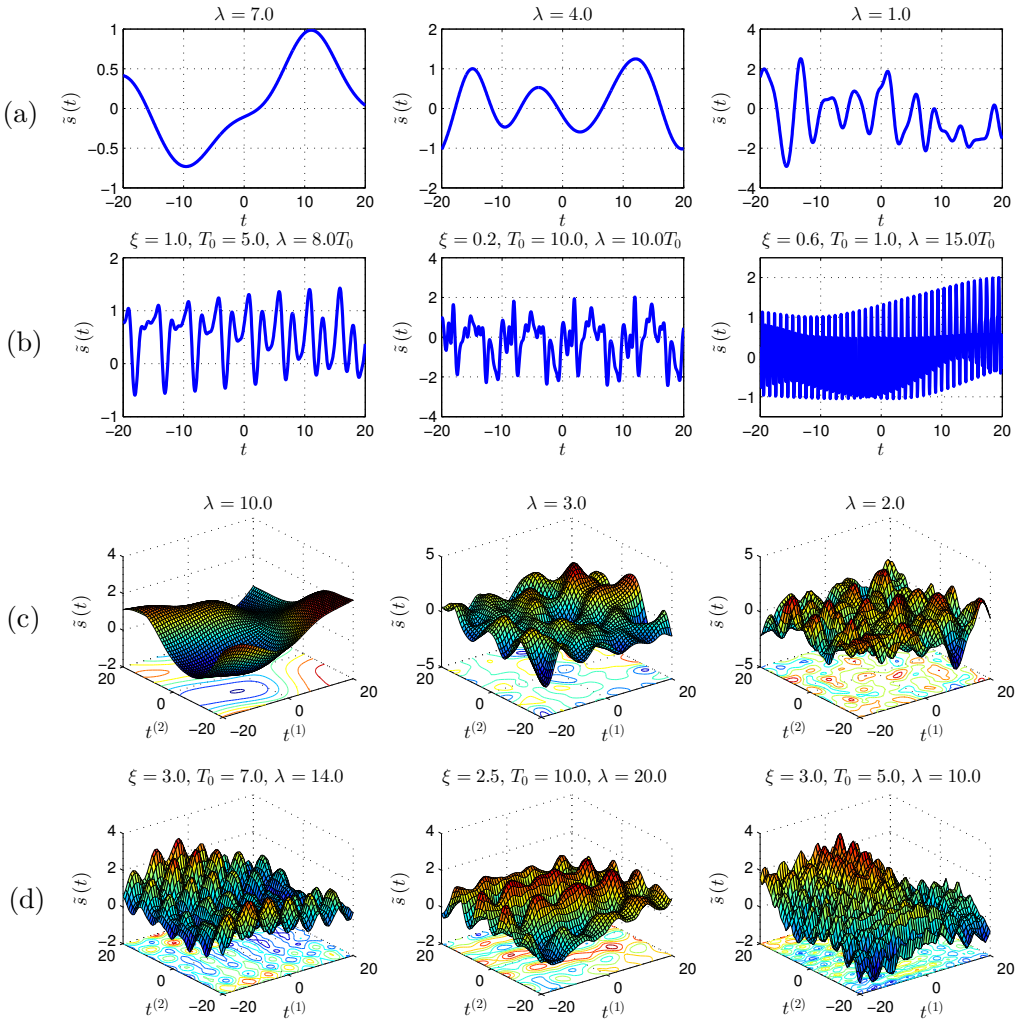

FIGURE 2.5: Tirages de processus gaussiens de moyenne nulle et de covariance exponentielle (a, c) ou pseudo-périodique (b, d) pour $\mathbb{T} = \mathbb{R}$ (a, b) et $\mathbb{T} = \mathbb{R}^2$ (c, d).

exponentielle carrée. La fonction de lien u considérée attribue à chaque instant de $\mathbb{T} = \mathbb{R}$ une position $u(t)$ sur le cercle unité, et donc dans $\mathbb{R}^2$, appelée *phase* :

$$\forall t \in \mathbb{R},\ u_{T_0}(t) = \begin{bmatrix} \sin \frac{2\pi t}{T_0} \\ \cos \frac{2\pi t}{T_0} \end{bmatrix}.$$

La fonction exponentielle carrée 2.3.2 est alors appliquée avec $M_\xi = \frac{1}{\xi^2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ pour donner :

$$\forall (t, t'),\ k_{per}(t, t' \mid \sigma, T_0, \xi) = k_{EC}(u_{T_0}(t), u_{T_0}(t') \mid M_\xi, \sigma), \tag{2.3.3}$$

dont on peut donner une forme plus directe :

$$k_{per1}(t, t' \mid \sigma, T_0, \xi) = \sigma^2 \exp\left(-\frac{2}{\xi^2} \sin^2\left(\frac{2\pi}{T_0}(t - t')\right)\right). \tag{2.3.4}$$

Dans le cas d'un signal périodique de période T_0, la fonction est corrélée à elle même toutes les périodes. On doit donc comparer t et t' *ramenés à la même période.*

Comme on le voit, cette fonction de covariance est périodique de période T_0, provoquant la même périodicité des réalisations du processus gaussien correspondant. Si l'hyperparamètre σ influe sur l'énergie du signal et T_0 sur la période du signal, le paramètre ξ est plus subtil : il indique la *stabilité* de l'onde au sein de chaque période. S'il est élevé, les différents échantillons se trouvant au sein d'une même période seront très corrélés. S'il est proche de 0, le motif qui se répètera sera très accidenté. Il influe ainsi sur la richesse spectrale des réalisations du processus.

Dans de nombreux cas, les signaux considérés ne sont pas strictement périodiques. En traitement du signal audio par exemple, on sait que les sons harmoniques sont seulement pseudo-périodiques, c'est-à-dire périodiques avec des modulations lentes de leurs composantes harmoniques. En termes de processus gaussien, on peut très simplement inclure cette connaissance dans le modèle en multipliant la fonction de covariance périodique 2.3.4 par une fonction exponentielle carrée de grande longueur caractéristique, imposant ainsi que deux valeurs du signal trop éloignées l'une de l'autre soient indépendantes, tout en permettant une corrélation périodique à court terme :

$$k_{pper}\left(t,t' \mid \sigma, T_0, \xi, \lambda\right) = k_{per}\left(t,t' \mid \sigma, T_0, \xi\right) k_{EC}\left(t,t' \mid 1, \lambda\right). \tag{2.3.5}$$

J'ai affiché cette fonction de covariance en fonction de $t-t'$ dans la figure 2.6 et des réalisations de processus gaussiens ayant cette fonction de covariance en figure 2.5 (b).

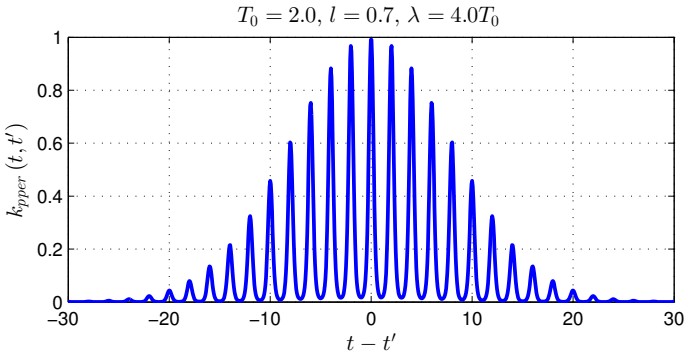

FIGURE 2.6: Fonction de covariance pseudo-périodique 2.3.4

Il est possible de généraliser la fonction de covariance pseudo-périodique 2.3.3 de nombreuses manières pour l'appliquer dans $\mathbb{T} = \mathbb{R}^D$. On peut à cette fin utiliser les différentes techniques vues en section 2.3.2. Pour ma part, j'ai considéré une généralisation simple en prenant :

$$k_{per}\left(t,t' \mid \sigma, T_0, \xi\right) = \prod_{d=1}^{D} \exp\left(-\frac{2}{\xi^2}\sin^2\left(\frac{2\pi}{T_0}\left(t_d - t'_d\right)\right)\right).$$

qui produit des réalisations dont la période est identique dans toutes les dimensions de $\mathbb{T}$. On peut multiplier cette fonction de covariance par la fonction EC pour modéliser des signaux pseudopériodiques définis sur $\mathbb{R}^D$. J'ai représenté des réalisations de processus ayant une telle fonction de covariance en figure 2.5 (d) pour $\mathbb{T} = \mathbb{R}^2$.

2.3.5 Autres types de fonctions de covariance

Toutes les fonctions de covariance que j'ai considérées jusqu'à présent sont *stationnaires*, c'est-à-dire qu'elles dépendent de $t - t'$. Ce n'est nullement une nécessité. Par exemple, la fonction qui à deux éléments t et t' de $\mathbb{T} = \mathbb{R}^D$, vecteurs de dimension $D \times 1$, associe

$$k\left(t,t'\right) \mapsto t^\top \Sigma t'$$

pour une matrice définie positive Σ de dimension $D \times D$ est une fonction de covariance, c'est un *produit scalaire* dans $\mathbb{T}$. Elle a été utilisée avec succès dans des problèmes de classification d'images

en utilisant des processus gaussiens. Dans ce cas, $\mathbb{T}$ est l'ensemble des images, et la fonction étudiée associe chaque image à une catégorie particulière [184, 178].

Un autre exemple célèbre de fonction de covariance non stationnaire est celle construite par Williams [219], qui établit un lien entre processus gaussien et réseaux de neurones [11]. On peut également généraliser la fonction de covariance EC pour qu'elle présente une longueur caractéristique qui varie en fonction de la position considérée [178].

2.4 Apprentissage des hyperparamètres

2.4.1 Motivations

Le choix de la fonction de covariance d'un processus gaussien est un des éléments fondamentaux dans la modélisation d'un signal. Cette fonction de covariance est en général adaptable au signal étudié par le biais d'un lot d'hyperparamètres. Comme je l'ai montré plus haut en section 2.3, la modification de ces hyperparamètres influe grandement sur l'allure des signaux modélisés. Ainsi, avec la même fonction de covariance, un signal qui pourra paraître probable avec un certain lot d'hyperparamètres deviendra très improbable avec un autre.

Dans cet exposé, je considèrerai toujours la fonction moyenne μ connue. Une fonction moyenne inconnue nécessite l'utilisation du cadre plus général des fonction aléatoires intrinsèques [32].

Un cas de figure très courant est celui où le choix de la fonction de covariance peut être justifié par certaines connaissances *a priori*, mais où la valeur exacte des hyperparamètres est inconnue. Ainsi, on peut par exemple savoir que le signal étudié manifeste une certaine régularité. Cette connaissance justifie le choix d'une fonction exponentielle carrée vue en section 2.3.3. Cependant, on peut ignorer les longueurs caractéristiques à utiliser.

Comme d'habitude, supposons qu'on observe la valeur $\tilde{\boldsymbol{s}}(T)$ de la fonction étudiée sur un ensemble $T = [t_1, \dots, t_L]$ de L points de l'espace $\mathbb{T}$. Supposons qu'on connaisse la fonction de covariance $k(t, t' \mid \theta)$ et qu'on souhaite estimer la valeur $\theta^\star$ qui soit la mieux adaptée. Ce problème porte le nom d'*estimation des hyperparamètres*. Plusieurs approches peuvent être suivies à cette fin. Dans cette étude, je me focaliserai sur une *estimation au maximum de vraisemblance*.

2.4.2 Maximisation de la vraisemblance

Les processus gaussiens offrent une manière efficace d'apprendre les hyperparamètres à partir de l'observation $\tilde{\boldsymbol{s}}(T)$. Comme on l'a vu en section 2.2.2, la probabilité de $\tilde{\boldsymbol{s}}(T)$ étant donnés μ, k et θ, aussi appelée *vraisemblance des observations*, est donnée par :

$$p(\tilde{\boldsymbol{s}}(T) \mid \mu, k, \theta) = \mathcal{N}(\tilde{\boldsymbol{s}}(T); \boldsymbol{\mu}(T), K_\theta), \tag{2.4.1}$$

où K_θ est la matrice de covariance, de dimension $L \times L$. Le principe de l'estimation au maximum de vraisemblance est de choisir θ qui maximise la vraisemblance des observations :

$$\theta^\star = \underset{\theta}{\operatorname{argmax}}\, p(\tilde{\boldsymbol{s}}(T) \mid \mu, k, \theta),$$

ce qui est équivalent à minimiser l'opposé de son logarithme népérien [12] :

$$\theta^\star = \underset{\theta}{\operatorname{argmin}}\, -\ln\left(p(\tilde{\boldsymbol{s}}(T) \mid \mu, k, \theta)\right). \tag{2.4.2}$$

Dans le cas gaussien et pour des processus à valeurs dans $\mathbb{R}$, l'équation 2.4.2 prend une forme particulière, obtenue par la simple application du logarithme à l'expression 2.2.2 de la probabilité des observations [13] :

$$\theta^\star = \underset{\theta}{\operatorname{argmin}}\, \frac{1}{2}\left(\tilde{\boldsymbol{s}}(T) - \boldsymbol{\mu}(T)\right)^H K_\theta \left(\tilde{\boldsymbol{s}}(T) - \boldsymbol{\mu}(T)\right) + \frac{1}{2}\ln|K_\theta| + \frac{L}{2}\ln 2\pi. \tag{2.4.3}$$

11. On montre qu'un réseau de neurones d'une couche cachée constituée d'une infinité de neurones est un processus gaussien.

12. Cette équivalence est due au fait que le logarithme est une fonction croissante.

13. Dans le cas de processus gaussiens à valeurs complexes, il suffit de remplacer $\frac{L}{2}\ln 2\pi$ par $L \ln \pi$ dans l'expression 2.4.3, ce qui ne change rien à l'estimation.

Comme on le voit, le problème d'estimation des hyperparamètres devient alors un problème standard d'optimisation, où l'objectif est d'estimer $\theta^\star$ qui minimise la *fonction de coût* 2.4.3. En fonction de la covariance et de l'hyperparamètre considéré, ce problème peut avoir une solution analytique ou pas. Il est fréquent d'utiliser pour la résolution de tels problèmes un algorithme de descente de gradient [21, 23] et de nombreux auteurs ont présenté de telles méthodes par le passé ainsi que diverses implémentations informatiques pour les fonctions de covariance les plus rencontrées.

En section 4, je montrerai comment des algorithmes récents de factorisation en tenseurs non négatifs peuvent être utilisés à cette fin, dans le cas particulier de signaux localement stationnaires et régulièrement échantillonnés.

2.5 Le cas stationnaire

Dans cette section, je considère le cas d'un domaine de définition discret de dimension finie D :

$$\mathbb{T} = \mathbb{Z}^D.$$

Ce cadre est très fréquent lorsqu'on considère des signaux régulièrement échantillonnés dans $\mathbb{R}^D$. Bien que le cas général de $\mathbb{T} = \mathbb{R}^D$ soit compatible avec les développements qui suivent, il est d'une inutile généralité pour les applications que je vais considérer.

Dans toute cette section, les réalisations des processus sont considérées sur un ensemble fini de positions, assimilable à une *grille régulière* :

$$T = \mathbb{N}_{L_1} \times \cdots \times \mathbb{N}_{L_D} \tag{2.5.1}$$

où L_d est le nombre de valeurs prises le long de la $d^{\text{ème}}$ dimension. Par exemple, les positions des différents pixels d'une image de dimension $L_1 \times L_2$ peuvent être comprises comme des éléments de $\mathbb{N}_{L_1} \times \mathbb{N}_{L_2}$.

Compte tenu du fait que toutes les réalisations sont envisagées sur le même ensemble fini T de points de $\mathbb{T}$, j'omettrai ici les indices correspondants de manière à alléger les notations. Ainsi, en l'absence d'ambiguïté sur ce point, les formes d'ondes $\tilde{\boldsymbol{s}}(T)$ étudiées seront notées $\tilde{\boldsymbol{s}}$.

2.5.1 Définition

Dans le cas où la fonction de covariance $k(t, t' \mid \theta)$ considérée est une fonction de la différence $t - t'$ entre ses arguments, on dit qu'elle est stationnaire. Dans ce cas, on note :

$$k(t, t' \mid \theta) = k(t - t' \mid \theta) \tag{2.5.2}$$

La plupart des fonctions de covariance que j'ai données en exemple en section 2.3 sont stationnaires. Lorsque la covariance considérée est fonction de la norme $\|t - t'\|$, on parle d'une fonction de covariance qui est en plus *isotrope*. C'est par exemple le cas de la fonction exponentielle carrée 2.3.2 si la matrice M est l'identité.

Intuitivement, une covariance stationnaire dépend de l'éloignement des points considérés, et non pas de leur position absolue. Les écarts à la moyenne d'un processus gaussien dont la fonction de covariance est stationnaire seront ainsi similaires quelle que soit l'origine choisie pour $\mathbb{T}$. Si en plus sa moyenne est partout identique, ses réalisations auront partout la même allure, puisque leur probabilité 2.2.12 devient indépendante de l'origine choisie pour $\mathbb{T}$. On dit alors que le processus est stationnaire au sens large (SSL).

Définition 2. Un processus gaussien est stationnaire au sens large si sa moyenne est constante et si sa fonction de covariance est stationnaire.

Dans le cas des processus gaussiens à valeur dans $\mathbb{R}$, qui correspond à la grande majorité des applications, il n'y a pas de différence entre les notions de stationnarité *stricte* et de stationnarité *au sens large*. On peut montrer qu'un processus gaussien $\tilde{s}$ à valeurs complexe sera strictement stationnaire si et seulement si sa moyenne est constante et si $\mathbb{E}\left[(\tilde{s}(t) - \mu)(\tilde{s}(t') - \mu)\right]$, en plus de sa covariance [14], est une fonction de $t - t'$. Dans la suite de cet exposé, la stationnarité sera toujours

14. Noter l'absence de conjugaison complexe.

entendue au sens large et je ne développerai donc pas ici la différence entre ces deux concepts.

2.5.2 Filtrage de Wiener ($D = 1$)

Avant d'introduire plus loin les résultats généraux sur les processus gaussiens stationnaires pour $D \geq 1$ quelconque, je propose d'en faire sentir l'intérêt dans le cas particulier des séries temporelles ($D = 1$). Pour ce faire, une approche élégante est d'utiliser les propriétés asymptotiques des matrices de Toeplitz [90] [15].

Ainsi, considérons pour l'instant le cas de séries temporelles ($\mathbb{T} = \mathbb{Z}$) et considérons le problème de la régression, posé plus haut en section 2.2.3. Je suppose alors que le processus $\tilde{x}$ observé est la somme d'un signal $\tilde{s}$ et d'un bruit additif $\tilde{\epsilon}$, tous deux indépendants et de moyenne nulle, et qu'on connaît les fonctions de covariance $k_{\tilde{s}}$ et $k_{\tilde{\epsilon}}$ de $\tilde{s}$ et $\tilde{\epsilon}$, supposées toutes deux stationnaires. On a donc :

Si les théorèmes de Bochner et de Wiener-Khinchin présentés plus loin suffisent à justifier les traitements fréquentiels appliqués aux processus gaussiens, je trouve l'approche algébrique intuitive et elle n'est pas fréquemment exposée.

$$\forall t \in \mathbb{T}, \tilde{x}(t) = \tilde{s}(t) + \tilde{\epsilon}(t),$$

et

$$\tilde{x} \sim \mathcal{PG}(0, k_{\tilde{s}} + k_{\tilde{\epsilon}}).$$

Supposons enfin qu'on observe $\tilde{x}$ sur un ensemble $T = \mathbb{N}_L$ de L points régulièrement espacés et qu'on souhaite estimer $\tilde{s}$ sur le même ensemble $T' = T$. L'estimateur $\widehat{\tilde{s}}$ de $\tilde{\boldsymbol{s}}$ est donc donné par $\boldsymbol{\mu}_{\text{post}}$ dans 2.2.16 page 28 :

$$\begin{aligned} \widehat{\tilde{s}} &= K(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{x}}) K(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}})^{-1} \tilde{\boldsymbol{x}} \\ &= K(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{s}}) (K(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{s}}) + K(\tilde{\boldsymbol{\epsilon}}, \tilde{\boldsymbol{\epsilon}}))^{-1} \tilde{\boldsymbol{x}} \end{aligned}$$

Du fait de la stationnarité de $k_{\tilde{s}}$ et de $k_{\tilde{\epsilon}}$, $K(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{s}})$ et $K(\tilde{\boldsymbol{\epsilon}}, \tilde{\boldsymbol{\epsilon}})$ sont des matrices de Toeplitz, c'est-à-dire qu'elles sont de la forme :

$$K = \begin{bmatrix} k(0) & k(-1) & k(-2) & \cdots & k(-(L-1)) \\ k(1) & k(0) & k(-1) & & k(-(L-2)) \\ \vdots & k(1) & k(0) & \ddots & \vdots \\ k(L-2) & & \ddots & \ddots & k(-1) \\ k(L-1) & k(L-2) & \cdots & k(1) & k(0) \end{bmatrix}.$$

On peut montrer que ces matrices bénéficient de certaines propriétés intéressantes [90]. En particulier, lorsque $L \to \infty$, elles deviennent équivalentes [16] à des matrices circulantes [17], c'est-à-dire dont chaque ligne est obtenue de la précédente par un décalage circulaire vers la droite :

$$C_L = \begin{bmatrix} k(0) & k(-1) & k(-2) & \cdots & k(-(L-1)) \\ k(-(L-1)) & k(0) & k(-1) & & k(-(L-2)) \\ \vdots & k(-(L-1)) & k(0) & \ddots & \vdots \\ k(-2) & & \ddots & \ddots & k(-1) \\ k(-1) & k(-2) & \cdots & k(-(L-1)) & k(0) \end{bmatrix}.$$

Ces matrices circulantes sont totalement déterminées par leur première ligne :

$$\boldsymbol{k}_L = [k(0), \ldots k(-(L-1))]^\top.$$

15. Je remercie Thomas Fillon pour m'avoir mis sur cette piste.

16. On dit que les séquences de matrices $\{K_L\}_L$ et $\{C_L\}_L$ sont équivalentes si $\lim_{L\to\infty} |K_L - C_L| = 0$, où $|\cdot|$ désigne une pseudo-norme matricielle.

17. K est également circulante si k est périodique et si L est un multiple de sa période. Pour une raison de clarté de l'exposé, j'omettrai de mentionner à chaque fois cette autre condition, qui joue cependant un rôle important dans les analyses dites *pitch-synchrones*.

Soit W_L la matrice de Fourier de dimension $L \times L$:

$$[W_L]_{l,l'} = \frac{1}{\sqrt{L}} \exp\left(-i2\pi \frac{(l-1)(l'-1)}{L}\right). \tag{2.5.3}$$

On montre que W_L forme une base orthonormée de $\mathbb{R}^L$, c'est-à-dire que :

$$W_L W_L^H = I_L, \tag{2.5.4}$$

où W_L^H désigne la transposée Hermitienne de W_L et où I_L est la matrice identité de dimensions $L \times L$.

Un résultat fondamental dans notre contexte est que toutes les matrices circulantes sont diagonalisables dans la base de Fourier [90]. Plus précisément, on a :

$$C_L = W_L^H \Lambda W_L, \tag{2.5.5}$$

où $\Lambda = \mathrm{diag}\,(W_L \boldsymbol{k}_L)$ est une matrice diagonale dont les éléments sont ceux de la transformée de Fourier discrète de $\boldsymbol{k}_L$. Par conséquent, pour L suffisamment grand, la matrice de covariance K d'un processus stationnaire devient :

$$K \approx W_L^H \mathrm{diag}\,(W_L \boldsymbol{k}_L)\, W_L$$

Considérons alors la transformée de Fourier discrète $\widehat{\boldsymbol{s}}$ de $\widehat{\tilde{\boldsymbol{s}}}$ qui se définit comme :

$$\widehat{\boldsymbol{s}} = \mathcal{F}_1\left\{\widehat{\tilde{\boldsymbol{s}}}\right\} = W_L \widehat{\tilde{\boldsymbol{s}}},$$

son expression devient [18] :

$$\begin{aligned} W_L \widehat{\tilde{\boldsymbol{s}}} &= W_L \left(K(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{s}}) \left(K(\tilde{\boldsymbol{s}}, \tilde{\boldsymbol{s}}) + K(\tilde{\boldsymbol{\epsilon}}, \tilde{\boldsymbol{\epsilon}})\right)^{-1} \tilde{\boldsymbol{x}}\right) \\ &\approx W_L W_L^H \mathrm{diag}\,(W_L \boldsymbol{k}_{\tilde{\boldsymbol{s}}})\, W_L \left(W_L^H \left(\mathrm{diag}\,(W_L \boldsymbol{k}_{\tilde{\boldsymbol{s}}}) + \mathrm{diag}\,(W_L \boldsymbol{k}_{\tilde{\boldsymbol{\epsilon}}})\right) W_L\right)^{-1} \tilde{\boldsymbol{x}} \\ &= \mathrm{diag}\,(W_L \boldsymbol{k}_{\tilde{\boldsymbol{s}}})\, W_L W_L^H \left(\mathrm{diag}\,(W_L \boldsymbol{k}_{\tilde{\boldsymbol{s}}}) + \mathrm{diag}\,(W_L \boldsymbol{k}_{\tilde{\boldsymbol{\epsilon}}})\right)^{-1} W_L \tilde{\boldsymbol{x}}, \end{aligned}$$

soit :

$$\widehat{\boldsymbol{s}} \approx \frac{\mathcal{F}_1\{\boldsymbol{k}_{\tilde{\boldsymbol{s}}}\}}{\mathcal{F}_1\{\boldsymbol{k}_{\tilde{\boldsymbol{s}}}\} + \mathcal{F}_1\{\boldsymbol{k}_{\tilde{\boldsymbol{\epsilon}}}\}} \cdot \boldsymbol{x} \tag{2.5.6}$$

où :

- $\boldsymbol{a} \cdot \boldsymbol{b}$ et $\frac{\boldsymbol{a}}{\boldsymbol{b}}$ représentent respectivement la multiplication et la division composante par composante de $\boldsymbol{a}$ et $\boldsymbol{b}$:

$$\begin{aligned} [\boldsymbol{a} \cdot \boldsymbol{b}]_k &= \boldsymbol{a}_k \boldsymbol{b}_k \\ \left[\frac{\boldsymbol{a}}{\boldsymbol{b}}\right]_k &= \frac{\boldsymbol{a}_k}{\boldsymbol{b}_k}. \end{aligned}$$

- $\boldsymbol{x}$ est la transformée discrète $W_L \tilde{\boldsymbol{x}}$ de $\tilde{\boldsymbol{x}}$

Autrement dit, pour L suffisamment grand, l'estimée aux moindres carrés de $\boldsymbol{s}$ s'obtient simplement composante par composante, ce qui implique que la complexité du calcul est en $\mathcal{O}(L)$, par contraste avec la complexité en $\mathcal{O}(L^3)$ requise par l'estimation en temporel 2.2.16. La forme d'onde $\widehat{\tilde{\boldsymbol{s}}}$ correspondante est obtenue par transformée de Fourier inverse $W_L^H \widehat{\boldsymbol{s}}$.

L'opération 2.5.6 porte le nom de filtrage de Wiener, en hommage à Norbert Wiener qui fut le premier à la présenter dans le cadre du traitement des séries temporelles ($D = 1$) dans les années 1940 [217]. Comme on le voit, il est d'une importance déterminante pour les applications pratiques des processus gaussiens, puisqu'il permet de filtrer les signaux sans avoir à procéder à la coûteuse inversion d'une matrice de covariance de rang plein.

18. où j'ai utilisé 2.5.4, 2.5.5 et le fait que $(AB)^{-1} = B^{-1}A^{-1}$ pour deux matrices A et B inversibles.

2.5.3 Représentation spectrale pour D quelconque

Dans le cas d'un domaine de définition de dimension $D \geq 1$ quelconque $\mathbb{T} = \mathbb{Z}^D$, les mêmes résultats peuvent être obtenus. Cependant, l'approche algébrique entreprise plus haut dans le cas $D = 1$ n'y est plus aussi intuitive.

Une meilleure manière d'aborder le problème dans ce cas est de directement considérer la transformée de Fourier $\boldsymbol{s} = \mathcal{F}_D(\tilde{\boldsymbol{s}})$ de la réalisation $\tilde{\boldsymbol{s}}$ d'un processus gaussien stationnaire [19] :

$$\boldsymbol{s}(f) = \mathcal{F}_D\{\tilde{\boldsymbol{s}}\}(f) = \sum_{t \in T} \tilde{\boldsymbol{s}}(t) \exp\left(-i2\pi f^\top t\right) \tag{2.5.7}$$

où $f^\top t$ est le produit scalaire des deux vecteurs f et t, tous deux de dimension $D \times 1$. On montre alors [20] que si la fonction de covariance k du processus gaussien $\tilde{s}$ considéré est stationnaire, on a :

Si $\tilde{s}$ est un processus gaussien de fonction de covariance k stationnaire, observé sur une grille

$$T = \mathbb{N}_{L_1} \times \cdots \times \mathbb{N}_{L_D},$$

alors, si tous les L_d sont suffisamment grands, on a :

$$\mathbb{E}\left[\left(\boldsymbol{s}(f) - \mathcal{F}_D\{\boldsymbol{\mu}\}(f)\right)\left(\boldsymbol{s}(f') - \mathcal{F}_D\{\boldsymbol{\mu}\}(f')\right)^\star\right] = \begin{cases} 0 & \text{si } f \neq f' \\ \mathcal{F}_D\{k\}(f) & \text{sinon} \end{cases}, \tag{2.5.8}$$

où $\mathcal{F}_D\{k\}$ désigne la transformée de Fourier de k. On a de plus :

$$\mathbb{E}\left[\left(\boldsymbol{s}(f) - \mathcal{F}_D\{\boldsymbol{\mu}\}(f)\right)^2\right] = 0.$$

Les composantes de $\boldsymbol{s}$ sont donc indépendantes et distribuées selon une loi circulaire gaussienne :

$$\begin{aligned} &\forall f,\ p\left(\boldsymbol{s}(f) \mid \boldsymbol{P} = \mathcal{F}_D(k)\right) \\ &\qquad = \frac{1}{\pi P(f)} \exp\left(-\frac{\left|\boldsymbol{s}(f) - \mathcal{F}_D\{\boldsymbol{\mu}\}(f)\right|^2}{P(f)}\right) = \mathcal{N}_c\left(\boldsymbol{s}(f) \mid \mathcal{F}_D\{\boldsymbol{\mu}\}(f), P(f)\right) \end{aligned} \tag{2.5.9}$$

Le fait que la matrice de covariance d'une série temporelle stationnaire se diagonalise dans la base de Fourier est un cas particulier de 2.5.9 pour $D = 1$. Comme on le verra en section 2.5.4, elle a en effet les mêmes conséquences en termes de simplification des calculs de régression, mais dans le cadre beaucoup plus général des processus définis sur $\mathbb{Z}^D$. Ces résultats se formalisent par le biais de deux théorèmes fondamentaux dans l'étude des processus gaussiens stationnaires.

Tout d'abord, le théorème de Wiener-Khinchin stipule que la Densité Spectrale de Puissance (DSP) $P(f)$ d'un processus gaussien stationnaire est la transformée de Fourier de sa fonction de covariance.

Ensuite, le théorème de Bochner stipule que les fonctions de covariance stationnaires sont caractérisées par une transformée de Fourier réelle et positive : une fonction $k(\tau \mid \theta)$ de $\mathbb{T}$ dans $\mathbb{C}$ est une fonction de covariance stationnaire $k(t - t' \mid \theta)$ si et seulement si elle peut être représentée par :

$$k(\tau) = \sum_{f \in \mathbb{T}} \exp(i2\pi f \cdot \tau) P(f) \tag{2.5.10}$$

où P est partout positive. Compte tenu du résultat du théorème de Wiener-Khintchin, ce résultat est cohérent puisqu'une densité spectrale de puissance ne se conçoit pas comme une quantité

19. Je rappelle que f désigne partout l'indice de fréquence dans une transformée discrète, et non pas une fréquence exprimée en Hertz. Si les processus considérés sont à valeurs réelles, je considère de plus dans tout le texte que seuls les indices correspondant à des fréquences positives sont conservés, puisque les valeurs de la transformée pour les fréquences négatives peuvent en être déduites de manière déterministe.

20. Je remercie ici Maurice Charbit et Roland Badeau d'avoir eu la gentillesse de m'aider à redémontrer ces résultats classiques.

négative ou complexe. On a :

Une conséquence importante du théorème de Bochner est que *toutes* les fonctions de covariance stationnaires dans $\mathbb{T} = \mathbb{Z}^D$ sont paramétrées par l'ensemble

$$\Theta = \left\{P \mid \forall f,\, P\left(f\right) \geq 0 \,\text{et}\, \mathcal{F}_D^{-1}\left\{P\right\} \,\text{existe}\right\}$$

des fonctions partout réelles et positives dans le domaine de Fourier. Chaque fonction P de la sorte peut alors être entendue comme la densité spectrale de puissance d'un processus gaussien.

2.5.4 Filtrage optimal de Wiener (D quelconque)

De manière à illustrer l'avantage pratique de manipuler des processus stationnaires, je vais considérer à nouveau le problème de débruitage d'une observation régulièrement échantillonnée, vu dans le cas $D = 1$ en section 2.5.2.

Supposons donc que le processus $\tilde{x}$ observé n'est pas le signal recherché $\tilde{s}$, mais sa somme avec un signal additif $\tilde{\epsilon}$, supposé indépendant de $\tilde{s}$:

$$\forall t \in \mathbb{T},\, \tilde{x}\left(t\right) = \tilde{s}\left(t\right) + \tilde{\epsilon}\left(t\right).$$

Si $\tilde{s}$ et $\tilde{\epsilon}$ sont tous les deux des processus gaussiens stationnaires de fonctions de covariance $k_{\tilde{s}}$ et $k_{\tilde{\epsilon}}$, $\tilde{x}$ est également un processus gaussien stationnaire de moyenne $\mu_{\tilde{x}} = \mu_{\tilde{s}} + \mu_{\tilde{\epsilon}}$ et de fonction de covariance $k_{\tilde{s}} + k_{\tilde{\epsilon}}$.

S'il est observé sur un ensemble de points T régulièrement espacés de suffisamment grande dimension, les résultats de la section précédente s'appliquent et tous les éléments du spectre $\boldsymbol{s}$ sont indépendants, de mêmes que les éléments de $\boldsymbol{x}$. Pour la suite, je noterai :

$$\begin{aligned} P_s\left(f\right) &= \mathcal{F}_D\left\{k_{\tilde{s}}\right\}\left(f\right) \\ P_\epsilon\left(f\right) &= \mathcal{F}_D\left\{k_{\tilde{\epsilon}}\right\}\left(f\right) \end{aligned}$$

les transformées de Fourier en dimension D des fonctions de covariance de $\tilde{s}$ et $\tilde{\epsilon}$ ainsi que

$$\begin{aligned} \boldsymbol{\mu}_s\left(f\right) &= \mathcal{F}_D\left\{\boldsymbol{\mu}_{\tilde{s}}\right\}\left(f\right) \\ \boldsymbol{\mu}_\epsilon\left(f\right) &= \mathcal{F}_D\left\{\boldsymbol{\mu}_{\tilde{\epsilon}}\right\}\left(f\right) \end{aligned}$$

les transformées de Fourier des vecteurs moyenne. Puisque la moyenne d'un processus stationnaire est constante, $\boldsymbol{\mu}_s$ et $\boldsymbol{\mu}_\epsilon$ sont identiquement nuls sauf en $f = 0$. La méthodologie classique consiste à tout d'abord écrire la distribution jointe de $\boldsymbol{s}\left(f\right)$ et $\boldsymbol{x}\left(f\right)$, pour un indice de fréquence f donné :

$$\begin{bmatrix} \boldsymbol{s}\left(f\right) \\ \boldsymbol{x}\left(f\right) \end{bmatrix} \sim \mathcal{N}_c\left(\begin{bmatrix} \boldsymbol{\mu}_s\left(f\right) \\ \boldsymbol{\mu}_s\left(f\right) + \boldsymbol{\mu}_\epsilon\left(f\right) \end{bmatrix}, \begin{bmatrix} P_s\left(f\right) & P_s\left(f\right) \\ P_s\left(f\right) & \left(P_s\left(f\right) + P_\epsilon\left(f\right)\right) \end{bmatrix}\right),$$

ce qui conduit facilement à la distribution *a posteriori* de $\boldsymbol{s}\left(f\right) \mid \boldsymbol{x}\left(f\right)$ grâce aux résultats 2.2.8 page 25 :

$$\boldsymbol{s}\left(f\right) \mid \boldsymbol{x}\left(f\right) \sim \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}}\left(f\right), \sigma^2_{\text{post}}\left(f\right)\right),$$

avec :

$$\begin{cases} \boldsymbol{\mu}_{\text{post}}\left(f\right) &= \boldsymbol{\mu}_s\left(f\right) + \frac{P_s(f)}{P_s(f)+P_\epsilon(f)}\left(\boldsymbol{x}\left(f\right) - \boldsymbol{\mu}_s\left(f\right) - \boldsymbol{\mu}_\epsilon\left(f\right)\right) \\ \sigma^2_{\text{post}}\left(f\right) &= \frac{P_s(f)P_\epsilon(f)}{P_s(f)+P_\epsilon(f)} \end{cases}. \tag{2.5.11}$$

Si les données sont centrées, c'est-à-dire si $\mu_{\tilde{s}}\left(t\right) = 0$ et $\mu_{\tilde{\epsilon}}\left(t\right) = 0$, les équations (2.5.11) se simplifient en :

$$\begin{cases} \boldsymbol{\mu}_{\text{post}}\left(f\right) &= \frac{P_s(f)}{P_s(f)+P_\epsilon(f)}\boldsymbol{x}\left(f\right) \\ \sigma^2_{\text{post}}\left(f\right) &= \frac{P_s(f)P_\epsilon(f)}{P_s(f)+P_\epsilon(f)} \end{cases}, \tag{2.5.12}$$

qui est la généralisation directe de 2.5.6 pour D quelconque.

Ainsi, la valeur estimée $\hat{\boldsymbol{s}}(f)$ de $\boldsymbol{s}(f)$ étant donnés $\boldsymbol{x}$ et $\theta = \{P_s, P_\epsilon, \mu_{\tilde{s}}, \mu_{\tilde{\epsilon}}\}$ qui minimise l'erreur quadratique moyenne [21] est obtenue en choisissant $\hat{\boldsymbol{s}}(f) = \boldsymbol{\mu}_{\text{post}}(f)$ dans 2.5.11. La forme d'onde estimée $\hat{\tilde{\boldsymbol{s}}}$ peut alors être récupérée par une transformation de Fourier inverse de $\hat{\boldsymbol{s}}$. Il est remarquable que ces résultats soient valides quelle que soit la dimension D du domaine de définition des signaux considérés.

Cette technique est toujours communément appelée filtrage (optimal) de Wiener. Bien qu'aucune difficulté majeure n'apparaisse lorsqu'on considère le cas plus général des signaux définis sur des espaces de dimension quelconque $D \geq 1$, il n'est pas si fréquent de trouver une telle présentation dans la littérature.

Les opérations 2.5.11 ne font intervenir que L opérations simples sur les transformées de Fourier des signaux, et non pas l'inversion coûteuse d'une matrice de covariance de dimension $L \times L$. Pour une image de dimension $N \times M$, la méthode proposée implique $L = NM$ opérations dans le domaine spectral, au lieu de l'inversion d'une matrice de taille $NM \times NM$.

Bien entendu, le filtrage de Wiener tel que je viens de l'exposer nécessite la connaissance des DSP P_s et P_ϵ des processus $\tilde{s}$ et $\tilde{\epsilon}$. C'est souvent sur leur estimation à partir de la seule observation de $\tilde{\boldsymbol{x}}$ que se situe la véritable difficulté d'un problème de débruitage. Cependant, il existe certains cas où ces DSP sont connues précisément. En particulier, le contexte de la séparation de sources informée, sur lequel je reviendrai en partie III, est particulièrement favorable sur ce point.

2.6 Conclusion

Dans ce chapitre, j'ai tout d'abord cherché à montrer que dans beaucoup de domaines du traitement du signal, les quantités étudiées peuvent être vues comme des fonctions scalaires, définies sur des espaces qui varient en fonction du domaine d'application. Ainsi, si une série temporelle sera définie sur $\mathbb{R}$, une grandeur spatiale sera définie sur $\mathbb{R}^2$, etc.

La plupart du temps, la nécessité d'un traitement vient du fait qu'on n'observe la fonction qu'en certains points, ou bien qu'on ne l'observe pas directement, mais une autre qui lui est liée. Dans ces conditions, l'objectif est souvent de déterminer la valeur de la fonction d'intérêt partout à partir des seules informations disponibles — on parle d'un problème de régression — ou bien d'extraire sa valeur des observations, auquel cas on parle souvent de débruitage, ou de lissage.

Dans le but d'aborder ces problématiques, il est indispensable de modéliser ce qu'on entend précisément par fonction. Une première approche dans ce but est d'établir un modèle paramétrique, comme un modèle linéaire, exponentiel, etc. Pour ce faire, il est nécessaire d'avoir sur la fonction une connaissance *a priori* suffisante, qui vient souvent de l'étude physique du problème considéré. En l'absence d'une telle connaissance, on court le risque d'établir un modèle inadéquat pour le phénomène étudié et d'ainsi obtenir des estimées mauvaises. Une deuxième approche, dite *non-paramétrique*, consiste à ne plus supposer de forme fixe à la fonction étudiée, mais plutôt de se concentrer sur des régularités locales comme sa continuité, sa dérivabilité, sa périodicité, etc.

Dans ce chapitre, j'ai présenté les processus gaussiens, qui sont un modèle très populaire pour l'étude non paramétrique de fonctions. En effet, ils bénéficient de plusieurs propriétés avantageuses. Tout d'abord, ils permettent de modéliser des fonctions scalaires définies sur n'importe quel domaine de définition. Ensuite, un processus gaussien est caractérisé par une fonction moyenne et une fonction de covariance, qu'il est souvent possible de choisir selon le problème considéré. Enfin, ils offrent une mécanique simple pour prendre en compte les observations et en tirer des conclusions sur la valeur de la fonction étudiée aux points d'intérêt. Dans le cas où la fonction étudiée est stationnaire, j'ai rappelé que les traitements dans le domaine fréquentiel permettent de grandement simplifier les calculs.

21. Puisque la distribution *a posteriori* est gaussienne, je rappelle que l'estimateur minimisant l'erreur quadratique moyenne se confond avec la moyenne. De plus, la transformée de Fourier est un opérateur unitaire, ce qui garantit qu'une estimée aux moindres carrés de $\boldsymbol{s}$ est équivalente à une estimée aux moindres carrés de $\tilde{\boldsymbol{s}}$.

Chapitre 3

Approximations et modèles structurés

Pour alléger les notations, je considérerai dans ce chapitre que la fonction moyenne de tous les processus gaussiens considérés est nulle. Comme on l'a vu, des fonctions moyennes non nulles (mais connues) peuvent être prises en compte aisément, au prix d'une légère complication des notations. Ce choix se justifie principalement par l'application des méthodes proposées au traitement des signaux audio, qui sont centrés la plupart du temps. Il est de plus extrêmement classique dans la littérature.

Dans mon travail de thèse, je ne me suis pas préoccupé du cas plus complexe où les fonctions moyennes sont inconnues. Il fait l'objet d'un champ de recherche plus vaste, qui occupe de nombreux travaux en géostatistiques [32, 44].

3.1 Approximations parcimonieuses

3.1.1 Processus gaussiens et complexité

Le maniement de processus gaussiens pour la régression, tel que je l'ai présenté en section 2.2.3 page 27, nécessite l'inversion de la matrice de covariance des observations pour procéder à l'inférence. S'il y a L observations, la matrice à inverser est donc de dimension $L \times L$, ce qui peut devenir prohibitif dans de nombreux cas. Par exemple, en traitement du signal audio, il est fréquent qu'un signal soit constitué de plusieurs millions d'échantillons, ce qui rend impossible l'utilisation des processus gaussiens telle que je l'ai présentée plus haut. Dans ce cas, il est nécessaire d'utiliser certaines techniques permettant de réduire considérablement la complexité des traitements pour pouvoir bénéficier des avantages de tels modèles.

Pour manipuler une observation de L points comme la réalisation d'un processus gaussien, il faut inverser une matrice de dimension $L \times L$, ce qui est beaucoup trop pour de nombreuses applications comme l'audio.

En effet, supposons qu'on étudie un processus gaussien $\tilde{s}$ dont on connaît la fonction de covariance k. Le processus observé n'est pas $\tilde{s}$, mais sa somme avec un bruit additif $\tilde{\epsilon}$, dont on connaît la fonction de covariance (généralement blanche $k_\epsilon(t,t') = \sigma^2 \delta_{tt'}$). Supposons que $\tilde{x}$ est observé pour L points $T = [t_1, \ldots, t_L]$ de $\mathbb{T}$. Cette observation est regroupée dans le vecteur $\tilde{\boldsymbol{x}}(T)$ de dimension $L \times 1$:

$$\tilde{\boldsymbol{x}}(T) = \tilde{\boldsymbol{s}}(T) + \tilde{\boldsymbol{\epsilon}}(T).$$

Admettons enfin que l'on souhaite obtenir la distribution *a posteriori* des valeurs de $\tilde{s}$ en un autre ensemble de L' points $T' = [t'_1, \ldots t'_{L'}]$ de $\mathbb{T}$. Comme souligné en section 2.2.3, cette distribution est donnée par :

$$\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{x}}(T) \sim \mathcal{N}\left(\tilde{\boldsymbol{\mu}}_{\text{post}}, K_{\text{post}}\right) \tag{3.1.1}$$

avec

$$\begin{cases} \boldsymbol{\mu}_{\text{post}} &= \boldsymbol{\mu}_{\tilde{s}}(T') + K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{x}}(T)) K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))^{-1} (\tilde{\boldsymbol{x}}(T) - \boldsymbol{\mu}_{\tilde{x}}(T)) \\ K_{\text{post}} &= K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{s}}(T')) - K(\tilde{\boldsymbol{s}}(T'), \tilde{\boldsymbol{x}}(T)) K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))^{-1} K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{s}}(T')) \end{cases}.$$

Si L est très grand, la principale difficulté de l'approche réside dans l'inversion de la matrice $K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))$ de dimension $L \times L$, requise pour le calcul de la moyenne et de la matrice de covariance de 3.1.1. Comme je l'ai déjà indiqué en section 2.5, le cas où la fonction de covariance considérée est stationnaire et où les observations sont régulièrement échantillonnées implique de considérables simplifications des calculs. En effet, l'inférence peut dans ce cas se faire dans le domaine fréquentiel et ne nécessite plus que $\mathcal{O}(L)$ opérations, auxquelles on doit rajouter les $\mathcal{O}(L \log L)$ opérations nécessaires à la transformation de Fourier et à son inverse. Cependant, adopter un modèle stationnaire n'est pas toujours adéquat. Il revient à considérer que la densité spectrale du signal est partout la même, ce qui n'est une bonne hypothèse que dans certains cas. En traitement du signal audio par exemple, il est très rare qu'on puisse considérer qu'un signal ait une DSP constante au cours du temps. En effet, le contenu spectral d'un enregistrement a toutes les raisons d'évoluer constamment. De plus, il est fréquent dans les applications que la fonction étudiée ne soit pas régulièrement échantillonnée sur son domaine de définition.

Dans le but de résoudre ce problème crucial de la complexité des traitements induits par l'utilisation de processus gaussiens, de nombreux auteurs ont proposé des techniques dont la plupart sont basées sur la notion de *parcimonie*. On peut distinguer les techniques basées sur l'introduction de *points support*, que je vais présenter rapidement en section 3.1.2, des techniques basées sur l'utilisation d'une fonction de covariance à support compact, que j'évoquerai en section 3.1.3.

3.1.2 Approches par points supports

De très nombreuses méthodes basées sur l'introduction de *points supports*[1] ont été suggérées pour permettre l'utilisation des processus gaussiens dans le cas d'une très grande quantité de données. Historiquement, elles furent proposées indépendamment les unes des autres, avant de bénéficier d'un travail d'unification effectué par Quiñonero-Candela et Rasmussen dans [176] qui a permis d'en identifier les points communs. Je me contenterai ici d'en présenter les principales idées ainsi que quelques variantes populaires, sans rentrer dans des détails qui seraient hors de la portée de ce texte. Cependant, le lecteur intéressé trouvera dans [178, 176] de nombreux pointeurs vers la littérature.

Principe

Pour comprendre les approches par points supports, on peut tout d'abord considérer un instant l'opération d'inférence standard 3.1.1. Le calcul de la distribution *a posteriori* $p(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(T))$ repose sur l'utilisation de la distribution jointe $p(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T'))$. C'est par conditionnement de cette distribution par rapport à la distribution marginale $p(\tilde{\boldsymbol{s}}(T))$ des observations que se fait l'inférence. Le réseau Bayésien correspondant est représenté en figure 3.1.

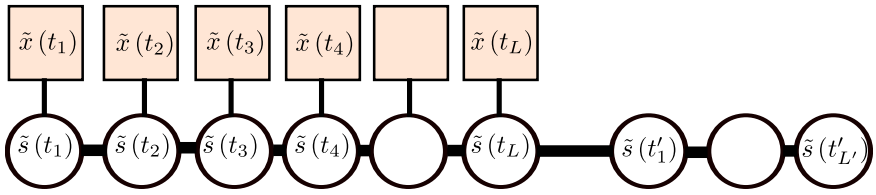

Figure 3.1: Réseau Bayésien correspondant à une inférence par processus gaussiens complète. Si on suppose une covariance blanche pour ϵ, les observations $\tilde{x}(t_l)$ ne sont liées qu'à la valeur latente du signal $\tilde{s}(t_l)$ à la même position. Les valeurs de $\tilde{s}$ sont quant à elles toutes reliées dans un réseau dense de relations, représenté par le trait plein horizontal.

Comme on le voit, l'ensemble des variables $\tilde{s}(t)$ sont reliées entre elles, conduisant à la nécessité d'inverser la matrice de covariance complète $K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))$ des observations. Une idée permettant

1. *Inducing inputs* en anglais

de simplifier grandement les calculs [198, 197, 176] consiste à introduire un groupe de variables aléatoires $\tilde{\boldsymbol{s}}(U)$, par le biais duquel transite toute l'information entre $\tilde{\boldsymbol{s}}(T)$ et $\tilde{\boldsymbol{s}}(T')$. Ces variables $\tilde{\boldsymbol{s}}(U)$ sont les valeurs prises par la fonction en un groupe U de L_U points fixés, appelés points supports.

En pratique, on introduit un ensemble $U = [u_1, \ldots, u_{L_U}]$ de $L_U \ll L$ points[2] de $\mathbb{T}$, appelés points support*s*, ainsi que le vecteur $\tilde{\boldsymbol{s}}(U)$ de dimension $L_U \times 1$ qui regroupe les valeurs du processus sur les points supports :

$$\tilde{s}(U) = \left[\tilde{s}(u_1), \ldots, \tilde{s}(u_{L_U})\right]^\top,$$

De manière à simplifier $p(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T'))$, on fait alors l'hypothèse que $\tilde{\boldsymbol{s}}(T)$ et $\tilde{\boldsymbol{s}}(T')$ sont conditionnellement indépendants étant donnés $\tilde{\boldsymbol{s}}(U)$, c'est-à-dire :

$$p(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U)) \approx p(\tilde{\boldsymbol{s}}(T) \mid \tilde{\boldsymbol{s}}(U))\, p(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U)). \tag{3.1.2}$$

Les approches à points supports introduisent un ensemble de points de $\mathbb{T}$ par lesquels transite l'information des données d'apprentissage $\tilde{\boldsymbol{s}}(T)$ vers les données de test $\tilde{\boldsymbol{s}}(T')$. Elles permettent ainsi de simplifier l'inférence.

Cette hypothèse n'est pas suffisante à elle seule pour éviter l'inversion de $K(\tilde{\boldsymbol{x}}(T), \tilde{\boldsymbol{x}}(T))$. Il est nécessaire pour cela d'introduire des approximations supplémentaires, qui consistent à remplacer $p(\tilde{\boldsymbol{s}}(T) \mid \tilde{\boldsymbol{s}}(U))$ et $p(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U))$ par des expressions $q(\tilde{\boldsymbol{s}}(T) \mid \tilde{\boldsymbol{s}}(U))$ et $q(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U))$ qui mènent à des simplifications de 3.1.1 :

$$\begin{aligned} p(\tilde{\boldsymbol{s}}(T) \mid \tilde{\boldsymbol{s}}(U)) &\approx q(\tilde{\boldsymbol{s}}(T) \mid \tilde{\boldsymbol{s}}(U)) \\ p(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U)) &\approx q(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U)) \end{aligned} \tag{3.1.3}$$

On peut montrer que pour certains choix des $q(\cdot \mid \tilde{\boldsymbol{s}}(U))$ dans 3.1.3, on obtient une distribution *a posteriori* beaucoup plus simple à calculer que 3.1.1. Sans préciser les détails des calculs, je vais à présent tâcher de présenter trois méthodes classiques de la littérature. Pour ce faire, je vais comparer le graphe de dépendance qu'elles introduisent à celui présenté en figure 3.1.

Fully Independent Conditional

L'approximation *Fully Independent Conditional*[3] (FIC) [198, 176] est représentée en figure 3.2. Elle consiste à supposer qu'étant donné $\tilde{\boldsymbol{s}}(U)$, tous les éléments de $\tilde{\boldsymbol{s}}(T)$ et de $\tilde{\boldsymbol{s}}(T')$ sont indépendants, c'est-à-dire à avoir :

$$q_{\text{FIC}}(\tilde{\boldsymbol{s}}(T) \mid \tilde{\boldsymbol{s}}(U)) = \prod_{l=1}^{L} p(\tilde{s}(t_l) \mid \tilde{\boldsymbol{s}}(U)) \tag{3.1.4}$$

$$q_{\text{FIC}}(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(U)) = \prod_{l'=1}^{L'} p(\tilde{s}(t'_{l'}) \mid \tilde{\boldsymbol{s}}(U)). \tag{3.1.5}$$

Intuitivement, l'approximation FIC permet de simplifier les calculs parce que tous les éléments de $\tilde{\boldsymbol{s}}(T)$ et $\tilde{\boldsymbol{s}}(T')$ sont supposés indépendants. La matrice de covariance à inverser pour procéder à l'inférence devient diagonale dans ce cas, ce qui permet de réduire la complexité de $\mathcal{O}(L^3)$ à $\mathcal{O}(LU^2)$.

Cependant, comme démontré dans [197], cette approximation ne donne de bons résultats que si les points supports sont suffisamment denses dans $\mathbb{T}$. En effet, si une position de test $t' \in T'$ est trop loin des points supports, l'estimée qui y sera faite sera de mauvaise qualité, même si de nombreux points observés de T sont très proches de t'.

2. Il faut noter que les points supports ne sont pas nécessairement un sous ensemble des points observés T. La question du choix de ces points, que je n'aborde pas ici, fait l'objet de recherches spécifiques [178].

3. Compte tenu de l'ancrage des termes en anglais dans l'usage, je n'ai pas trouvé bon de traduire le nom de ces méthodes en français.

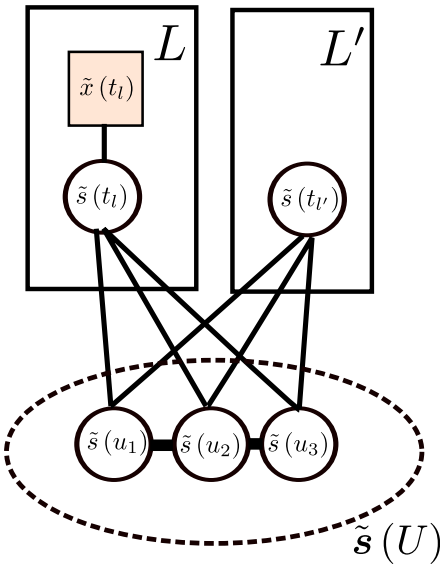

FIGURE 3.2: Réseau Bayésien correspondant à l'approximation FIC. Si on suppose une covariance blanche pour ϵ, les observations $\tilde{x}(t_l)$ ne sont toujours liées qu'à la valeur correspondante du signal $\tilde{s}(t_l)$, mais contrairement au modèle complet, aucune des valeurs de $\tilde{\boldsymbol{s}}(T)$ et de $\tilde{\boldsymbol{s}}(T')$ ne sont reliées entre elles directement : elles sont toutes supposées indépendantes étant donné $\tilde{\boldsymbol{s}}(U)$. Par contre, tous les éléments de $\tilde{\boldsymbol{s}}(U)$ sont reliés entre eux. Les rectangles, correspondant à la notation en assiette (*plate*), indiquent que les motifs encadrés sont répétés L et L' fois.

Partial Independent Training Conditional

Pour tâcher de palier à cet inconvénient, l'approximation *Partial Independent Training Conditional* (PITC) a été proposée dans [197], qui vise à atténuer la forte hypothèse 3.1.4 d'indépendance de tous les éléments de $\tilde{\boldsymbol{s}}(T)$ étant donné $\tilde{\boldsymbol{s}}(U)$. Pour ce faire, l'idée consiste à regrouper les éléments de $\tilde{\boldsymbol{s}}(T)$ en B blocs au sein desquels toutes les interdépendances sont conservées. Les blocs sont construits de telle manière à regrouper des points proches dans $\mathbb{T}$. Le réseau Bayésien correspondant est représenté en figure 3.3.

La justification de ce modèle est présentée par SNELSON comme un moyen d'introduire un compromis entre une simplification globale telle que FIC et une approche purement locale qui consisterait à n'utiliser pour la régression en un point donné $t' \in \mathbb{T}$ que le bloc le plus proche de t'. On peut en effet voir PITC comme un compromis entre ces deux approches, où une certaine partie de l'information globale est conservée par le biais des points supports, mais où une information locale reste manifeste au sein des différents blocs. La distribution $p(\tilde{\boldsymbol{s}}(T') \mid \tilde{\boldsymbol{s}}(T))$ correspondante repose sur l'inversion d'une matrice qui n'est plus diagonale comme dans FIC, mais qui est bloc-diagonale. Une telle matrice s'inverse très facilement, et la complexité de la méthode est de $O\left(LN_B^3\right)$ où N_B est le nombre de points dans chaque bloc.

SNELSON constate cependant [197] que PITC ne se démarque que très peu en termes de performances de FIC et ne permet pas réellement d'outrepasser les limitations dues à l'introduction des points supports par lesquels transite toute l'information entre l'observation $\tilde{\boldsymbol{s}}(T)$ et les valeurs recherchées $\tilde{\boldsymbol{s}}(T')$. En effet, l'hypothèse d'indépendance conditionnelle force la moyenne *a posteriori* à être une somme pondérée de L_U contributions centrées sur les points supports, ce qui empêche une modélisation correcte dès que la régression concerne des points plus éloignés. C'est donc le principe même de supposer $\tilde{\boldsymbol{s}}(T)$ et $\tilde{\boldsymbol{s}}(T')$ conditionnellement indépendants étant donnés $\tilde{\boldsymbol{s}}(U)$ qui est responsable de la limitation des performances.

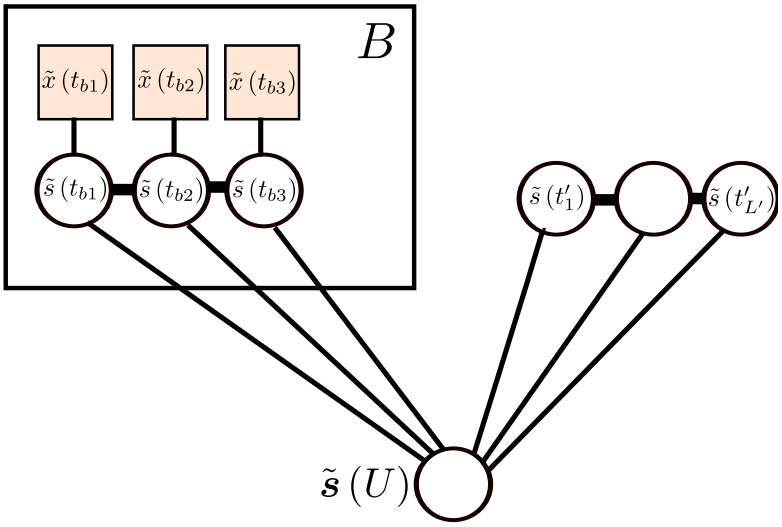

FIGURE 3.3: Réseau Bayésien correspondant à l'approximation PITC. Contrairement à FIC, les observations ne sont plus toutes indépendantes étant donné $\tilde{\boldsymbol{s}}(U)$. Au contraire, elles sont découpées en B blocs au sein desquels tous les liens sont conservés, ainsi que ceux avec les variables de $\tilde{\boldsymbol{s}}(U)$.

Partially Independent Conditional

De manière à remédier aux limitations introduites par l'hypothèse d'indépendance conditionnelle 3.1.2, SNELSON la remplace par un autre type d'indépendance. Au lieu de séparer conceptuellement $\tilde{\boldsymbol{s}}(T)$ et $\tilde{\boldsymbol{s}}(T')$, l'approche *Partially Independent Conditional* (PIC) consiste à d'abord regrouper l'ensemble des points observés T et recherchés T' en un ensemble $T_{\text{tot}} = T \cup T'$, puis à découper T_{tot} en B blocs, supposés conditionnellement indépendants étant donnés les points supports.

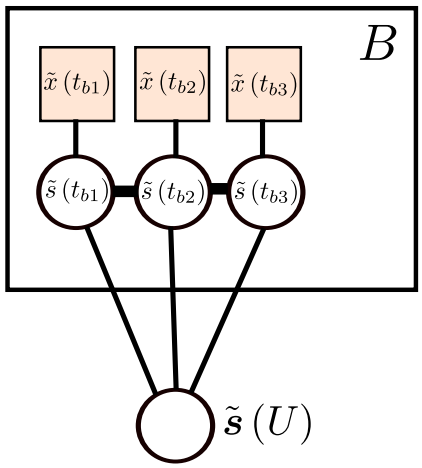

FIGURE 3.4: Réseau Bayésien correspondant à l'approximation PIC. On ne fait plus comme dans FIC ou PITC de distinction entre T et T'. L'ensemble des points sont regroupés en $T_{\text{tot}} = T \cup T'$, et c'est T_{tot} qui est découpé en blocs indépendants étant donné $\tilde{\boldsymbol{s}}(U)$. Au sein de chaque bloc, toutes les dépendances sont conservées. Il faut donc comprendre $t_{bi} \in T_{\text{tot}}$ et les $\tilde{x}(t_{bi})$ ne sont observés que pour $t_{bi} \in T$.

L'approximation PIC sort donc du cadre proposé par QUIÑONERO-CANDELA et RASMUSSEN [176] dans le sens où si les points supports sont conservés, ils ne servent plus à séparer T de T'. Cependant, ses avantages sont multiples. Pour un point $t' \in T'$, la régression se fait en utilisant

conjointement tous les points observés qui sont dans le même bloc que t', mais aussi l'ensemble des autres observations, qui n'interviennent que par le biais des points supports. De cette manière, PIC fournit un réel compromis entre une régression complètement locale et l'utilisation de l'ensemble des données d'apprentissage $\tilde{\boldsymbol{s}}(T)$. Sa complexité calculatoire est du même ordre de grandeur que PITC.

PIC a cependant, tout comme PITC, l'inconvénient de ne pas garantir une estimation lisse, même si la fonction de covariance considérée k implique des réalisations lisses pour le processus gaussien. Ce fait peut n'avoir qu'une importance très secondaire dans beaucoup d'applications, où l'essentiel est de parvenir à de bonnes estimations. Cependant, dans d'autres applications, l'introduction de discontinuités dans les estimées est rédhibitoire. Dans le cas du traitement du signal audio qui va nous intéresser principalement, la moindre discontinuité s'entend comme un bruit désagréable et il est nécessaire de considérer d'autres techniques d'approximation.

3.1.3 Fonctions de covariance à support compact

Une direction de recherche indépendante des méthodes à points supports [206, 145] repose sur l'utilisation de processus gaussiens munis de fonctions de covariance à support compact.

Si $\mathbb{T}$ est muni d'une norme $\|\cdot\|$, on dit qu'une fonction de covariance $k : \mathbb{T} \times \mathbb{T} \to \mathbb{R}$ est à support compact [85, 145] si k s'annule lorsque ses deux arguments sont suffisamment loin l'un de l'autre. Formellement, k est à support compact s'il existe $E \in \mathbb{R}_+$ tel que :

$$\|t - t'\| > E \Rightarrow k(t, t') = 0.$$

L'utilisation d'une fonction de covariance à support compact permet d'obtenir une matrice de covariance $K(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T))$ creuse, pour laquelle il existe des procédures d'inversion rapides.

La particularité des fonctions de covariance à support compact est qu'elles conduisent à des matrices de covariance $K(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T))$ *creuses*, c'est-à-dire dont peu d'entrées sont non nulles. L'intérêt de l'approche réside dans le fait qu'on dispose d'algorithmes rapides pour l'inversion de matrices creuses [150]. Ainsi, les méthodes basées sur l'utilisation de fonctions de covariance à support compact ne procèdent pas à des *approximations* de la procédure d'inférence comme celles évoquées plus haut, mais donnent lieu à une inférence à la fois exacte et rapide.

Dans ces conditions, l'enjeu de telles approches est la mise au point de fonctions de covariance à support compact. En effet, tronquer une fonction de covariance valide pour qu'elle soit nulle si $\|t - t'\|$ est suffisamment grand ne conduit pas nécessairement à l'obtention d'une fonction définie positive, ce qui est pourtant nécessaire pour une fonction de covariance.

Dans [206], Vanhatalo et Vehtari montrent par exemple que si $\mathbb{T} = \mathbb{R}^D$, la fonction

$$k(t, t' \mid \sigma_0, \Omega) = \begin{cases} \sigma_0^2 \left[\frac{2+\cos(2\pi r)}{3}(1-r) + \frac{1}{2\pi}\sin(2\pi r)\right] & \text{si } r = \sqrt{(t-t')^\top \Omega (t-t')} < 1 \\ 0 & \text{sinon} \end{cases}$$

est une fonction de covariance à support compact et peut donc être utilisée pour la régression par processus gaussiens. Dans cette expression, Ω est une matrice définie positive de dimension $D \times D$. Cette fonction induit des réalisations presque partout continues et dérivables. Comme je l'ai évoqué en section 2.3, elle peut être combinée avec d'autres fonctions pour modéliser un *a priori* particulier sur le problème traité, où une dépendance à support borné se combine par exemple avec une périodicité. D'autres auteurs ont introduit des fonctions de covariance à support compact dont on trouvera un inventaire dans [85].

Dans cette section, j'ai présenté les différentes approches explorées dans la littérature pour accélérer la procédure d'inférence dans des modèles de processus gaussiens. Malgré leur efficacité significative, ces modèles souffrent en effet d'une complexité algorithmique prohibitive lorsque le nombre L d'observations devient important.
Pour résoudre ce problème, certains auteurs ont proposé des approximations parcimonieuses reposant sur l'introduction de points supports, tandis que d'autres ont montré que l'utilisation d'un certain type de fonctions de covariance permet de grandement simplifier les calculs. Bien entendu, ces deux approches peuvent être combinées, comme dans [206], pour bénéficier de leurs avantages respectifs.

3.2 Tramage

3.2.1 Cadre de travail

Lors de mon travail, j'ai été amené à faire le rapprochement entre le problème de la régression par processus gaussiens et celui de la séparation de sources. Je présenterai plus avant ces rapprochements en partie II. L'essentiel pour l'heure est d'indiquer que les méthodes de l'état de l'art que j'ai cherché à généraliser ont toutes la particularité d'effectuer la séparation dans le domaine Temps-Fréquences (TF) [17, 30]. En pratique, les signaux sont découpés en trames, qui en sont des sections de petite longueur (généralement un millier d'échantillons) et la séparation est effectuée par un filtrage de Wiener [4] différent dans chaque trame.

Il m'est apparu que ce procédé de tramage offre une similarité frappante avec les approximations PITC et PIC présentées plus haut. En effet, il revient à effectuer un découpage par bloc des données. Cependant, ces méthodes ne sont pas équivalentes, parce que le tramage est souvent fait avec un recouvrement entre les trames successives, alors que PITC et PIC considèrent des blocs distincts. Dans ces conditions, je me suis interrogé sur la pertinence d'étendre ce procédé de tramage à un domaine de définition $\mathbb{T} = \mathbb{R}^D$ plus large, et de le voir comme une méthode d'approximation pour l'utilisation de processus gaussiens.

Il s'avère que le tramage bénéficie de propriétés intéressantes, en particulier la production d'estimées lisses, au prix d'une sous-estimation de la variance *a posteriori* des résultats. Ce travail s'insère dans un cadre de recherche prometteur, ouvert par certains travaux récents sur la séparation de sources [127, 11] qui tentent de prendre en compte les dépendances déterministes entre les échantillons de trames successives.

3.2.2 Opérateur de tramage ($\mathbb{T} = \mathbb{Z}$)

Si on considère un instant le cas d'une série temporelle $\tilde{s}$ régulièrement échantillonnée ($\mathbb{T} = \mathbb{Z}$), les traitements sont la plupart du temps précédés d'une opération de *tramage*, dite aussi de *fenêtrage*, qui consiste à extraire des portions successives du signal, de durée identique $L_0 \ll L$ et qui ont entre elles un certain *recouvrement* [5] $\rho \in [0\,1[$. Soit

$$\mathbb{T}_0 = [0, \dots (L_0 - 1)]\,,$$

le tramage $\mathcal{G}\{\tilde{s}\}$ de $\tilde{s}$ peut se comprendre comme un signal, défini sur $\mathbb{T}_0 \times \mathbb{Z}$ et donné par :

$$\forall n \in \mathbb{Z},\ \mathcal{G}\{\tilde{s}\}(\cdot, n) = \boldsymbol{g} \cdot [\tilde{s}(t)]_{t \in \mathcal{T}_G(n)} \tag{3.2.1}$$

où :

- $\mathcal{T}_{\mathcal{G}}(n) = \{t \in \mathbb{T} \mid t - nL_0(1-\rho) \in \mathbb{T}_0\}$ est l'ensemble des échantillons de $\mathbb{T}$ apparaissant dans la trame n.
- $\boldsymbol{g}$ est un vecteur de dimension $L_0 \times 1$, appelé *fenêtre de pondération*, par lequel sont multipliés les éléments de chaque trame. Je supposerai pour des raisons de simplicité qu'aucun élément de $\boldsymbol{g}$ n'est nul.

4. voir section 2.5 page 35

5. Les trames ne se recouvrent pas du tout si $\rho = 0$ et se recouvrent presque totalement si ρ est proche de 1. Les choix classiques sont $\rho = 0.5$ ou $\rho = 0.75$. Je considérerai le cas général, en supposant cependant toujours que $(1-\rho)\,L_0$ est entier.

Si un signal $\tilde{s}$ est observé sur un ensemble $T = \mathbb{N}_L$ de points, son tramage $\mathcal{G}\{\tilde{s}\}$ est observé sur

$$T_{\mathcal{G}} = [0, \ldots, (L_0 - 1)] \times [0, \ldots, (N - 1)],$$

où N est le nombre de trames correspondant à T [6]. Ainsi, le tramage d'une réalisation régulièrement échantillonnée $\tilde{\boldsymbol{s}}(T)$ de $\tilde{s}$ est une matrice $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$, de dimension $L_0 \times N$.

Compte tenu des redondances introduites par le recouvrement des trames successives, le même échantillon est représenté dans plusieurs trames. Par conséquent, une matrice quelconque $\hat{S}$ de dimension $L_0 \times N$ n'est pas nécessairement le tramage d'un signal observé $\tilde{\boldsymbol{s}}(T)$.

Cependant, il est important pour les applications de pouvoir reconstruire le signal $\widehat{\tilde{\boldsymbol{s}}(T)}$ à partir d'un tramage modifié $\hat{S}$. En effet, après avoir modifié le tramage $\mathcal{G}\{\tilde{\boldsymbol{s}}_0(T)\}$ d'un signal $\tilde{\boldsymbol{s}}_0(T)$ pour aboutir à $\hat{S}$, on veut pouvoir produire un signal résultat $\widehat{\tilde{\boldsymbol{s}}(T)}$ dans T. En d'autres termes, il est important pour les applications de disposer d'une inversion du tramage. Soit $\hat{S}$ une matrice de dimension $L_0 \times N$. Dans un article célèbre [92], Griffin et Lim posent le problème comme celui de trouver

La redondance des échantillons lorsqu'il y a recouvrement implique que toute matrice $L_0 \times N$ n'est pas nécessairement le tramage d'un signal. Ce n'est en effet pas le cas dès que les contraintes induites par le recouvrement ne sont pas respectées.

$$\widehat{\tilde{\boldsymbol{s}}(T)} = \underset{\tilde{\boldsymbol{s}}(T)}{\operatorname{argmin}} \sum_{t,n} \left(\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}(t,n) - \hat{S}(t,n) \right)^2 \tag{3.2.2}$$

et montrent que le signal recherché est donné par :

$$\forall t \in T,\ \mathcal{G}^{-1}\left\{\hat{S}\right\}(t) = \frac{1}{\sum_{n \in \mathcal{N}_{\mathcal{G}}(t)} \boldsymbol{g}(t - nL_0(1-\rho))^2} \times \left(\sum_{n \in \mathcal{N}_{\mathcal{G}}(t)} \hat{S}(t - nL_0(1-\rho), n)\, \boldsymbol{g}(t - nL_0(1-\rho)) \right) \tag{3.2.3}$$

où

$$\mathcal{N}_{\mathcal{G}}(t) = \{n \in \mathbb{Z} \mid t - nL_0(1-\rho) \in \mathbb{T}_0\} \tag{3.2.4}$$

est l'ensemble des trames dans lesquelles l'échantillon $t \in \mathbb{T}$ apparaît dans le fenêtrage $\mathcal{G}$. Cette opération porte souvent le nom d'*addition-recouvrement* et on peut aisément constater que $\mathcal{G}^{-1} \circ \mathcal{G}$ est bien l'identité.

Même si l'exposé de ces résultats classiques peut aujourd'hui paraître superflu, ils prendront une importance significative lorsque j'introduirai l'hypothèse locale en section 3.2.4.

3.2.3 Opérateur de tramage ($\mathbb{T} = \mathbb{R}^D$)

Une contribution mineure de mon travail a été d'introduire une généralisation du tramage classique des signaux temporels au cas des signaux définis sur $\mathbb{T} = \mathbb{R}^D$. L'intérêt de cette contribution est de permettre d'appliquer aux signaux multidimensionnels les mêmes simplifications calculatoires qu'aux signaux audio. Bien entendu, ces simplifications se font au prix de certains inconvénients, sur lesquels je reviendrai plus loin.

L'introduction du tramage pour $\mathbb{T} = \mathbb{R}^D$ nécessite la généralisation des notations présentées plus haut pour $\mathbb{T} = \mathbb{Z}$.

– De la même manière que pour $\mathbb{T} = \mathbb{Z}$, où chaque trame est un signal de longueur L_0, chaque trame pour $\mathbb{T} = \mathbb{R}^D$ sera définie sur un *domaine de trame* :

$$\mathbb{T}_0 = [0, \ldots, L_{0,1}] \times \cdots \times [0, \ldots, L_{0,D}] \tag{3.2.5}$$

où $L_{0,d}$ donne la longueur de $\mathbb{T}_0$ le long de la $d^{\text{ème}}$ dimension. Soit

$$\boldsymbol{L}_0 = [L_{0,1}, \ldots, L_{0,D}]^\top.$$

6. Pour des raisons de simplicité, je supposerai que L permet d'observer un nombre entier de trames. Un bourrage par des 0 permet de le garantir si $\tilde{\boldsymbol{s}}(T)$ est trop court.

– Les recouvrements $\boldsymbol{\rho}_1, \ldots, \boldsymbol{\rho}_D$ des trames le long de chacune des dimensions d de $\mathbb{T}$ seront regroupés dans un vecteur $\boldsymbol{\rho}$, de dimension $D \times 1$

$$\boldsymbol{\rho} \in [0\,1[^D .$$

– Une trame sera identifiée par son index $\boldsymbol{n}$, qui est un élément de $\mathbb{Z}^D$:

$$\boldsymbol{n} = (\boldsymbol{n}_1, \ldots, \boldsymbol{n}_D) \in \mathbb{Z}^D.$$

– L'ensemble des trames dans lesquelles apparaît l'échantillon $t \in \mathbb{T}$ sera noté :

$$\mathcal{N}_{\mathcal{G}}(t) = \left\{\boldsymbol{n} \in \mathbb{Z}^D \mid (t - \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho})) \in \mathbb{T}_0\right\}$$

et on notera

$$\mathcal{T}_{\mathcal{G}}(\boldsymbol{n}) = \{t \in \mathbb{T} \mid (t - \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho})) \in \mathbb{T}_0\}$$

l'ensemble des positions de $\mathbb{T}$ apparaissant dans la trame $\boldsymbol{n}$.

– L'ensemble des trames correspondant à un ensemble de points T sera noté

$$\mathcal{N}_{\mathcal{G}} = \cup_{t \in T} \mathcal{N}_{\mathcal{G}}(t) .$$

– La fenêtre de pondération g est une fonction de $\mathbb{T}_0$ dans $\mathbb{R}^\star$.

Le tramage d'un signal $\tilde{s}$ de $\mathbb{T} \to \mathbb{C}$ sera alors un signal $\mathcal{G}\{\tilde{s}\}$ de $\mathbb{T}_0 \times \mathbb{Z}^D \to \mathbb{C}$, donné par :

$$\forall (t, \boldsymbol{n}) \in \mathbb{T}_0 \times \mathbb{Z}^D,\ \mathcal{G}\{\tilde{s}\}(t, \boldsymbol{n}) = g(t)\, \tilde{s}(t + \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho})) . \tag{3.2.6}$$

Comme on le voit, l'expression 3.2.6 est la généralisation directe de 3.2.1. En effet, les notations ne doivent pas éclipser la simplicité du procédé, qui consiste simplement à découper le signal en trames de même taille, qui ont entre elles un certain recouvrement, et à multiplier chacune par la fenêtre de pondération.

Si on observe un signal $\tilde{s}$ sur L positions $T = [t_1, \ldots, t_L]$ de manière à avoir une observation $\tilde{\boldsymbol{s}}(T)$ de dimension $L \times 1$, $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$ est obtenu comme l'observation de $\mathcal{G}\{\tilde{s}\}$ sur :

$$T_{\mathcal{G}} = \left\{\{(t - \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho}), \boldsymbol{n})\}_{\boldsymbol{n} \in \mathcal{N}_{\mathcal{G}}(t)}\right\}_{t \in T} . \tag{3.2.7}$$

L'intérêt du tramage devient apparent si la taille $\boldsymbol{L}_0$ des trames est suffisamment petite pour assurer que le nombre d'échantillons $|\mathcal{T}_{\mathcal{G}}(\boldsymbol{n})|$ de chaque trame $\boldsymbol{n}$ est petit devant L.

Dans le cas discret, si $\mathbb{T} = \mathbb{Z}^D$, $\mathbb{T}_0 = \mathbb{N}_{L_{0,1}} \times \cdots \times \mathbb{N}_{L_{0,D}}$ et si le signal $\tilde{s}$ est régulièrement échantillonné, toutes les trames auront le même nombre d'échantillons $|\mathcal{T}_{\mathcal{G}}(\boldsymbol{n})| = \prod_{d=1}^{D} L_{0,d}$ et seront régulièrement échantillonnées.

Si $\tilde{s}$ est une fonction de $\mathbb{R}^D$ dans $\mathbb{C}$, son tramage $\mathcal{G}\{\tilde{s}\}$ est une fonction de $\mathbb{T}_0 \times \mathbb{Z}^D$ dans $\mathbb{C}$, où $\mathbb{T}_0 \subset \mathbb{T}$ est le (petit) domaine sur lequel est définie chaque trame.

Si on suppose à présent que certains traitements ont conduit à effectuer une modification de $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$ de manière un produire un signal $\hat{S}$ observé sur $T_{\mathcal{G}}$, peut-on récupérer le signal $\widehat{\tilde{\boldsymbol{s}}(T)}$ correspondant ? De la même manière que dans le cas unidimensionnel, tout $\hat{S}$ n'est pas nécessairement le tramage d'un signal défini sur $\mathbb{T}$, parce que le recouvrement impose des contraintes sur la valeur de certains échantillons du tramage.

Cependant, on peut appliquer la même méthodologie et généraliser le résultat de [92] dans notre cas, pour rechercher :

$$\widehat{\tilde{\boldsymbol{s}}(T)} = \underset{\tilde{\boldsymbol{s}}(T)}{\operatorname{argmin}} \sum_{(t,\boldsymbol{n}) \in T_{\mathcal{G}}} \left(\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}(t, \boldsymbol{n}) - \hat{S}(t, \boldsymbol{n})\right)^2 . \tag{3.2.8}$$

On montre que la solution à ce problème est donnée par :

$$\forall t \in T,\ \mathcal{G}^{-1}\left\{\hat{S}\right\}(t) = \frac{1}{\sum_{\boldsymbol{n} \in \mathcal{N}_{\mathcal{G}}(t)} g(t - \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho}))^2} \times \left(\sum_{\boldsymbol{n} \in \mathcal{N}_{\mathcal{G}}(t)} \hat{S}(t - \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho}), \boldsymbol{n})\, g(t - \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho}))\right), \tag{3.2.9}$$

qui n'est autre que la simple généralisation de la procédure d'addition-recouvrement au cas $\mathbb{T} = \mathbb{R}^D$. Si l'expression 3.2.9 peut paraître déroutante, l'opération correspondante est en fait assez simple. Elle revient à remettre à sa place initiale chaque échantillon du tramage, en pondérant sa contribution dans le résultat en fonction de g. Cette manœuvre permet d'avoir des transitions douces entre les trames pour peu que g soit lisse, et d'ainsi éviter les discontinuités à leurs frontières. On peut remarquer que $\mathcal{G}^{-1} \circ \mathcal{G}$ est toujours l'identité.

Cette procédure de reconstruction des signaux à partir de leur tramage se distingue de l'addition-recouvrement telle que présentée initialement par Allen et Rabiner [5, 6, 40] pour des séries temporelles régulièrement échantillonnées ($D = 1$). En effet, elle s'affranchit de la notion de *fenêtre de synthèse*, distincte de la fenêtre d'analyse g. Pourtant, étant donné un tramage, on pourrait chercher à utiliser une autre fenêtre g_s pour garantir des transitions douces entre les trames, différente de celle utilisée pour l'analyse. Une telle procédure est possible. Dans ce cas, l'expression 3.2.9 devient :

$$\forall t \in T,\ \mathcal{G}^{-1}\left\{\hat{S}\right\}(t) = \frac{1}{\sum_{\boldsymbol{n}\in\mathcal{N}_{\mathcal{G}}(t)} g\left(t - \boldsymbol{n}\cdot\boldsymbol{L}_0\cdot(1-\boldsymbol{\rho})\right) g_s\left(t - \boldsymbol{n}\cdot\boldsymbol{L}_0\cdot(1-\boldsymbol{\rho})\right)} \times \left(\sum_{\boldsymbol{n}\in\mathcal{N}_{\mathcal{G}}(t)} \hat{S}\left(t - \boldsymbol{n}\cdot\boldsymbol{L}_0\cdot(1-\boldsymbol{\rho}), \boldsymbol{n}\right) g_s\left(t - \boldsymbol{n}\cdot\boldsymbol{L}_0\cdot(1-\boldsymbol{\rho})\right)\right), \quad (3.2.10)$$

et on constate que les deux approches coïncident si $g_s = g$. Même si la méthode de reconstruction 3.2.10 apparaît comme une heuristique plutôt que fondée sur la minimisation rigoureuse d'un critère objectif comme peut l'être 3.2.9, nous verrons en section 6.1.3 un exemple où il est important d'utiliser une fenêtre de synthèse différente de celle d'analyse. Dans tous les cas, $\mathcal{G}^{-1} \circ \mathcal{G}$ est toujours l'identité.

3.2.4 Hypothèse locale

Muni de l'opération de tramage et de celle, complémentaire, d'addition-recouvrement, je peux introduire une hypothèse simplificatrice forte, que j'appellerai *indépendance des trames*, ou encore hypothèse *locale*.

Définition 3. Un signal $\tilde{s}$ de $\mathbb{T} = \mathbb{R}^D$ dans $\mathbb{C}$ est à *trames indépendantes* pour un tramage $\mathcal{G}$ si ses différentes trames $\mathcal{G}\{\tilde{s}\}(\cdot, \boldsymbol{n})$ sont indépendantes.

En termes probabilistes, l'hypothèse locale est équivalente à affirmer que :

$$p\left(\tilde{\boldsymbol{s}}(T) \mid \theta\right) \approx \prod_{\boldsymbol{n}} p\left(\mathcal{G}\left\{\tilde{\boldsymbol{s}}(T)\right\}(\cdot, \boldsymbol{n}) \mid \theta\right).$$

où θ sont les hyperparamètres des distributions considérées. Si en plus de l'hypothèse de trames indépendantes, on suppose que chaque trame est un processus gaussien défini sur $\mathbb{T}_0$ et à valeurs dans $\mathbb{C}$, alors $\tilde{s}$ est lui-même un processus gaussien. Se donner un processus gaussien $\tilde{s}$ à trames indépendantes, défini sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$ revient ainsi à se donner un processus gaussien $\mathcal{G}\{\tilde{s}\}$, défini sur $\mathbb{T}_0 \times \mathbb{Z}^D$, dont la fonction de covariance k est donnée par :

$$k\left((t, \boldsymbol{n}), (t', \boldsymbol{n}')\right) \approx \delta_{\boldsymbol{n}\boldsymbol{n}'} k\left(t, t', \boldsymbol{n}\right). \quad (3.2.11)$$

S'il est clair que 3.2.11 ne peut être qu'une approximation compte tenu de l'occurrence des mêmes échantillons dans plusieurs trames, les procédures d'inférences telles que celles présentées en section 2.2.3 sont grandement simplifiées dans ce cas. Elles ne font en effet plus intervenir la matrice de covariance jointe $K\left(\tilde{\boldsymbol{s}}(T), \tilde{\boldsymbol{s}}(T)\right)$ de toutes les observations, de dimension $L \times L$, mais les N matrices de covariance $K\left(\mathcal{G}\left\{\tilde{\boldsymbol{s}}(T)\right\}(\cdot, \boldsymbol{n}), \mathcal{G}\left\{\tilde{\boldsymbol{s}}(T)\right\}(\cdot, \boldsymbol{n})\right)$ des différentes trames, qui sont chacune d'une dimension

En pratique, on observe $\tilde{\boldsymbol{s}}(T)$, puis on calcule son tramage $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$. Enfin, on suppose simplement que les différentes trames sont indépendantes.

$|\mathcal{T}_{\mathcal{G}}(\boldsymbol{n})| \times |\mathcal{T}_{\mathcal{G}}(\boldsymbol{n})|$, et donc beaucoup plus petites. Ainsi, l'hypothèse locale permet de réduire énormément la complexité calculatoire d'un modèle de processus gaussiens.

Une autre propriété fondamentale de l'hypothèse locale est que si les trames sont modélisées comme des processus gaussiens de fonctions de covariance dérivables et que g est elle-même dérivable, alors on peut montrer que la fonction de covariance de $\tilde{s}$ est dérivable également, conduisant à des réalisations presque sûrement dérivables. Autrement dit, l'addition-recouvrement de fonctions lisses reste lisse pour peu que la fenêtre de pondération soit bien conditionnée. Ce résultat est fondamental en audio, car il signifie que le signal résultant d'un traitement distinct dans les différentes trames ne présentera pas de discontinuités gênantes à l'oreille, qui caractérisent les approximations PIC ou PITC de la section précédente.

En figure 3.5, extraite de notre publication [131] sur la séparation de processus gaussiens, j'ai représenté les performances comparées d'un modèle complet, de PITC et du tramage sur un problème simple de régression par processus gaussiens. Il apparaît nettement sur cette figure que le principal avantage du tramage, en plus du gain calculatoire qu'il représente, est l'obtention de moyennes *a posteriori* lisses.

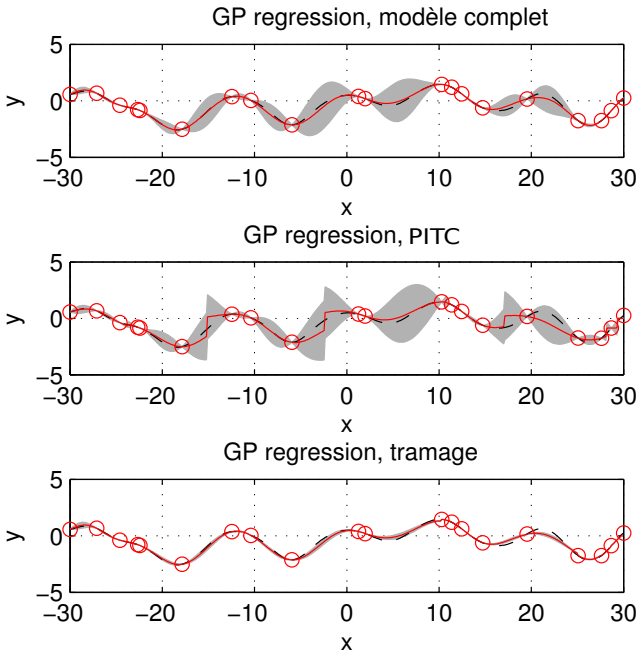

FIGURE 3.5: Comparaison des résultats donnés par un modèle de processus gaussien complet (haut), l'approximation PITC (milieu) et le tramage (bas) pour un problème très simple de régression, pour un recouvrement de $\rho = 0.9$. On observe que le signal estimé par tramage est lisse et correspond de manière fidèle au modèle complet, contrairement à PITC, qui présente certaines discontinuités. Cependant, la variance *a posteriori* donnée par tramage est très nettement sous-évaluée, ce qui n'est pas le cas pour PITC.

Tous ces avantages de considérer des trames indépendantes viennent au prix de plusieurs inconvénients. Tout d'abord, cette hypothèse interdit de considérer des covariances non nulles sur des échelles plus grandes que celles d'une trame. Par conséquent, le choix du domaine de définition $\mathbb{T}_0$ des trames doit permettre d'inclure la majeure partie des dépendances, si on souhaite obtenir de bonnes performances. Ainsi, plus $\mathbb{T}_0$ sera grand, plus l'estimation permettra d'inclure des dépendances à long terme, mais plus l'estimation sera complexe, puisqu'elle mènera à l'inversion de matrices de covariance de trames plus grandes.

En ce sens, cette hypothèse peut être rapprochée de l'utilisation de fonctions de covariance

à support compact, telles qu'évoquées en section 3.1.3. En effet, si on se donne un processus gaussien $\tilde{s}$, à trames indépendantes, le calcul de sa fonction de covariance

$$\forall\,(t,t')\,,\; k\,(t,t') = \mathbb{E}\left[\mathcal{G}^{-1}\left\{\tilde{s}\right\}(t)\,\mathcal{G}^{-1}\left\{\tilde{s}\right\}(t')^{\star}\right] \tag{3.2.12}$$

est extrêmement similaire à ceux menés par Melkumyan et Ramos dans [145], où la "fonction de base" devient ici la fenêtre de pondération g utilisée pour le tramage. Il est d'ailleurs remarquable que ces auteurs utilisent pour g la fenêtre de Hann, extrêmement classique pour le tramage du signal audio. Ce rapprochement n'est cependant pas explicité dans leur article.

Une autre limite de l'hypothèse de trames indépendantes apparaît clairement sur la figure 3.5. On voit que si la moyenne *a posteriori* est bien estimée, il n'en va pas de même pour la variance. En effet, lorsqu'on envisage l'expression 3.2.9 de l'addition-recouvrement en dimension D, on s'aperçoit que supposer des trames indépendantes alors qu'elles ne le sont pas revient à sous-estimer la variance de l'estimée $\mathcal{G}^{-1}$. En effet, plus le recouvrement est grand, plus le nombre de trames dans lesquelles un échantillon donné $t \in \mathbb{T}$ apparaît sera grand. Par conséquent, l'addition-recouvrement 3.2.9 prise en t sera modélisée comme la somme pondérée d'un grand nombre de variables aléatoires indépendantes alors qu'elle est en fait la somme pondérée de variables dépendantes. Sa variance sera ainsi sous-estimée.

Le tramage permet de réduire fortement la difficulté des calculs car il permet de traiter indépendamment des petites portions du signal. Il offre en outre l'avantage de produire des reconstructions lisses. En contrepartie, il conduit à une sous-estimation de la variance *a posteriori* des estimées.

Même si le tramage est une opération extrêmement classique en traitement du signal audio, ses conséquences sur les modèles probabilistes pour la séparation de sources ont été largement négligées jusqu'à très récemment. Dans [127], Le Roux *et al.* mettent au point des procédés pour prendre en compte les dépendances déterministes entre les trames après traitement. Ce faisant, ils ne s'attaquent pas au problème fondamental de leur prise en compte dans les prémices du modèle. L'étude [11] de Badeau est un exemple où ce problème est explicitement abordé. Ce champ de recherche semble aujourd'hui prometteur, et certains chercheurs du domaine s'attellent à tirer profit des dépendances déterministes introduites par le recouvrement pour améliorer les performances de la séparation de sources [118]. Je ne me préoccuperai pas plus avant de cette problématique pour le reste de mon exposé, mais tirerai largement profit de l'hypothèse d'indépendance des trames.

3.3 Processus gaussiens localement stationnaires

3.3.1 Formalisation

Soit $\tilde{s}$ un signal défini sur $\mathbb{T} = \mathbb{R}^D$ et à valeur dans $\mathbb{C}$. Comme vu plus haut en section 3.2, son tramage $\mathcal{G}\left\{\tilde{s}\right\}$ est une fonction de $\mathbb{T}_0 \times \mathbb{Z}^D$ dans $\mathbb{C}$. L'hypothèse de stationnarité locale suppose que toutes les trames $\mathcal{G}\left\{\tilde{s}\right\}(\cdot, \boldsymbol{n})$ de $\tilde{s}$ sont des processus indépendants stationnaires, définis sur $\mathbb{T}_0$ et à valeur dans $\mathbb{C}$.

Définition 4. Un signal $\tilde{s}$ est localement stationnaire pour un tramage $\mathcal{G}$ s'il est à trames indépendantes et si chacune de ses trames est un processus stationnaire.

Si on suppose en plus que chacune des trames est un processus gaussien défini sur $\mathbb{T}_0$, alors le processus $\tilde{s}$ sera un processus gaussien localement stationnaire. La fonction de covariance $k\,(t,t',\boldsymbol{n})$ d'une trame $\boldsymbol{n}$ donnée sera donc :

$$\forall\,(t,t') \in \mathbb{T}_0^2,\; k\,(t,t',\boldsymbol{n}) = k\,(t-t',\boldsymbol{n})\,.$$

Se donner un processus gaussien localement stationnaire $\tilde{s}$, défini sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$ revient ainsi à se donner un processus gaussien $\mathcal{G}\left\{\tilde{s}\right\}$, défini sur $\mathbb{T}_0 \times \mathbb{Z}^D$, dont la fonction de covariance k est donnée par :

$$k\left((t,\boldsymbol{n})\,,(t',\boldsymbol{n}')\right) = \delta_{\boldsymbol{n}\boldsymbol{n}'}\,k\,(t-t',\boldsymbol{n})\,.$$

3.3.2 Représentation spectrale

Considérons à présent comme en section 2.5 que $\mathbb{T} = \mathbb{Z}^D$ et que $\tilde{s}$ est régulièrement échantillonné. Soit $\tilde{\boldsymbol{s}}(T)$ une réalisation de ce signal et $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$ son tramage. Du fait de l'échantillonnage régulier de $\tilde{s}$, chacune des N trames $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}(\cdot, \boldsymbol{n})$ est elle-même régulièrement échantillonnée sur $\mathbb{T}_0$.

Une grandeur qui prendra dans la suite une importance significative sera la transformée de Fourier de chaque trame de $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$. Dans toute la suite de cet exposé, je la noterai [7] :

$$\forall (\boldsymbol{f}, \boldsymbol{n}) \in \mathbb{F} \times \mathcal{N}_{\mathcal{G}},\ s(\boldsymbol{f}, \boldsymbol{n}) = \mathcal{F}_D\{\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}(\cdot, \boldsymbol{n})\}(\boldsymbol{f}). \tag{3.3.1}$$

Comme en section 2.5, seuls les indices fréquentiels $\boldsymbol{f}$ non redondants seront conservés. Ainsi, si les signaux sont à valeurs réelles, seules les indices $\boldsymbol{f}$ correspondants à des fréquences positives seront conservés. Sinon, tous les indices de la transformée de Fourier seront conservés. Dans tous les cas, je désignerai par $\mathbb{F}$ l'ensemble des indices fréquentiels conservés par la transformée de Fourier discrète.

La TFCT d'un signal $\tilde{\boldsymbol{s}}(T)$ est obtenue en calculant le fenêtrage de $\tilde{\boldsymbol{s}}(T)$, puis en calculant sa transformée de Fourier dans chaque trame. Il s'agit ainsi d'une fonction à la fois de l'indice de fréquence $\boldsymbol{f} \in \mathbb{F}$ et de l'index de trame $\boldsymbol{n} \in \mathcal{N}_{\mathcal{G}}$

s porte le nom de Transformée de Fourier à Court Terme (TFCT) de $\tilde{\boldsymbol{s}}(T)$. C'est un signal défini pour $\boldsymbol{f} \in \mathbb{F}$ et $\boldsymbol{n} \in \mathcal{N}_{\mathcal{G}}$ [8].

On peut appliquer les théorèmes de Bochner et de Wiener-Khinchin, évoqués en section 2.5, à chaque trame de $\mathcal{G}\{\tilde{\boldsymbol{s}}(T)\}$. Les principaux résultats correspondants sont donnés en table 3.1.

On voit l'intérêt de l'hypothèse de stationnarité locale. Elle permet de supposer que tous les éléments de la TFCT d'un signal sont indépendants, et donc de les traiter isolément les uns des autres. Je vais illustrer cette propriété en section 3.3.3 lorsque je présenterai le filtrage de Wiener généralisé. On peut noter que ce modèle est aussi appelé dans la littérature modèle gaussien local [70, 210, 214]. Il est cependant rare que sa connexion avec un modèle de formes d'ondes soit explicité comme c'est le cas ici et je n'ai pas connaissance d'une étude le développant pour un domaine de définition $\mathbb{T} = \mathbb{Z}^D$ de dimension quelconque.

3.3.3 Filtrage de Wiener généralisé

Une fois de plus, je vais aborder le problème de régression déjà considéré en sections 2.2.3, 2.5.2 et 2.5.4, pour démontrer l'intérêt du modèle localement stationnaire pour des problèmes classiques. Supposons que l'observation $\tilde{x}$ soit la somme d'un processus gaussien localement stationnaire (PGLS) $\tilde{s}$, de DSP $P_s(\boldsymbol{f}, \boldsymbol{n})$ et d'un autre PGLS $\tilde{\epsilon}$ indépendant de $\tilde{s}$, de DSP $P_\epsilon(\boldsymbol{f}, \boldsymbol{n})$. Du fait de la linéarité de la TFCT, on a :

$$\forall (\boldsymbol{f}, \boldsymbol{n}),\ x(\boldsymbol{f}, \boldsymbol{n}) = s(\boldsymbol{f}, \boldsymbol{n}) + \epsilon(\boldsymbol{f}, \boldsymbol{n}),$$

et on a par ailleurs :

- Tous les coefficients $x(\boldsymbol{f}, \boldsymbol{n})$ sont indépendants, puisque c'est le cas de ceux de s et de ϵ, qui sont par ailleurs indépendants entre eux.
- Chaque $x(\boldsymbol{f}, \boldsymbol{n})$ a pour distribution :

$$\boldsymbol{x}(\boldsymbol{f}, \boldsymbol{n}) \sim \mathcal{N}_c(0, P_s(\boldsymbol{f}, \boldsymbol{n}) + P_\epsilon(\boldsymbol{f}, \boldsymbol{n})).$$

7. Pour des signaux de grande taille, je ne serai jamais intéressé par la simple transformée de Fourier $\mathcal{F}_D\{\tilde{\boldsymbol{s}}(T)\}$ des observations, mais plutôt par leur TFCT. A partir de ce point de l'exposé, la notation $\boldsymbol{s}$ désignera par conséquent toujours la TFCT d'une forme d'onde $\tilde{\boldsymbol{s}}$.

8. $\mathcal{N}_{\mathcal{G}}$ est l'ensemble des trames $\boldsymbol{n} \in \mathbb{Z}^D$ dans lesquelles apparaît au moins une fois un des points de T. Pour $D = 1$, c'est tout simplement $[0, \ldots, (N-1)]$ où N est le nombre de trames.

1. La DSP $P(\boldsymbol{f}, \boldsymbol{n})$ de la trame $\boldsymbol{n}$ de $\mathcal{G}\{\tilde{s}\}$ à l'indice discret de fréquence $\boldsymbol{f}$ est donnée par
$$P(\boldsymbol{f}, \boldsymbol{n}) = \mathcal{F}_D\{k(\cdot, \boldsymbol{n})\}(\boldsymbol{f}).$$
2. Si les trames sont suffisamment grandes, alors **tous les éléments de $s(\boldsymbol{f}, \boldsymbol{n})$ sont indépendants.** L'indépendance pour une trame donnée provient de la stationnarité, l'indépendance pour des trames différentes provient de l'hypothèse locale.
3. De plus, on a :
$$s(\boldsymbol{f}, \boldsymbol{n}) \sim \mathcal{N}_c(0, P(\boldsymbol{f}, \boldsymbol{n})), \tag{3.3.2}$$
où $\mathcal{N}_c$ est la distribution complexe circulaire 2.5.9 :
$$\mathcal{N}_c\left(z \mid \mu, \sigma^2\right) = \frac{1}{\pi\sigma^2} \exp\left(-\frac{|z-\mu|^2}{\sigma^2}\right).$$
4. Tous les points $(\boldsymbol{f}, \boldsymbol{n})$ étant indépendants, l'opposé $\mathcal{L}(\tilde{\boldsymbol{s}} \mid P)$ de la log-vraisemblance des observations étant donnée la DSP P est donnée par :
$$\mathcal{L}(\tilde{\boldsymbol{s}} \mid P) = -\log p(\tilde{s} \mid P) = \sum_{(\boldsymbol{f}, \boldsymbol{n})} \left(\log \pi + \log P(\boldsymbol{f}, \boldsymbol{n}) + \frac{|s(\boldsymbol{f}, \boldsymbol{n})|^2}{P(\boldsymbol{f}, \boldsymbol{n})}\right)$$
qui vaut, à une constante additive indépendante de P près :
$$\mathcal{L}(\tilde{\boldsymbol{s}} \mid P) = \sum_{(\boldsymbol{f}, \boldsymbol{n})} d_0\left(|s(\boldsymbol{f}, \boldsymbol{n})|^2 \mid P(\boldsymbol{f}, \boldsymbol{n})\right) \tag{3.3.3}$$
où
$$d_0(x \mid y) = \frac{x}{y} - \log\frac{x}{y} - 1 \tag{3.3.4}$$
est la divergence d'Itakura-Saito [69, 71].
5. Puisqu'on a 3.3.3, étant donnée la TFCT $s(\boldsymbol{f}, \boldsymbol{n})$ d'un processus gaussien localement stationnaire, l'estimation $v(\boldsymbol{f}, \boldsymbol{n})$ de la DSP au maximum de vraisemblance est :
$$v(\boldsymbol{f}, \boldsymbol{n}) = |s(\boldsymbol{f}, \boldsymbol{n})|^2, \tag{3.3.5}$$
et est appelée *spectrogramme* de $\tilde{\boldsymbol{s}}(T)$.

Table 3.1: Propriétés de la représentation spectrale d'un processus gaussien localement stationnaire

– La DSP de $\tilde{x}$ est donc la somme de celle de $\tilde{s}$ et de $\tilde{\epsilon}$:

$$\forall (\boldsymbol{f}, \boldsymbol{n}),\, P_x(\boldsymbol{f}, \boldsymbol{n}) = P_s(\boldsymbol{f}, \boldsymbol{n}) + P_\epsilon(\boldsymbol{f}, \boldsymbol{n}).$$

Dans ces conditions, on peut facilement montrer que la distribution *a posteriori* de $\boldsymbol{s}$ étant donné $\boldsymbol{x}$ devient :

$$\boldsymbol{s}(\boldsymbol{f}, \boldsymbol{n}) \mid \boldsymbol{x}(\boldsymbol{f}, \boldsymbol{n}) \sim \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}}(\boldsymbol{f}, \boldsymbol{n}), \sigma^2_{\text{post}}(\boldsymbol{f}, \boldsymbol{n})\right),$$

avec :

$$\begin{cases} \boldsymbol{\mu}_{\text{post}}(\boldsymbol{f}, \boldsymbol{n}) & = \frac{P_s(\boldsymbol{f}, \boldsymbol{n})}{P_s(\boldsymbol{f}, \boldsymbol{n}) + P_\epsilon(\boldsymbol{f}, \boldsymbol{n})} \boldsymbol{x}(\boldsymbol{f}, \boldsymbol{n}) \\ \sigma^2_{\text{post}}(\boldsymbol{f}, \boldsymbol{n}) & = \frac{P_s(\boldsymbol{f}, \boldsymbol{n}) P_\epsilon(\boldsymbol{f}, \boldsymbol{n})}{P_s(\boldsymbol{f}, \boldsymbol{n}) + P_\epsilon(\boldsymbol{f}, \boldsymbol{n})} \end{cases}. \tag{3.3.6}$$

Puisque cette distribution *a posteriori* est gaussienne, l'estimée aux moindres carrés de $\boldsymbol{s}$ étant donné $\boldsymbol{x}$ est :

$$\forall (\boldsymbol{f}, \boldsymbol{n}),\, \widehat{s}(\boldsymbol{f}, \boldsymbol{n}) = \frac{P_s(\boldsymbol{f}, \boldsymbol{n})}{P_s(\boldsymbol{f}, \boldsymbol{n}) + P_\epsilon(\boldsymbol{f}, \boldsymbol{n})} x(\boldsymbol{f}, \boldsymbol{n}). \tag{3.3.7}$$

L'opération 3.3.7 va prendre une importance déterminante dans toute la suite de cet exposé. Elle permet de récupérer une estimée de la TFCT du signal recherché à partir de l'observation de x, pour peu qu'on dispose des DSP P_s et P_ϵ des signaux constituants $\tilde{x}$. Elle est souvent appelée filtrage de Wiener généralisé, et est couramment utilisée en séparation de sources [30, 17]. Le signal temporel $\widehat{\tilde{s}}$ est récupéré facilement par TFCT inverse, incluant une addition-recouvrement.

Un processus gaussien localement stationnaire et centré (PGLS) $\tilde{s}$ est complètement déterminé par sa DSP $P_s(\boldsymbol{f}, \boldsymbol{n}) \geq 0$. La somme de PGLS reste un PGLS, et ce modèle permet des opérations très rapides de filtrage.

Il est remarquable que cette opération de filtrage apporte les mêmes gains en complexité pour des processus localement stationnaires que le filtrage de Wiener pour les processus stationnaires. Si le modèle stationnaire n'est souvent pas un bon choix, comme en audio ($D = 1$) où la DSP du signal évolue constamment, le modèle localement stationnaire est quant à lui souvent un excellent choix. Dans cette section, j'ai montré qu'on n'a que peu à gagner à se restreindre au cas $D = 1$, puisque tous les résultats s'appliquent de la même manière aux processus localement stationnaires définis sur $\mathbb{Z}^D$.

3.4 Conclusion

Dans ce chapitre, j'ai évoqué la grande complexité des calculs requis par toute tâche d'inférence basée sur un modèle de processus gaussien, ainsi que quelques-unes des approches actuelles les plus populaires pour leur simplification.

Dans ce contexte, j'ai introduit la technique du fenêtrage, ou tramage, qui est classique en traitement des séries temporelles. Le tramage d'un signal consiste en la production d'un ensemble de portions de taille réduite du signal, qui ont entre elles un certain recouvrement. Si on fait l'hypothèse que ces trames sont des processus gaussiens indépendants, on peut effectuer les tâches d'inférence indépendamment dans chacune, ce qui conduit à une simplification importante de la complexité des calculs. Le signal original peut être récupéré par une technique simple d'addition-recouvrement. J'ai montré que les liens entre cette approche et l'état de l'art sont importants.

Si on fait de plus l'hypothèse que le signal est un processus gaussien stationnaire dans chaque trame, le modèle correspondant y gagne encore en simplicité, puisque les procédures d'inférence peuvent être faites très simplement sur la TFCT du signal. On a vu qu'un PGLS est caractérisé par sa DSP pour chaque trame, estimée par son spectrogramme.

Si ces résultats peuvent paraître classiques pour $D = 1$, une contribution de ce chapitre est de les présenter dans leur généralité en dimension quelconque. On verra par ailleurs que ces développements me permettront en partie II d'introduire très simplement la séparation de PGLS.

Chapitre 4

Modèles de densités spectrales de puissance

Dans ce chapitre, au lieu de considérer un seul processus gaussien $\tilde{s}$ défini sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$ comme je l'ai fait dans la plupart des chapitres précédents, je vais utiliser les notations introduites en section 1.3.2 pour envisager plutôt un groupe de J processus gaussiens indépendants $\{\tilde{s}(\cdot, j)\}_{j=1,\dots,J}$, tous définis sur $\mathbb{T} = \mathbb{Z}^D$ et dont je considérerai une réalisation $\{\tilde{\boldsymbol{s}}(T, j)\}_{j=1,\dots,J}$ régulièrement échantillonnée sur L points $T = [t_1, \dots, t_L]$ de $\mathbb{T}$. Par extension, $\tilde{s}$ désignera le groupe de processus, défini sur $\mathbb{T} \times \mathbb{N}_J$ et à valeurs dans $\mathbb{C}$. De la même manière, toutes les grandeurs définies plus haut pour un seul processus telles que sa DSP P ou son spectrogramme v sont étendues au cas des groupes, et les notations restent les mêmes, à part l'introduction d'un indice j supplémentaire, relatif à l'onde considérée.

Dans l'ensemble de ce chapitre, les processus sont supposés centrés (à moyenne nulle) et modélisés comme des PGLS tels que définis en section 3.3.

4.1 Motivations et critère d'apprentissage

4.1.1 Modèles paramétriques de DSP

En section 3.3, j'ai montré qu'un PGLS est complètement déterminé par sa DSP $P(\boldsymbol{f}, \boldsymbol{n})$ dans chaque trame. De la même manière, J processus $\{\tilde{s}(\cdot, j)\}_{j=1,\dots,J}$ gaussiens localement stationnaires et indépendants sont caractérisés par leur DSP $P(\boldsymbol{f}, \boldsymbol{n}, j)$. En effet, par application du théorème de Wiener-Khinchin, la connaissance de cette DSP est équivalente à celle de la fonction de covariance de chacun des J processus dans chaque trame et détermine ainsi complètement les processus du groupe.

La plupart du temps, on ne connaît pas les DSP des processus étudiés, mais on dispose d'une réalisation $\tilde{\boldsymbol{s}}$. Cette observation permet d'estimer les DSP et comme on l'a vu en 3.3.5, l'estimation au maximum de vraisemblance des DSP est donnée par le spectrogramme v, défini par :

$$\forall (\boldsymbol{f}, \boldsymbol{n}, j),\ v(\boldsymbol{f}, \boldsymbol{n}, j) = |s(\boldsymbol{f}, \boldsymbol{n}, j)|^2,$$

où $\boldsymbol{s}$ est la TFCT de $\tilde{\boldsymbol{s}}$. Ainsi, les opérations faisant intervenir la DSP des signaux, comme le problème du débruitage considéré en section 3.3.3 peuvent être menées en utilisant leurs spectrogrammes v au lieu de leur DSP P.

Cependant, le principal problème de cette approche est que le nombre de points $(\boldsymbol{f}, \boldsymbol{n})$ est en pratique comparable au nombre L d'observations[1], ce qui signifie que le nombre de paramètres d'un PGLS est comparable au nombre de ses observations.

Bien que je présenterai ces problématiques plus loin, les applications pratiques des PGLS à la séparation de sources que je vais considérer rendent indispensable la réduction du nombre de ces paramètres. En effet, elles impliquent deux problématiques principales :

1. Pour L fixé, ce nombre varie en fonction du recouvrement choisi entre les trames.

1. L'estimation des DSP des J signaux à partir de I observations. Pour peu que $I < J$, cette estimation implique plus de paramètres que d'observations.
2. La transmission des DSP pour le codage. Dans ce cas, le problème n'est pas d'estimer des DSP inconnues mais plutôt de transmettre des DSP connues à bas coût.

Dans ces deux cas de figure, il est indispensable de réduire le nombre de paramètres caractérisant les PGLS étudiés. Pour comprendre l'approche adoptée, on peut voir que les DSP $P(\boldsymbol{f}, \boldsymbol{n}, j)$ d'un groupe de J ondes se comprennent comme une fonction de $\mathbb{F}\times\mathcal{N}_\mathcal{G}\times\mathbb{N}_J$ dans $\mathbb{R}_+$ où je rappelle que $\mathbb{F}$ et $\mathcal{N}_\mathcal{G}$ sont respectivement l'ensemble des indices fréquentiels et des indices de trames de la TFCT de chaque onde. Chacun de ces indices est un vecteur de dimension D [2]. Le fait que P soit à valeurs dans $\mathbb{R}_+$ est une conséquence du théorème de Bochner vu en section 2.5.3 page 38.

Les processus gaussiens localement stationnaires sont caractérisés par leur DSP. Si on veut les utiliser en pratique, il est souvent nécessaire d'en réduire la dimension.

Au lieu de considérer comme paramètres du modèle les valeurs $P(\boldsymbol{f}, \boldsymbol{n}, j)$ prises par la DSP en tout point $(\boldsymbol{f}, \boldsymbol{n}, j)$, on peut procéder à une approche *paramétrique*, et supposer que P appartient à un ensemble $\mathcal{P}(\cdot \mid \theta)$ de fonctions de $\mathbb{F} \times \mathcal{N}_\mathcal{G} \times \mathbb{N}_J \to \mathbb{R}_+$, indexé par un *lot de paramètres* θ, appelés *paramètres de source* dans la suite de cet exposé. On supposera ainsi que :

$$\forall P \in \mathbb{R}_+^{\mathbb{F}\times\mathcal{N}_\mathcal{G}\times\mathbb{N}_J},\ \exists\theta :\ \forall(\boldsymbol{f}, \boldsymbol{n}, j),\ P(\boldsymbol{f}, \boldsymbol{n}, j) \approx \mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta) \tag{4.1.1}$$

L'intérêt fondamental de cette approche réside dans le fait que si 4.1.1 est vérifiée, alors il suffit de transmettre les paramètres de sources θ pour avoir la valeur approchée de P en chaque point $(\boldsymbol{f}, \boldsymbol{n}, j)$ au lieu de transmettre toutes ses valeurs directement. Si θ contient peu d'éléments, la réduction du nombre de paramètres est considérable. On peut envisager plusieurs modèles de sources $\mathcal{P}$. Dans la suite de cet exposé, j'en considérerai deux que je présenterai en sections 4.2 et 4.3. Le premier est est valable pour $D = 1$ et est inspiré de techniques de compression d'image tandis que le deuxième est basé sur une décomposition de P en facteurs non-négatifs et vaut pour tout D.

Au lieu de devoir envoyer la valeur prise par une droite en chaque point, il est plus efficace d'en envoyer la pente et l'ordonnée à l'origine. Tel est le sens de l'approche paramétrique 4.1.1. Elle n'est bien sûr valable pour cet exemple que si les données sont bien assimilables à une droite.

La justification de l'utilisation d'un modèle paramétrique pour représenter un ensemble de DSP de processus gaussiens localement stationnaires peut rappeler la discussion de la section 2.1, où les processus gaussiens eux-mêmes ont été introduits comme une alternative possible à un modèle paramétrique.

En ce sens, si les approches actuelles font l'hypothèse que les DSP des signaux sont membres d'une famille paramétrique donnée (comme le modèle NMF), il me semble clair aujourd'hui qu'une piste de recherche intéressante est de ne plus faire cette hypothèse simplificatrice pour plutôt considérer que P est la réalisation d'un processus aléatoire, caractérisé par des hyperparamètres pertinents. Certains auteurs ont déjà fait le premier pas dans cette direction [45, 149], mais les modèles correspondants ne permettent encore que d'inclure des dépendances locales ou souffrent d'une grande complexité calculatoire. La découverte de processus aléatoires à valeurs positives suffisamment souples pour rendre compte de la diversité des DSP de signaux réels reste un enjeu non résolu. Je ne me préoccuperai pas de cette idée pour la suite de ce document.

4.1.2 Apprentissage des paramètres

Étant donnée une famille $\mathcal{P}$ de modèles, la question centrale de l'approche paramétrique est celle de la détermination des paramètres $\theta^\star$ qui permettent au mieux de rendre compte des observations

2. Pour une image par exemple, les fréquences sont spatiales et les trames situées dans le plan.

$\boldsymbol{s}$. En d'autres termes, si on dispose de l'observation $\boldsymbol{s}$ des TFCT d'un groupe de J PGLS, la question se pose d'identifier le lot de paramètres $\theta^\star$ qui est le mieux susceptible de rendre compte des observations $\boldsymbol{s}$.

Le cadre théorique des processus gaussiens offre une manière élégante d'aborder ce problème. Comme on l'a vu en section 3.3.2, l'opposé de la log-vraisemblance des observations $\tilde{\boldsymbol{s}}$ d'un PGLS est donnée par 3.3.3 page 54, qui devient pour J signaux indépendants [3] :

$$\mathcal{L}\left(\tilde{\boldsymbol{s}} \mid P\right) = \sum_{(\boldsymbol{f},\boldsymbol{n},j)} d_0\left(\left|s\left(\boldsymbol{f},\boldsymbol{n},j\right)\right|^2 \mid P\left(\boldsymbol{f},\boldsymbol{n},j\right)\right). \tag{4.1.2}$$

Dans ces conditions, si on remplace P par $\mathcal{P}\left(\cdot \mid \theta\right)$ comme le suggère l'approche paramétrique 4.1.1, le problème de trouver $\theta^\star$ peut être abordé selon le critère du maximum de vraisemblance [4], et devient :

$$\begin{aligned} \theta^\star &= \operatorname*{argmin}_\theta \mathcal{L}\left(\tilde{\boldsymbol{s}} \mid \theta\right) \\ &= \operatorname*{argmin}_\theta \sum_{(\boldsymbol{f},\boldsymbol{n},j)} d_0\left(v\left(\boldsymbol{f},\boldsymbol{n},j\right) \mid \mathcal{P}\left(\boldsymbol{f},\boldsymbol{n},j \mid \theta\right)\right). \end{aligned} \tag{4.1.3}$$

L'être humain a une sensibilité acoustique reliée au logarithme de l'amplitude des sons entendus [26] et non pas à leur valeur absolue. Il est intéressant qu'une estimation au maximum de vraisemblance d'un modèle gaussien mène à utiliser une divergence proche d'une différence quadratique de log-spectrogrammes.

Comme on le voit, *quelle que soit* la famille $\mathcal{P}$ de modèles paramétriques choisie pour modéliser la DSP de processus gaussiens localement stationnaires, l'apprentissage d'un modèle de sources se comprend simplement comme la recherche du modèle de DSP qui est le plus proche du spectrogramme des observations. La seule subtilité introduite par le modèle gaussien est que l'écart entre ce spectrogramme et le modèle n'est pas mesuré en utilisant une fonction classique comme l'erreur quadratique, mais plutôt une distance spécifique, qui peut se comprendre grossièrement (voir 10.3.3 page 134) comme une erreur quadratique sur les logarithmes de ses opérandes.

Le problème de la recherche des paramètres de sources $\theta^\star$ qui rendent au mieux compte des observations aboutit donc sur des bases probabilistes à la minimisation d'une fonction de coût 4.1.3. Nous quittons alors les terres de l'analyse probabiliste pour résoudre un problème d'*optimisation* : il s'agit de trouver la valeur $\theta^\star$ qui maximise la vraisemblance $p\left(\boldsymbol{s} \mid \theta\right)$, dont on connaît l'expression analytique 4.1.2. En fonction de la famille paramétrique $\mathcal{P}$ choisie, ce problème admet des solutions plus ou moins directes à obtenir.

4.2 Modèle par compression d'images (CI) dans le cas $D = 1$

4.2.1 Modèle

Dans le cas des séries temporelles ($D = 1$), on a déjà vu que les spectrogrammes d'un groupe de J DSP peuvent être compris comme un tenseur v de dimension $F \times N \times J$. La principale idée qui guide cette section est que ce tenseur peut être vu comme un ensemble de J *images* de même dimension $F \times N$. À ce titre, il peut être efficacement transmis en utilisant des algorithmes de compression d'images. Partant de cette idée [5], j'ai proposé dans [137] d'utiliser des techniques de compression d'images avec pertes appliquées sur les spectrogrammes des observations, comme

3. Je rappelle que $d_0\left(x \mid y\right) = \frac{x}{y} - \log\frac{x}{y} - 1$ désigne la divergence d'Itakura-Saito.

4. Je ne considèrerai pas le critère du *maximum a posteriori*, qui permet en outre d'intégrer une distribution *a priori* $p\left(\theta\right)$ sur la valeur des paramètres recherchés. Ce critère est utile lorsqu'on souhaite privilégier des paramètres ayant certaines propriétés, en plus de leur seule adéquation avec les observations. Par exemple, on peut souhaiter obtenir des décompositions NTF dont les activations ou les spectres de bases sont lisses [181, 69, 19, 46]. Je reviendrai au chapitre 7 page 93 sur cette problématique.

5. C'est lors d'une discussion avec Laurent Girin à Grenoble que nous est apparue cette idée.

l'algorithme JPEG ou JPEG2000 [216, 220], pour réduire le poids des paramètres du modèle PGLS[6]. L'idée sous-jacente est que s'il est possible de compresser une image par un facteur 100 sans que cela soit visible, la même approximation doit pouvoir donner des résultats intéressants si l'image considérée est un spectrogramme.

De manière à formaliser cette approche, il me faut évoquer rapidement la structure globale commune à la plupart des algorithmes de compression d'images. La première étape du traitement consiste à découper l'image en trames. Pour JPEG, ces trames sont nécessairement de taille 8×8, tandis qu'elles sont de taille arbitraire pour JPG2000. Ensuite, chacune des trames subit une transformation qui dépend de la méthode. Ainsi, on considère une transformée en cosinus discrète (*Discrete Cosine Transform*, DCT, en anglais) pour les trames 8×8 extraites par JPEG tandis qu'on utilise une transformée en ondelettes pour JPEG2000.

Dans tous les cas, si M est une image de dimension $F \times N$, je désignerai par $\mathcal{C}\{M\}$ l'ensemble des coefficients produits après tramage et transformation par la méthode considérée. Ainsi, pour JPEG, $\mathcal{C}\{M\}$ regroupera des coefficients de DCT des trames 8×8 composant l'image M, tandis qu'il regroupera des coefficients d'ondelettes pour JPG2000 [220]. Par ailleurs, $\mathcal{C}^{-1}$ désignera l'opération inverse à $\mathcal{C}$, qui reconstruit une image de taille $F \times N$ à partir des coefficients de $\mathcal{C}\{M\}$. De cette manière, l'application successive de $\mathcal{C}$ et de $\mathcal{C}^{-1}$ ne produit aucune modification sur l'image :

$$\mathcal{C}^{-1}\{\mathcal{C}\{M\}\} = M.$$

Étant donné un algorithme de compression d'image, l'ensemble des paramètres θ_{CI} (Compression d'Image) nécessaires à l'encodage de v est donc :

$$\theta_{CI} = \bigcup_{j=1}^{J} \mathcal{C}\{v(\cdot,\cdot,j)\} \tag{4.2.1}$$

et regroupe simplement les coefficients produits par l'analyse indépendante des J spectrogrammes des formes d'ondes.

Comme on peut le voir, les paramètres 4.2.1 du modèle CI ne sont pas moins nombreux que ceux de v, contrairement à ceux du modèle NTF. Au contraire, s'agissant simplement des coefficients d'une autre représentation des spectrogrammes, ils ont toutes les chances d'être exactement aussi nombreux.

Le modèle par compression d'image (CI) revient en pratique à utiliser une compression de type JPEG sur les spectrogrammes des signaux. D'un point de vue formel, le nombre de paramètres de CI est très important, mais ces paramètres sont très efficacement transmis après une étape de quantification, qui en annule la plupart.

L'avantage de ce paramétrage apparaît lorsqu'on considère que bien qu'ils soient aussi nombreux que le nombre d'entrées de v, les éléments de $\mathcal{C}\{M\}$ peuvent généralement être considérés comme indépendants, ce qui permet d'adopter une stratégie de quantification scalaire pour les encoder au lieu d'une difficile quantification vectorielle, tout en atteignant des gains importants en débit.

Je reviendrai plus en détail sur le problème de la quantification des paramètres du modèle CI en section 10.3 page 133. Pour l'heure, il me suffira de dire qu'au lieu de compresser des images correspondant aux spectrogrammes des observations, il faut plutôt compresser leurs logarithmes, qui offrent une dynamique beaucoup moins importante et qui sont donc bien mieux encodés comme une image d'un faible nombre de niveaux de gris. J'ai représenté en figure 4.1 le résultat de la reconstruction d'un spectrogramme de voix par JPEG, après quantification de θ_{CI} en utilisant un paramètre de qualité $q = 10/100$.

6. Cependant, l'utilisation de $\log v$ au lieu de v dans les traitements n'était justifié dans [137] que sur une base empirique et non pas par les considérations plus rigoureuses de la section 10.3.1. L'utilisation de techniques de compression d'images dans ce contexte a fait l'objet d'un dépôt de brevet dont je suis co-inventeur avec Laurent Girin, Roland Badeau et Gaël Richard [83].

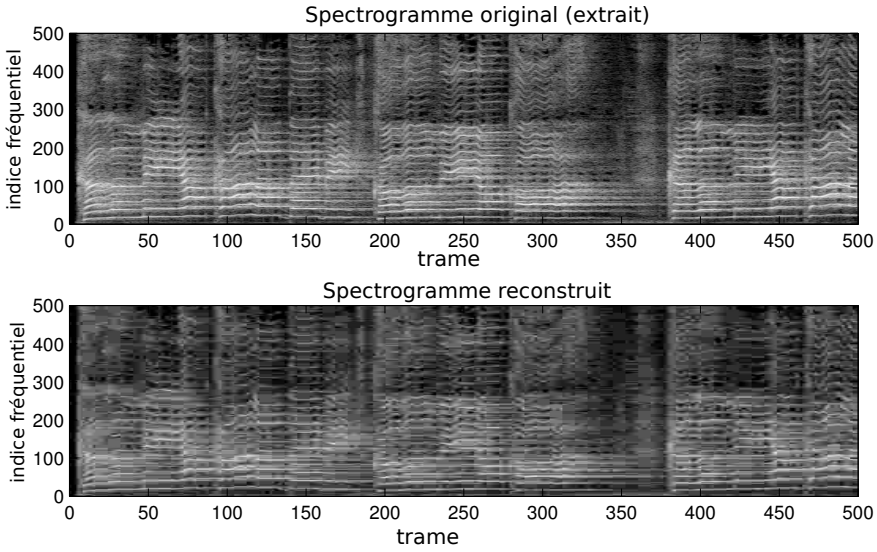

FIGURE 4.1: Illustration du modèle CI. L'apprentissage et la quantification du modèle se fait simplement par compression JPEG des log-spectrogrammes $\log v$ des observations. Le modèle θ_{CI} seul produit une reconstruction sans perte. C'est sa quantification $\bar{\theta}_{CI}$ qui introduit un codage efficace, en même temps qu'une erreur de reconstruction (d'après [137]).

4.2.2 Apprentissage des paramètres

Dans la mesure où le modèle CI est simplement une autre représentation inversible de $\boldsymbol{v}$, on a :

$$\theta^{\star}_{CI} = \mathcal{C}\left\{\boldsymbol{v}\right\}. \tag{4.2.2}$$

En effet, on a ainsi $\mathcal{P}\left(\cdot \mid \theta^{\star}_{CI}\right) = \boldsymbol{v}$, qui minimise nécessairement 4.1.3. Bien entendu, la transformation $\mathcal{C}$ de n'importe quelle fonction bijective de $\boldsymbol{v}$ peut être envisagée avec les mêmes conséquences, puisque $\boldsymbol{v}$ peut toujours être récupéré exactement dans ce cas. On verra en particulier en section 10.3 page 133 qu'il est plus efficace de considérer les paramètres :

$$\theta^{\star}_{CI} = \mathcal{C}\left\{\log \boldsymbol{v}\right\}, \tag{4.2.3}$$

à la place de 4.2.2. Dans ce cas, $\boldsymbol{v}$ est reconstruit simplement par

$$\boldsymbol{v} = \mathcal{P}\left(\cdot \mid \theta^{\star}_{CI}\right) = \exp\left(\mathcal{C}^{-1}\left\{\theta^{\star}_{CI}\right\}\right). \tag{4.2.4}$$

4.3 Factorisation non négative (D quelconque)

4.3.1 Modèle

Le premier modèle de sources que je vais présenter porte le nom de *factorisation non négative*. Ce modèle s'inscrit dans la continuité de travaux menés par une très large communauté de chercheurs [34, 69, 19] sur le problème de la factorisation non négative de matrices ou de tenseurs (Nonnegative Matrix/Tensor Factorization, NMF/NTF, en anglais). Le principe de ces approches est le suivant.

Considérons un tenseur [7] V de dimensions $F \times N \times J$, dont toutes les entrées $V\left(f,n,j\right)$ sont des réels positifs. Il est souvent utile pour faire l'analyse de V de le décomposer comme un produit d'un nombre réduit de facteurs. Ainsi, le modèle dit *canonique*, *polyadique*, *NTF*, ou encore *nonnegative PARAFAC* pose :

$$\forall\left(f,n,j\right), V\left(f,n,j\right) = \sum_{k=1}^{K} W\left(f,k\right) H\left(n,k\right) Q\left(j,k\right), \tag{4.3.1}$$

7. Un tenseur peut être compris comme un tableau à plusieurs entrées.

où W, H et Q sont des matrices de dimensions $F \times K$, $N \times K$ et $J \times K$ respectivement, pour $K \in \mathbb{N}$ appelé nombre de facteurs, ou de composantes, selon la communauté. Ce modèle se distingue du très classique PARAFAC [96] par le fait que toutes les matrices W, H et Q considérées sont supposées ne contenir que des entrées positives. Lorsque $J = 1$, on parle de NMF parce que V est une matrice dans ce cas, tandis qu'on parle plus volontiers de NTF dans le cas général.

En fonction du contexte applicatif, les matrices W, H et Q peuvent être interprétées différemment. Cependant, W sera souvent compris comme une collection de K *motif*s, ou *formes de base*, dont H donnera *l'activation* pour chaque trame n, tandis que Q sera souvent compris comme indiquant les coefficients de *gain* des motifs sur les J *modalités* de V. Quoiqu'il en soit, il est remarquable que ce modèle ait des applications dans des domaines aussi différents que l'analyse statistique de corpus de textes [106], l'analyse de données biomédicales [34], le débruitage et la reconnaissance d'images [142, 128], la transcription polyphonique [19] ou enfin la séparation de sources audio [69, 155, 163].

Cette popularité s'explique par l'existence d'algorithmes à la fois simples à implémenter et très efficaces pour estimer W, H et Q à partir de la seule observation de V. Ces algorithmes ont été introduits dans l'article célèbre de Lee et Seung [128] dans le cas $J = 1$ et ont fait l'objet d'un constant effort de recherche depuis. Leurs propriétés de convergence ont fait l'objet de nombreux travaux [13, 148, 71] et sont à présent établies sur des bases théoriques solides.

Dans ce texte, je propose d'utiliser une factorisation non-négative pour modéliser la DSP d'un groupe de J processus gaussiens localement stationnaires définis sur $\mathbb{Z}^D$. Pour ce faire, j'étends la notation habituelle 4.3.1 à des indices fréquentiels $\boldsymbol{f}$ et $\boldsymbol{n}$ de dimension D :

- Soit $K \in \mathbb{N}$, appelé le nombre de composantes.
- Soit $\left\{ \{W(\boldsymbol{f}, k) \in \mathbb{R}_+\}_{\boldsymbol{f} \in \mathbb{F}} \right\}_{k=1,\dots,K}$ un ensemble de K *bases spectrales.*
- Soit $\left\{ \{H(\boldsymbol{n}, k) \in \mathbb{R}_+\}_{\boldsymbol{n} \in \mathcal{N}_\mathcal{G}} \right\}_{k=1,\dots,K}$ un ensemble de K *activations*, donnant l'amplitude (positive) de chaque base spectrale sur chaque trame.
- Soit $\{Q(j,k) \in \mathbb{R}_+\}_{(j,k) \in \mathbb{N}_J \times \mathbb{N}_K}$ un ensemble de $J \times K$ réels positifs, appelés *gains.*

Le modèle NTF que je considère pose alors :

$$\mathcal{P}^{NTF}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta) = \sum_{k=1}^{K} W(\boldsymbol{f}, k) H(\boldsymbol{n}, k) Q(j, k) \tag{4.3.2}$$

et est paramétré par :

$$\theta_{NTF} = \{W, H, Q\}.$$

Je propose dans cette étude d'utiliser un modèle NTF dans le cas général des PGLS définis sur $\mathbb{Z}^D$. Cette extension n'implique pas de difficulté majeure : la seule différence à la situation classique est que les indices fréquentiels $\boldsymbol{f}$ et les indices de trames $\boldsymbol{n}$ sont des vecteurs de dimension D. Je ne présenterai cependant pas d'applications d'une telle extension dans ce document.

Comme on le voit, le modèle NTF revient à approximer la valeur de la DSP de chacune des J formes d'ondes en chaque point $(\boldsymbol{f}, \boldsymbol{n})$ comme une somme pondérée de K formes spectrales. La matrice Q de gains peut s'entendre comme donnant la répartition des formes spectrales sur les différentes ondes. On voit qu'en plus de conduire à une paramétrisation intuitive, le modèle NTF permet de réduire le nombre de paramètres d'un PGLS de JFN à

$$\#\theta_{NTF} = K(J + F + N),$$

ce qui constitue un gain considérable.

Comme on l'a vu en section 4.1, l'apprentissage d'un modèle de DSP pour un PGLS se fait en minimisant la divergence d'Itakura-Saito entre le spectrogramme v des observations et $\mathcal{P}^{NTF}$. On a ainsi :

$$\theta^\star_{NTF} = \operatorname*{argmin}_\theta \sum_{\boldsymbol{f}, \boldsymbol{n}, j} d_0\left(v(\boldsymbol{f}, \boldsymbol{n}, j) \mid \mathcal{P}^{NTF}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta)\right). \tag{4.3.3}$$

Dans le cas où $D = 1$, c'est-à-dire où les signaux considérés sont des séries temporelles, le modèle 4.3.2 est équivalent à celui qu'on retrouve dans la littérature [75, 71, 34, 155, 69] sur le

traitement du signal audio [8] et on retrouve les résultats déjà donnés dans [69, 19] que l'utilisation de la divergence d'Itakura-Saito d_0 et du spectrogramme de puissance v pour l'apprentissage d'un modèle NTF correspondent à l'apprentissage au maximum de vraisemblance des paramètres d'un PGLS. En figure 4.2, j'ai représenté un exemple de décomposition d'un spectrogramme d'enregistrement audio ($J = 1$) par le modèle NMF. On trouvera un autre exemple d'utilisation du modèle NTF ($J = 3$) en figure 10.3 page 139.

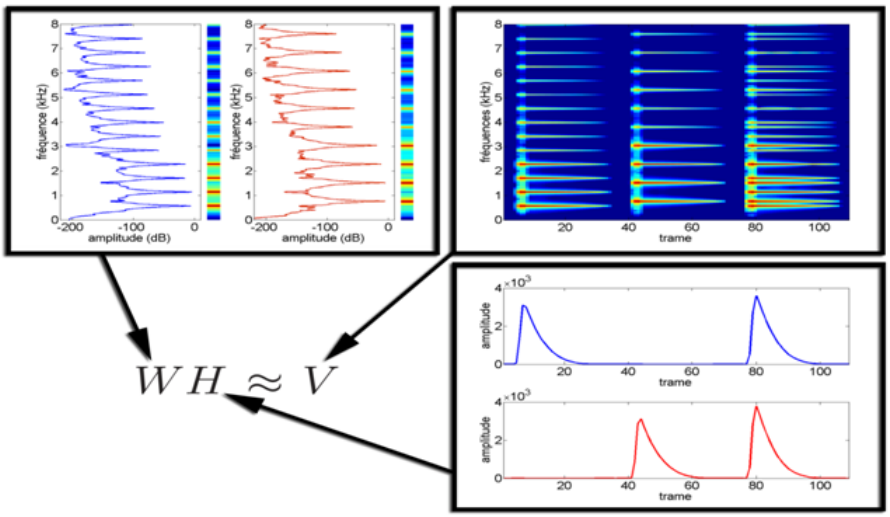

Figure 4.2: Illustration du modèle NMF. L'apprentissage du modèle se fait par décomposition du spectrogramme v des observations en un produit WH de matrices non négatives. Ici, un enregistrement de piano comporte deux notes, jouées alternativement puis simultanément. Un modèle NMF peut décomposer ce spectrogramme comme le produit d'une matrice de bases spectrales W par leurs activations H. (d'après Romain Hennequin [100])

D'autres configurations alternatives à 4.3.3 sont utilisées dans l'état de l'art pour la modélisation audio par factorisation non-négative. En particulier, de nombreuses études [195, 196, 187] font intervenir la divergence de Kullback-Leibler au lieu de celle d'Itakura-Saito et le module de la STFT plutôt que son énergie. Cependant, il n'existe pas de modèle correspondant connu à ce jour sur les formes d'ondes.
Ainsi, ces configurations ne peuvent être aujourd'hui entendues que comme des modèles d'images, et l'étape de filtrage par laquelle ces études récupèrent des formes d'onde, similaire au filtrage de Wiener généralisé 3.3.7 n'est pas justifiée théoriquement.
Il est cependant clair que leurs performances sont souvent comparables, voire meilleures, à celles obtenues par l'utilisation de la divergence d'Itakura-Saito et du spectrogramme de puissance. Mon expérience montre en effet que le choix de la divergence à utiliser où la puissance à laquelle il faut élever le module de la STFT n'a pas une importance fondamentale sur la qualité des résultats obtenus. Cependant, aucune explication théorique à ce phénomène n'est encore disponible.

4.3.2 Apprentissage des paramètres

Dans le cas du modèle NTF, l'apprentissage des paramètres optimaux $\theta^\star_{NTF}$ est plus délicat que pour le modèle CI. En effet, sauf situation exceptionnelle, aucun lot θ de paramètres [9] ne permettra

8. Je suis cependant, à ma connaissance, le premier à l'avoir exprimé pour J quelconque dans le cadre du traitement du signal audio ($D = 1$) dans [130, 137]. La plupart des études se sont auparavant concentrées sur le cas $J = 1$ (mono) ou $J = 2$ (stéréo).

9. Le modèle NTF peut devenir exact si on a affaire à des observations $\boldsymbol{v}$ de synthèse, ou bien si on considère $K = \min(F, N)$. Aucun de ces deux cas n'a de réel intérêt pratique.

d'obtenir exactement $\boldsymbol{v} = \mathcal{P}(\cdot \mid \theta_{NTF})$, contrairement au modèle CI. Autrement dit, un modèle NTF est nécessairement une *approximation*. Pour commencer, considérons données les TFCT $\boldsymbol{s}$ d'un ensemble $\tilde{\boldsymbol{s}}$ de J formes d'ondes, et leurs spectrogrammes $\boldsymbol{v}$:

$$\forall (\boldsymbol{f}, \boldsymbol{n}, j)\,,\ \boldsymbol{v}(\boldsymbol{f}, \boldsymbol{n}, j) = |\boldsymbol{s}(\boldsymbol{f}, \boldsymbol{n}, j)|^2\,.$$

La recherche des paramètres NTF $\theta^\star_{NTF}$ qui maximisent la vraisemblance des observations est donnée par 4.1.3, qui devient dans ce cas :

$$\begin{aligned}\theta^\star_{NTF} &= \underset{\theta}{\operatorname{argmin}}\ \mathcal{L}(\tilde{\boldsymbol{s}} \mid \theta) \\ &= \underset{\theta}{\operatorname{argmin}} \sum_{(\boldsymbol{f},\boldsymbol{n},j)} d_0\left(v(\boldsymbol{f}, \boldsymbol{n}, j) \mid \sum_{k=1}^{K} W(\boldsymbol{f}, k)\, H(\boldsymbol{n}, k)\, Q(j, k)\right). \qquad (4.3.4)\end{aligned}$$

Comme on le voit, l'apprentissage d'un lot de paramètres NTF qui maximise la vraisemblance des observations dans un modèle PGLS devient la recherche 4.3.4 des coefficients NTF $\theta^\star_{NTF} = \{W, H, Q\}$ qui expliquent au mieux le spectrogramme $\boldsymbol{v}$ des observations, lorsque la fonction de coût utilisée est la divergence d'Itakura-Saito. Ce problème a donné lieu à de très nombreuses études [69, 19, 155, 163, 75, 71, 34] et il existe aujourd'hui des algorithmes efficaces pour la recherche de $\theta^\star_{NTF}$. La plupart de ces études sont focalisées sur le cas $J = 1$ ou $J = 2$ et ne s'intéressent toutes qu'au cas $D = 1$. J'ai été le premier à ma connaissance à étendre ces algorithmes au cas de J quelconque (dans [130, 137]) et de D quelconque (ce texte). Cependant, de telles extensions sont immédiates et font intervenir exactement les mêmes principes que ceux présentés dans la littérature.

Historiquement, les considérations probabilistes menant à 4.3.4 ont attendu les travaux de C. Févotte, N. Bertin et J.L. Durrieu [69, 19]. Auparavant, la NMF/NTF était déjà utilisée dans le cadre de l'analyse de signaux audio mais se justifiait par des considérations heuristiques.

Le principe des algorithmes minimisant 4.3.4 est d'identifier les paramètres $\theta^\star_{NTF}$ par une procédure itérative. Puisque la minimisation conjointe de 4.3.4 en fonction de $\{W, H, Q\}$ est très complexe, la stratégie adoptée est d'en effectuer alternativement une minimisation par rapport à chacun des paramètres W, H, puis Q, les autres restants fixés, et d'itérer cette procédure jusqu'à convergence. Plusieurs types de critères d'arrêt peuvent être considérés mais il est classique de simplement effectuer un nombre fixé d'itérations.

La difficulté principale de l'optimisation d'un modèle NTF et sa distinction avec le modèle PARAFAC [96] réside dans la contrainte de n'avoir que des entrées non-négatives pour les paramètres W, H et Q. En effet, si on procédait à une descente de gradient classique [21], la stratégie pour mettre un jour un des paramètres θ_p (par exemple une des entrées de W) consisterait à choisir :

$$\theta_p \leftarrow \theta_p - \alpha \frac{\partial \mathcal{L}(\tilde{\boldsymbol{s}} \mid \theta)}{\partial \theta_p}, \qquad (4.3.5)$$

où $\alpha > 0$ serait un paramètre de *pas*, dont la valeur peut être fixée *a priori* ou bien dépendante de l'itération selon une loi empirique ou encore être fonction de la dérivée seconde de $\mathcal{L}(\tilde{\boldsymbol{s}} \mid \theta)$ si cette dernière est disponible. Dans tous les cas, une application naïve de 4.3.5 avec un paramètre α fixé peut conduire à des valeurs négatives pour θ_p, ce qui ne respecte pas la contrainte de non-négativité. Une idée originale proposée par Lee et Seung dans leur article [128] est de ne pas effectuer une telle mise à jour additive, mais de privilégier plutôt une mise à jour *multiplicative*, dont le principe est le suivant.

Considérons une fonction $f(x)$ à minimiser, définie sur $\mathbb{R}$. Supposons que cette fonction soit dérivable et qu'on connaisse l'expression de sa dérivée $f'(x)$. La principale astuce des algorithmes d'optimisation non négative repose sur une décomposition particulière de f', lorsque c'est possible, comme la différence de deux termes positifs :

$$\forall x \in \mathbb{R},\ \begin{cases} f'(x) & = G^+(x) - G^-(x) \\ G^+(x) & > 0 \\ G^-(x) & > 0 \end{cases}. \qquad (4.3.6)$$

La mise à jour du paramètre x suggérée par [128] consiste alors à choisir :

$$\forall x,\ x \leftarrow x\frac{G^{-}(x)}{G^{+}(x)}. \tag{4.3.7}$$

Ce procédé est illustré en figure 4.3 pour $f(x)$ parabolique.

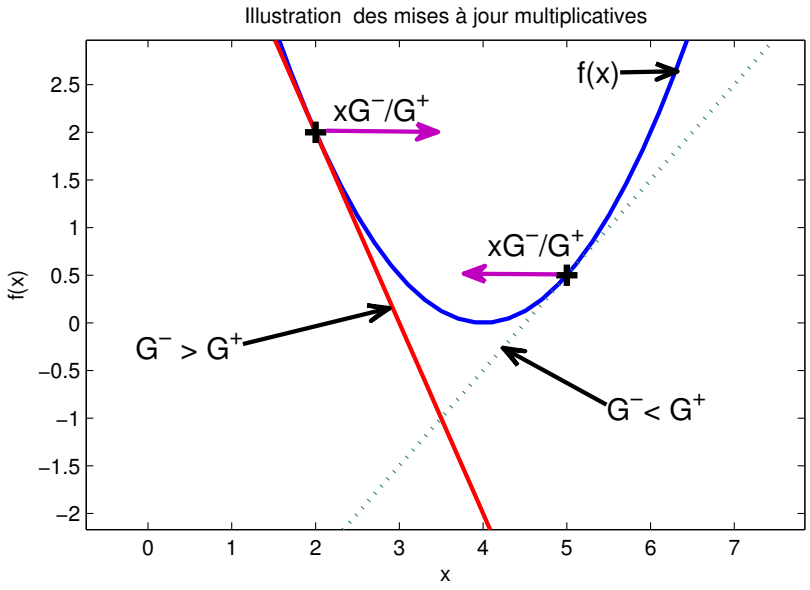

FIGURE 4.3: Illustration du comportement des mises à jour multiplicatives pour la recherche du point minimum de $f(x)$. On suppose que la dérivée $f'(x)$ est donnée par $f'(x) = G^{+}(x) - G^{-}(x)$ avec $G^{+}(x)$ et $G^{-}(x)$ positifs et connus. Dans tous les cas, la mise à jour multiplicative $x \leftarrow x\frac{G^{-}(x)}{G^{+}(x)}$ rapproche le point x du minimum de f.

Une des propriétés intéressantes de cette méthode de mise à jour est qu'elle garantie la non-négativité de l'estimée, pour peu que le choix initial de x soit positif. En effet, le terme $\frac{G^{-}(x)}{G^{+}(x)}$ par lequel x est multiplié est positif.

Les mises à jour multiplicatives de modèles non-négatifs minimisant la divergence d'ITAKURA-SAITO convergent rapidement et sont faciles à implémenter. Leurs propriétés de convergence ont été établies récemment.

Bien entendu, la question se pose de la convergence de cette technique d'optimisation en fonction de la nature de la fonction f. De nombreuses études ont porté sur cette question dans le cas des fonctions de coût usuelles et la décroissance du critère fut en premier établie [123] pour une large famille de fonctions de coût convexes [10]. La question est cependant demeurée longtemps ouverte dans le cas de la divergence d'ITAKURA-SAITO d_0. Si toutes les études empiriques concluaient à la validité de la procédure, il a fallu attendre des études très récentes pour démontrer que les mises à jour multiplicatives conduisent à la décroissance du critère ([148, 12] et surtout [71]). En substance, on peut montrer que si f est la divergence d'ITAKURA-SAITO [11], la mise à jour multiplicative 4.3.7 correspond à l'application d'une procédure de *maximisation-égalisation*. Dans tous les cas, ce que démontrent ces études n'est pas la convergence des mises à jour multiplicatives vers un minimum global, mais la seule décroissance du critère. Le problème se pose ainsi de l'existence de minima locaux, mais en pratique, ces méthodes conduisent à de très bonnes estimées.

10. Les $\beta-$divergences pour $\beta \in [1\,2]$.
11. Ces résultats s'appliquent plus généralement à n'importe quelle $\beta-$divergence.

Je vais à présent développer l'application de ce schéma d'optimisation au cas de la fonction de coût 4.3.4, donc pour J et D quelconques. Je vais par exemple considérer l'exemple de la mise à jour d'un des coefficients $W(\boldsymbol{f}_0, k_0)$ des paramètres NTF. Le cas des éléments de H et Q est identique. Pour commencer, on peut montrer que :

$$\frac{\partial d_0(x \mid y)}{\partial y} = y^{-2}(y - x). \tag{4.3.8}$$

Si on se rappelle que $\theta = \{W, H, Q\}$ et que la notation $\mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta)$ désigne

$$\mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta) = \sum_{k=1}^{K} W(\boldsymbol{f}, k) H(\boldsymbol{n}, k) Q(j, k),$$

on a :

$$\begin{aligned}\frac{\partial \mathcal{L}(\tilde{\boldsymbol{s}} \mid \theta)}{\partial W(\boldsymbol{f}_0, k_0)} &= \sum_{\boldsymbol{f}, \boldsymbol{n}, j} \frac{\partial d_0(v(\boldsymbol{f}, \boldsymbol{n}, j) \mid \mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta))}{\partial \mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta)} \frac{\partial \mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta)}{W(\boldsymbol{f}_0, k_0)} \\ &= \sum_{\boldsymbol{n}, j} \mathcal{P}(\boldsymbol{f}_0, \boldsymbol{n}, j \mid \theta)^{-2} (\mathcal{P}(\boldsymbol{f}_0, \boldsymbol{n}, j \mid \theta) - v(\boldsymbol{f}_0, \boldsymbol{n}, j)) H(\boldsymbol{n}, k_0) Q(j, k_0),\end{aligned}$$

dont les termes peuvent être regroupés de manière à obtenir une décomposition du type de 4.3.6 :

$$\frac{\partial \mathcal{L}(\tilde{\boldsymbol{s}} \mid \theta)}{\partial W(\boldsymbol{f}_0, k_0)} = \underbrace{\sum_{\boldsymbol{n}, j} \mathcal{P}(\boldsymbol{f}_0, \boldsymbol{n}, j \mid \theta)^{-1} H(\boldsymbol{n}, k_0) Q(j, k_0)}_{G^+(W(\boldsymbol{f}_0, k_0))} - \underbrace{\sum_{\boldsymbol{n}, j} \mathcal{P}(\boldsymbol{f}_0, \boldsymbol{n}, j \mid \theta)^{-2} v(\boldsymbol{f}_0, \boldsymbol{n}, j) H(\boldsymbol{n}, k_0) Q(j, k_0)}_{G^-(W(\boldsymbol{f}_0, k_0))}.$$

Dans ces conditions, l'application du procédé de mise à jour multiplicative de $W(\boldsymbol{f}_0, k_0)$ peut se faire par application directe de 4.3.7, de manière à obtenir :

$$W(\boldsymbol{f}_0, k_0) \leftarrow W(\boldsymbol{f}_0, k_0) \frac{\sum_{\boldsymbol{n}, j} \mathcal{P}(\boldsymbol{f}_0, \boldsymbol{n}, j \mid \theta)^{-2} v(\boldsymbol{f}_0, \boldsymbol{n}, j) H(\boldsymbol{n}, k_0) Q(j, k_0)}{\sum_{\boldsymbol{n}, j} \mathcal{P}(\boldsymbol{f}_0, \boldsymbol{n}, j \mid \theta)^{-1} H(\boldsymbol{n}, k_0) Q(j, k_0)}.$$

L'algorithme 4.1 donne la procédure complète d'estimation des paramètres NTF à partir de l'observation de J formes d'onde régulièrement échantillonnées. Il est possible d'implémenter efficacement ces opérations avec des multiplications matricielles, que j'ai données en détail dans [137]. On peut de plus considérer des implémentations de cet algorithme sur des systèmes parallèles et de nombreuses techniques existent qui permettent de l'utiliser en pratique sur des signaux d'une taille considérable. Une bonne référence sur ce sujet est [34]. J'ai de plus diffusé des implémentations de cet algorithme pour $D = 1$ en PYTHON (Licence LGPL) et en MATLAB (licence BSD).

Algorithme 4.1 Optimisation des paramètres du modèle NTF pour J et D quelconques. Dans ces mises à jour, $\mathcal{P}(\boldsymbol{f}, \boldsymbol{n}, j \mid \theta)$ utilise toujours les dernières versions de W, H et Q disponibles.

Entrées :
- J signaux régulièrement échantillonnés $\tilde{\boldsymbol{s}}$
- Paramètres $\boldsymbol{\rho}$ et $\mathbb{T}_0$ de tramage
- Nombre K de composantes NTF
- Nombre N_I d'itérations

Initialisation
- Calculer la TFCT $\boldsymbol{s}$ des J signaux $\tilde{\boldsymbol{s}}$
- Calculer le spectrogramme $\boldsymbol{v}$ des signaux :

$$\forall (\boldsymbol{f}, \boldsymbol{n}, j)\,,\ \boldsymbol{v}(\boldsymbol{f}, \boldsymbol{n}, j) = |\boldsymbol{s}(\boldsymbol{f}, \boldsymbol{n}, j)|^2$$

- Choisir une valeur initiale aléatoire non négative pour tous les paramètres dont on ne fournit pas une valeur initiale.

Répéter N_I fois
- $\forall (\boldsymbol{f}_0, k_0)\,,\ W(\boldsymbol{f}_0, k_0) \leftarrow W(\boldsymbol{f}_0, k_0)\, \frac{\sum_{\boldsymbol{n},j} \mathcal{P}(\boldsymbol{f}_0,\boldsymbol{n},j|\theta)^{-2} v(\boldsymbol{f}_0,\boldsymbol{n},j) H(\boldsymbol{n},k_0) Q(j,k_0)}{\sum_{\boldsymbol{n},j} \mathcal{P}(\boldsymbol{f}_0,\boldsymbol{n},j|\theta)^{-1} H(\boldsymbol{n},k_0) Q(j,k_0)}$
- $\forall (\boldsymbol{n}_0, k_0)\,,\ H(\boldsymbol{n}_0, k_0) \leftarrow H(\boldsymbol{n}_0, k_0)\, \frac{\sum_{\boldsymbol{f},j} \mathcal{P}(\boldsymbol{f},\boldsymbol{n}_0,j|\theta)^{-2} v(\boldsymbol{f},\boldsymbol{n}_0,j) W(\boldsymbol{f},k_0) Q(j,k_0)}{\sum_{\boldsymbol{f},j} \mathcal{P}(\boldsymbol{f},\boldsymbol{n}_0,j|\theta)^{-1} W(\boldsymbol{f},k_0) Q(j,k_0)}$
- $\forall (j, k)\,,\ Q(j_0, k_0) \leftarrow Q(j_0, k_0)\, \frac{\sum_{\boldsymbol{f},\boldsymbol{n}} W(\boldsymbol{f},k_0) H(\boldsymbol{n},k_0) \mathcal{P}(\boldsymbol{f},\boldsymbol{n},j_0|\theta)^{-2} v(\boldsymbol{f},\boldsymbol{n},j_0)}{\sum_{\boldsymbol{f},\boldsymbol{n}} W(\boldsymbol{f},k_0) H(\boldsymbol{n},k_0) \mathcal{P}(\boldsymbol{f},\boldsymbol{n},j_0|\theta)^{-1}}$

Sortie
- Retourner Q, W, H

Conclusion de la première partie

Dans la première partie de cet exposé, j'ai présenté les processus gaussiens. Un processus gaussien est un espace probabiliste de fonctions définies sur un *domaine de définition* $\mathbb{T}$ quelconque et à valeurs dans $\mathbb{C}$. J'ai rappelé qu'un *processus gaussien* est la généralisation au cas des fonctions de la *distribution* gaussienne, qui porte sur des vecteurs. Le principal intérêt de ce modèle est qu'il ne contraint pas le signal à appartenir à une certaine famille paramétrique prédéterminée comme celle des polynômes. Au lieu de cela, il est défini par ses fonctions de *moyenne* et de *covariance*, qui viennent caractériser l'allure de ses réalisations, sans leur imposer une forme fixe. De manière à illustrer la vaste étendue des types de signaux que les processus gaussiens permettent de modéliser, j'ai considéré de nombreux exemples de fonctions de covariance et j'ai discuté de l'influence de leurs *hyperparamètres*. Si la fonction de covariance d'un processus gaussien est stationnaire, j'ai rappelé que la transformée de Fourier de ses réalisations a la propriété d'avoir ses différents éléments indépendants et distribués selon une loi gaussienne dont la variance porte le nom de *densité spectrale de puissance* (DSP) du processus correspondant.

Parmi les résultats présentés dans cette partie, le matériel général sur les processus gaussiens, leur utilisation pour la régression et l'apprentissage de leurs paramètres sont largement établis. L'extension des propriétés des processus stationnaires au cas de D quelconque, quoique déjà établie, n'est pas couramment présentée dans la littérature.

J'ai de plus montré comment les processus gaussiens peuvent être utilisés efficacement pour la régression en section 2.2.3. Dans cette première présentation, aucune hypothèse particulière n'a été faite sur la nature du domaine de définition $\mathbb{T}$ du processus, ce qui en fait un modèle particulièrement général pour aborder ce genre de problématique. Je suis alors revenu sur ce problème plus tard en section 2.5.4 dans le cas où $\mathbb{T}$ est un groupe, où la fonction de covariance considérée est stationnaire et où les signaux sont régulièrement échantillonnés. Les calculs bénéficient alors de simplifications importantes qui les rendent particulièrement attractifs pour le lissage et la régression. Dans ce cas, la régression prend le nom de filtrage optimal, ou filtrage de Wiener. On a vu de plus que ces résultats se généralisent très bien au cas d'un domaine de définition $\mathbb{T} = \mathbb{Z}^D$ de dimension quelconque et ne sont pas limités au cas classique des séries temporelles régulièrement échantillonnées ($D = 1$).

J'ai présenté le tramage et l'hypothèse locale, classiques en audio, comme une technique générale d'approximation dans des modèles de processus gaussiens.

Malgré leur intérêt conceptuel et la facilité avec laquelle ils permettent d'apporter une solution à un problème de régression, les processus gaussiens sont pénalisés par la lourde complexité calculatoire des procédures d'inférence correspondantes, qui nécessitent l'inversion d'une matrice $L \times L$, où L est le nombre d'observations. Dans de nombreux cas, cette complexité est prohibitive. Dans le but de simplifier les calculs, de nombreuses approximations ont été proposées dans la littérature. C'est ainsi que j'en ai présenté certaines, des plus influentes, dans le chapitre 3, qui peuvent souvent se comprendre comme simplifiant le réseau dense des dépendances considérées par le modèle gaussien. Pour ce faire, elles sélectionnent un certain nombre de *points supports,* seuls points par lesquels transite l'information entre données d'observations et de test.

C'est dans ce contexte que j'ai étendu la technique du tramage, courante en traitement des séries temporelles, au cas général d'un domaine de définition Euclidien de dimension quelconque. Le tramage revient à découper le signal en différentes zones géographiques appelées *trames*, qui ont

entre elles un certain *recouvrement*. Le signal original peut être reconstruit aux moindres carrés à partir d'un tramage par une opération *d'addition-recouvrement*.

Une fois le tramage défini, j'ai introduit *l'hypothèse locale*, qui consiste à ignorer les dépendances entre trames. J'ai montré que cette hypothèse a plusieurs avantages. Tout d'abord, elle conduit à de très importantes simplifications des calculs, puisqu'il suffit de procéder indépendamment à l'inférence dans chacune des trames, avant de reconstruire le signal résultant par addition-recouvrement. Un autre avantage de la méthode est que si les fenêtres de pondération utilisées pour le tramage sont lisses, le signal reconstruit garde les propriétés de régularité de chacune de ses trames. En pratique, l'hypothèse locale permet d'obtenir des estimées lisses, ce qui est crucial en audio. En contrepartie de ces avantages, j'ai montré qu'elle produit une sous-estimation de la variance *a posteriori*.

C'est alors que j'ai défini les processus gaussiens localement stationnaires (PGLS) comme des processus gaussiens à trames indépendantes dont chacune des trames est stationnaire et centrée. Si le signal est régulièrement échantillonné, on peut définir sa Transformée de Fourier à Court Terme (TFCT), qui correspond à l'ensemble des transformées de Fourier des différentes trames. Cette TFCT a ainsi été définie pour une dimension D quelconque du domaine de définition, avec des notations similaires à celles usuelles en audio. La seule différence du cas général est que les indices de fréquence et de trame sont des vecteurs de dimension $D \times 1$. Quoiqu'il en soit, une propriété cruciale des PGLS est que tous les éléments de leur TFCT peuvent être considérés comme indépendants [12]. En pratique, cela signifie qu'il est possible de procéder à l'inférence sur des PGLS pour chacun de ces points indépendamment des autres. Ce modèle permet de rendre compte efficacement de signaux qui peuvent être considérés comme localement stationnaires, mais qui sont non-stationnaires globalement. Si ce modèle est classique en traitement du signal audio, son extension au cas général d'un domaine de définition $\mathbb{T} = \mathbb{Z}^D$ de dimension quelconque est originale.

La plupart des études mettant en œuvre les processus gaussiens pour la séparation de source se focalisent sur les séries temporelles et commencent d'emblée par poser le modèle PGLS. Il n'est qu'un cas particulier — mais important — du formalisme proposé.

Une fois le modèle PGLS établi, j'ai souligné le fait qu'il est paramétré par la DSP de chacune de ses trames, qui donne la variance de chacun des éléments de sa TFCT. En pratique, ce nombre de paramètres est considérable et il est nécessaire pour les applications de réduire leur nombre. Dans ce but, on considère des *modèles* de DSP, dont le principe est d'approximer la DSP du processus par une forme paramétrique plus simple. J'ai montré en toute généralité comment les paramètres de tels modèles sont appris à partir d'observations.

J'ai alors présenté la factorisation en tenseurs non-négatifs (NTF) qui permet de réduire d'un ordre de grandeur le nombre de paramètres de la DSP, ainsi que le modèle par Compression d'Image (CI), qui consiste à modéliser la DSP d'un PGLS comme une image à compresser. Ma présentation du modèle NTF généralise naturellement sa restriction classique au cas des séries temporelles. J'ai indiqué l'algorithme permettant d'apprendre les paramètres de tels modèles.

Telle que je l'ai introduite, la NTF et l'algorithme d'apprentissage correspondant fonctionnent en dimension quelconque. Cette extension s'est faite très naturellement.

Pour compléter cette partie de présentation des processus gaussiens, on trouvera en annexe A la procédure permettant de générer des réalisations d'un processus gaussien dont on connaît les fonctions de moyenne et de covariance. Cette procédure de synthèse est celle que j'ai utilisée tout au long de mon exposé pour présenter mes exemples. On peut remarquer que les propriétés spectrales des processus gaussiens stationnaires m'ont permis de produire presque instantanément des réalisations sur le plan, là où une synthèse naïve aurait nécessité des calculs d'une complexité prohibitive.

12. Pour peu que les trames soient suffisamment grandes.

Deuxième partie

Séparation de processus gaussiens

Introduction

Cette partie est consacrée à l'utilisation des processus gaussiens comme des modèles performants pour aborder le problème de la séparation de sources, que j'ai présenté en section 1.2. La principale contribution de mon travail a été d'aborder la question en modélisant les sources comme la réalisation de J processus gaussiens indépendants. De nombreuses techniques de l'état de l'art [17, 30, 50, 53, 155, 163] peuvent elles aussi être comprises comme plaçant sur les sources un *a priori* gaussien. Cependant, ces méthodes se focalisent sur le seul cas des séries temporelles et sont restreintes aux processus gaussiens localement stationnaires, vus en section 3.3. Dans cette partie, je me placerai dans le cas général où les sources sont des processus gaussiens indépendants définis sur des espaces quelconques, ce qui aura pour conséquence d'étendre significativement la portée du problème considéré.

J'ai choisi de structurer ma présentation en fonction de la nature du procédé de mixage, c'est-à-dire en fonction de la manière dont les mélanges sont obtenus à partir des sources. Dans un premier temps, je ne considèrerai dans le chapitre 5 que le cas où les mélanges sont des combinaisons linéaires des sources. J'étendrai la discussion aux cas plus complexes des mélanges convolutifs et diffus au chapitre 6.

Il est clair que l'ensemble de cette discussion contraste nettement avec les approches traditionnelles usitées pour la séparation aveugle de sources évoquées en section 1.2, comme l'Analyse en Composantes Indépendantes [109, 38], où les connaissances disponibles sur les signaux à séparer sont très parcellaires, pour ne pas dire inexistantes. En effet, je commencerai par supposer dans les chapitres 5 et 6 que les fonctions de moyenne et de covariance de toutes les sources sont connues. Si cette hypothèse de *séparation informée* peut sembler extrêmement forte, on verra dans les parties III et IV qu'elle correspond à des scénarios applicatifs bien réels.

Cependant, j'aborderai la question de l'estimation de ces paramètres à partir de la seule observation des mélanges au chapitre 7. En effet, si la valeur précise des fonctions de covariance des sources peut être inconnue, il est néanmoins courant de pouvoir supposer qu'elles appartiennent à une certaine famille paramétrique comme celles évoquées plus haut en sections 2.3 page 28 et 4.3 page 61. Dans ce cas, je parlerai de *séparation semi-informée* et il est possible d'estimer les représentants de ces familles qui sont les mieux susceptibles d'expliquer l'observation des seuls mélanges et d'ainsi procéder à la séparation. Cette approche est d'ailleurs celle employée par la plupart des études récentes en séparation de sources audio.

Pour finir, je considèrerai brièvement deux applications du formalisme proposé pour la séparation semi-informée au chapitre 8. La première sera la séparation des rythmiques d'enregistrements musicaux, tandis que la deuxième consistera à décomposer des données multidimensionnelles de mouvements comme une somme de composantes élémentaires.

Chapitre 5

Mélanges linéaires instantanés

5.1 Un seul mélange linéaire instantané ($I = 1$)

Soit $\mathbb{T}$ un ensemble quelconque et $\tilde{s}$ un groupe de J processus gaussiens indépendants définis sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$, appelés *sources*. De la même manière que dans les chapitres précédents, $\tilde{s}(\cdot, j)$ désigne la $j^{\text{ème}}$ source. Dans tout ce chapitre, je suppose connues les fonctions de covariance k_j des J sources, ainsi que leurs fonctions moyenne μ_j. Soit $T = [t_1, \ldots, t_L]$ un ensemble de L positions dans $\mathbb{T}$. Je suppose de plus que ce ne sont pas les sources qui sont observées, mais plutôt leur somme

$$\tilde{x} = \sum_{j=1}^{J} \tilde{s}(\cdot, j). \tag{5.1.1}$$

Par conséquent, l'observation

$$\tilde{\boldsymbol{x}}(T) = [\tilde{x}(t_1), \ldots, x(t_L)]^\top$$

est constituée de la valeur de cette somme en chacun des points de T. Le signal $\tilde{x}$ est aussi appelé le *mélange*. Du fait de la simplicité du modèle de mixage 5.1.1, on parle de mélange linéaire instantané. Puisque $\tilde{x}$ est une somme de processus gaussiens, il est lui même un processus gaussien. De plus, sa fonction moyenne et sa fonction de covariance sont les sommes de celles des sources. En effet, celles-ci sont indépendantes et s'ajoutent simplement pour former le mélange :

$$\tilde{x} \sim \mathcal{PG}\left(\sum_{j=1}^{J} \mu_j, \sum_{j=1}^{J} k_j\right).$$

Un problème de séparation de sources est caractérisé par le fait que l'objectif est de récupérer les sources $\tilde{s}$ à partir de l'observation de $\tilde{\boldsymbol{x}}(T)$. En toute généralité, on peut vouloir récupérer la valeur des sources en d'autres points $T' = [t'_1, \ldots, t'_{L'}]$ de $\mathbb{T}$ que ceux où leur mélange est observé. La méthodologie pour résoudre ce genre de problème dans un contexte gaussien est toujours la même que celle présentée aux sections 2.2.3, 2.5.2 et 2.5.4.

– Tout d'abord, on écrit la distribution jointe des sources en T' et des observations en T :

$$\begin{bmatrix} \tilde{\boldsymbol{s}}(T', 1) \\ \vdots \\ \tilde{\boldsymbol{s}}(T', J) \\ \tilde{\boldsymbol{x}}(T) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \tilde{\boldsymbol{\mu}}_1(T') \\ \vdots \\ \tilde{\boldsymbol{\mu}}_J(T') \\ \sum_{j=1}^{J} \tilde{\boldsymbol{\mu}}_j(T) \end{bmatrix}, \begin{bmatrix} K_1(T', T') & 0 & 0 & K_1(T', T) \\ 0 & \ddots & 0 & \vdots \\ 0 & 0 & K_J(T', T') & K_J(T', T) \\ K_1(T, T') & \cdots & K_J(T, T') & \sum_{j=1}^{J} K_j(T, T) \end{bmatrix}\right), \tag{5.1.2}$$

où $[K_j(T, T')]_{l,l'} = k_j(t_l, t'_{l'})$ est une matrice de dimension $L \times L'$ et donne la covariance de la source j entre les positions de T et de T'.

– Ensuite, on obtient la distribution *a posteriori* des sources étant donné le mélange en utilisant simplement les résultats classiques des distributions gaussiennes, exposés en section 2.2.1 :

$$\begin{bmatrix} \tilde{\boldsymbol{s}}\left(T',1\right) \\ \vdots \\ \tilde{\boldsymbol{s}}\left(T',J\right) \end{bmatrix} \mid \tilde{\boldsymbol{x}}\left(T\right) \sim \mathcal{N}\left(\boldsymbol{\mu}_{\text{post}},\, K_{\text{post}}\right), \tag{5.1.3}$$

avec :

$$\boldsymbol{\mu}_{\text{post}} = \begin{bmatrix} \tilde{\boldsymbol{\mu}}_1\left(T'\right) \\ \vdots \\ \tilde{\boldsymbol{\mu}}_J\left(T'\right) \end{bmatrix} + \begin{bmatrix} K_1\left(T',T\right) \\ \vdots \\ K_J\left(T',T\right) \end{bmatrix} \left(\sum_{j=1}^{J} K_j\left(T,T\right)\right)^{-1} \left(\tilde{\boldsymbol{x}}\left(T\right) - \sum_{j=1}^{J} \tilde{\boldsymbol{\mu}}_j\left(T\right)\right) \tag{5.1.4}$$

et

$$\begin{aligned} K_{\text{post}} = &\begin{bmatrix} K_1\left(T',T'\right) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & K_J\left(T',T'\right) \end{bmatrix} \\ &- \begin{bmatrix} K_1\left(T',T\right) \\ \vdots \\ K_J\left(T',T\right) \end{bmatrix} \left(\sum_{j=1}^{J} K_j\left(T,T\right)\right)^{-1} \begin{bmatrix} K_1\left(T,T'\right) & \cdots & K_J\left(T,T'\right) \end{bmatrix} \end{aligned} \tag{5.1.5}$$

– Dans la mesure où cette distribution *a posteriori* est gaussienne, les valeurs $\widehat{\tilde{\boldsymbol{s}}\left(T',j\right)}$ des sources estimées qui minimisent l'erreur quadratique moyenne sont données simplement par $\boldsymbol{\mu}_{\text{post}}$ dans 5.1.4.

Si les expressions 5.1.4 et 5.1.5 peuvent paraître déroutantes, il n'en demeure pas moins qu'elles ne font intervenir que des opérations matricielles très simples à implémenter et que tous les paramètres de la séparation sont donnés par les fonctions moyennes μ_j et les fonctions de covariance k_j des sources, sur lesquelles je n'ai fait aucune hypothèse particulière. Comme on le voit, cette procédure de séparation de sources est très générale et ne suppose même pas une structure particulière pour le domaine de définition $\mathbb{T}$ considéré. Par conséquent, le choix des processus gaussiens pour la modélisation des sources permet d'effectuer la séparation de fonctions additives définies sur des domaines de définition quelconques.

Alors que les approches traditionnelles pour la séparation de sources, de type ACI, permettent de séparer des signaux dont on n'a aucune connaissance *a priori*, le modèle gaussien permet très simplement de séparer des signaux dont les fonctions de covariance, ou du moins la famille auxquelles elles appartiennent, sont connues.

J'ai illustré ce cas simple de séparation de sources en figure 5.1 pour la séparation du mélange de deux séries temporelles ($\mathbb{T} = \mathbb{R}$). Leurs fonctions de covariance sont des cas particuliers de celles présentées en section 2.3 page 28. Dans cet exemple, j'ai illustré le cas où les sources sont estimées sur les mêmes points que les observations, i.e. le cas $T' = T$. On constate que les estimées sont très fidèles aux sources originales.

Il est remarquable que la méthode proposée permette d'inclure une large variété de connaissances *a priori* sur les signaux séparés. En effet, chaque source est caractérisée par ses fonctions de moyenne et de covariance, ce qui laisse une marge considérable à l'utilisateur de cette technique pour définir la nature des signaux à séparer. Évidemment, le calcul 5.1.4 fait intervenir l'inversion d'une matrice de dimension $L \times L$, ce qui peut être prohibitif dans de nombreux domaines applicatifs. Cependant, toutes les méthodes d'approximation vues au chapitre 3 peuvent être utilisées pour simplifier cette opération. Je montrerai en particulier en section 5.3 comment les calculs sont simplifiés si les sources sont des PGLS.

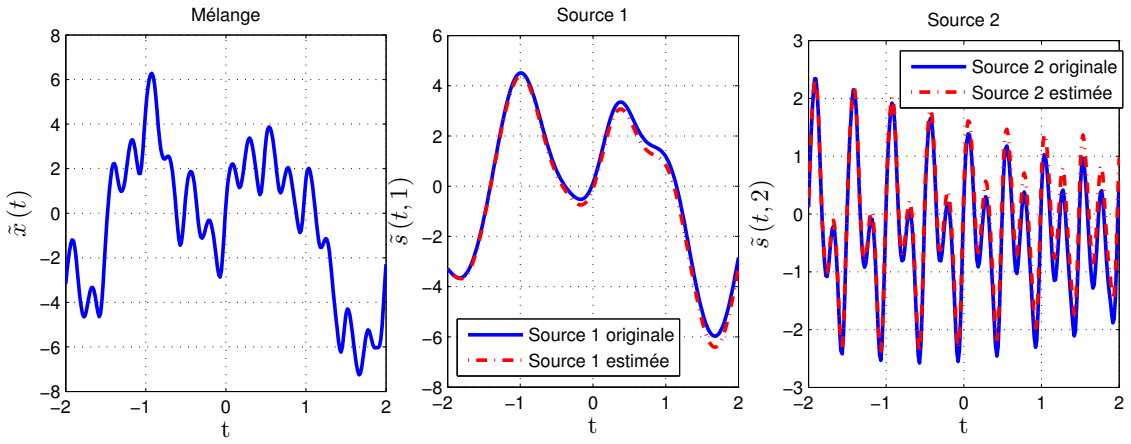

FIGURE 5.1: Séparation du mélange linéaire instantané de deux processus gaussiens indépendants. Les deux processus sont centrés et leurs fonctions de covariance k_1 et k_2 sont connues. k_1 est la fonction EC 2.2.13 page 26, tandis que k_2 est la fonction de covariance pseudo-périodique 2.3.5 page 33. J'ai généré une réalisation de chacun des processus, puis je les ai sommées pour obtenir le mélange. Comme on le voit, les sources estimées en utilisant 5.1.4 sont fidèles aux originales.

5.2 Mélange linéaire instantané multicanal (I quelconque)

Dans la section précédente, j'ai considéré le cas où l'unique mélange observé est la simple somme des sources. Dans de nombreuses applications pratiques, on dispose non pas d'un seul mélange des sources, mais de plusieurs. Par exemple, dans le cas de la séparation de sources audio, si notre objectif est de séparer les différentes pistes audio présentes dans un morceau de musique, il est bien plus fréquent de disposer d'un signal stéréophonique plutôt que d'un signal monophonique. Dans ce cas, on parlera d'un mélange multicanal, par opposition au cas du mélange *monocanal*. Tout au long de ce texte, I désigne le nombre de mélanges disponibles.

Dans un mélange linéaire instantané, Le mélange pour chaque $t \in \mathbb{T}$ est une combinaison linéaire des sources à la même position. $A_{ij} \in \mathbb{C}$ peut se comprendre comme le *gain* de la source j dans le mélange i.

Comme signalé en section 1.2.2, il est courant d'introduire la notion d'*image* d'une source $\tilde{s}(\cdot, j)$. Alors qu'une source est par définition un signal monocanal, l'image $\tilde{y}(\cdot, \cdot, j)$ est un signal multicanal, défini sur $\mathbb{T} \times \mathbb{N}_I$. Dans tout ce texte, on aura par définition :

$$\tilde{x} = \sum_{j=1}^{J} \tilde{y}(\cdot, \cdot, j). \tag{5.2.1}$$

Un processus de mélange, ou de *mixage*, comme cela a déjà été souligné en section 1.2.2 page 2, se définit alors comme l'opération à partir de laquelle les différentes images $\tilde{y}(\cdot, \cdot, j)$ sont obtenues à partir des sources $\tilde{s}(\cdot, j)$ correspondantes. Dans ce chapitre, je considère le cas très simple d'un mélange linéaire instantané, pour lequel il existe une matrice A de dimension $I \times J$, telle que :

$$\tilde{y}(\cdot, i, j) = A_{ij} \tilde{s}(\cdot, j). \tag{5.2.2}$$

De cette manière, en combinant 5.2.2 et 5.2.1, on obtient

$$\forall t \in \mathbb{T},\ \tilde{x}(t, \cdot) = \tilde{s}(t, \cdot)\, A^{\top}. \tag{5.2.3}$$

On constate que le cas 5.1.1 est un cas particulier de 5.2.3 pour $I = 1$ et $A_{ij} = 1$.

Les opérations de la section précédente peuvent être menées exactement à l'identique pour la séparation d'un mélange multicanal. Pour alléger les notations, je supposerai que les sources $\tilde{s}(\cdot, j)$ sont toutes de moyenne nulle. Il est cependant entendu qu'une moyenne non nulle mais connue est prise en compte aisément, comme dans la section précédente.

Ainsi, je suppose toujours un domaine de définition $\mathbb{T}$ quelconque et J processus gaussiens indépendant $\tilde{s}(\cdot, j)$ définis sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$. Leurs fonctions de covariance k_j sont toujours supposées connues [1]. Les mélanges 5.2.3, en tant que combinaisons linéaires de processus gaussiens sont eux-mêmes des processus gaussiens, et on peut une fois de plus écrire la distribution jointe des observations et des sources :

$$\begin{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{s}}(T',1) \\ \vdots \\ \tilde{\boldsymbol{s}}(T',J) \end{bmatrix} \\ \begin{bmatrix} \tilde{\boldsymbol{x}}(T,1) \\ \vdots \\ \tilde{\boldsymbol{x}}(T,I) \end{bmatrix} \end{bmatrix} \sim \mathcal{N}\left(0, K_{\text{jointe}}\right), \tag{5.2.4}$$

où la matrice K_{jointe}, de dimension $(JL' + IL) \times (JL' + IL)$ est structurée de la manière suivante :

$$K_{\text{jointe}} = \begin{bmatrix} K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right) & K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right) \\ K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right) & K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right) \end{bmatrix},$$

où les matrices $K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right)$, $K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right)$, $K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right)$ et $K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right)$, de dimensions respectives $JL' \times JL'$, $IL \times JL'$, $JL' \times IL$ et $IL \times IL$, sont données par :

$$K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right) = \begin{bmatrix} K_1(T',T') & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & K_J(T',T') \end{bmatrix},$$

$$K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right) = \begin{bmatrix} K_1(T,T')\,A_{11} & \cdots & K_J(T,T')\,A_{1J} \\ \vdots & \ddots & \vdots \\ K_1(T,T')\,A_{I1} & \cdots & K_J(T,T')\,A_{IJ} \end{bmatrix},$$

$$K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right) = K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right)^H$$

$$K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right) = \begin{bmatrix} \sum_{j=1}^{J} A_{1j} K_j(T',T')\,A_{1j}^* & \cdots & \sum_{j=1}^{J} A_{1j} K_j(T',T')\,A_{Ij}^* \\ \vdots & \ddots & \vdots \\ \sum_{j=1}^{J} A_{Ij} K_j(T',T')\,A_{1j}^* & \cdots & \sum_{j=1}^{J} A_{Ij} K_j(T',T')\,A_{Ij}^* \end{bmatrix}. \tag{5.2.5}$$

Dans ces conditions, il est facile de déterminer la distribution *a posteriori* des J sources en T' étant donnés les I mélanges observés en T en procédant une fois de plus au conditionnement de la distribution jointe 5.2.4 par rapport aux I observations $\tilde{\boldsymbol{x}}(T,i)$. Pour ce faire, j'utilise encore les résultats de la section 2.2.1 :

$$\begin{bmatrix} \tilde{\boldsymbol{s}}(T',1) \\ \vdots \\ \tilde{\boldsymbol{s}}(T',J) \end{bmatrix} \mid \begin{bmatrix} \tilde{\boldsymbol{x}}(T,1) \\ \vdots \\ \tilde{\boldsymbol{x}}(T,I) \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\text{post}}, K_{\text{post}}\right), \tag{5.2.6}$$

avec

$$\boldsymbol{\mu}_{\text{post}} = K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right) K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right)^{-1} \begin{bmatrix} \tilde{\boldsymbol{x}}(T,1) \\ \vdots \\ \tilde{\boldsymbol{x}}(T,I) \end{bmatrix} \tag{5.2.7}$$

et

$$\begin{aligned} K_{\text{post}} = {} & K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right) \\ & - K\left(\tilde{\boldsymbol{s}}(T',\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right) K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{x}}(T,\cdot)\right)^{-1} K\left(\tilde{\boldsymbol{x}}(T,\cdot), \tilde{\boldsymbol{s}}(T',\cdot)\right). \end{aligned} \tag{5.2.8}$$

1. Ainsi que leurs fonctions moyennes, ici identiquement nulles.

Dans le formalisme gaussien proposé, il existe une dissociation complète du problème de la séparation et de celui de la modélisation. Si on dispose des hyperparamètres des sources ainsi que des coefficients de la matrice de mélange, on peut procéder efficacement à la séparation.

Une fois de plus, puisque leur distribution *a posteriori* 5.2.6 est gaussienne, l'estimée des sources qui minimise l'erreur quadratique est simplement leur moyenne *a posteriori* 5.2.7.

Évidemment, le calcul de 5.2.7 et 5.2.8 fait intervenir l'inversion de $K\left(\tilde{\boldsymbol{x}}\left(T,\cdot\right),\tilde{\boldsymbol{x}}\left(T,\cdot\right)\right)$, qui est une matrice de dimension $IL\times IL$. Dans de nombreux cas de figures, la complexité de ce calcul est prohibitive.

Cependant, il est remarquable que dans tout ce qui précède, aucune hypothèse particulière n'ait été faite sur la nature du domaine de définition $\mathbb{T}$ des sources, ni sur l'expression précise de leurs fonctions de covariance. Bien que ces résultats soient classiques dans le cas de la séparation de séries temporelles [38], on voit qu'il est possible d'utiliser ce formalisme gaussien dans des contextes jusqu'à présent inédits de séparation. On peut de plus remarquer que les expressions ci-dessus sont valables quel que soit le nombre I de mélanges par rapport au nombre J de sources. En d'autres termes, le formalisme gaussien proposé n'est pas restreint au cas de la séparation sous-déterminée ou bien à celui de la séparation sur-déterminée.

5.3 Processus gaussiens localement stationnaires

Bien qu'elles permettent la séparation de processus gaussiens indépendants définis sur des espaces quelconques et caractérisés par leurs seules fonctions de covariance et de moyenne, les opérations proposées en section précédente souffrent d'une complexité calculatoire qui peut devenir prohibitive si le nombre LI des observations est trop grand. Dans le cas des signaux audio, il est fréquent d'avoir $L\approx 10^6$. Par conséquent, ces techniques ne peuvent pas être utilisées telles quelles [2].

Au chapitre 3, j'ai présenté plusieurs approximations permettant de rendre les calculs possibles dans le cas d'un nombre très important de données. D'une manière générale, toutes les techniques permettant de réduire la complexité de l'inférence dans des modèles de processus gaussiens peuvent être mises à profit dans le formalisme proposé pour la séparation de sources. Ainsi, il est fréquent de pouvoir considérer un cas particulier dans lequel le calcul de la distribution *a posteriori* des sources 5.2.6 se simplifie.

Je vais à présent montrer que le modèle localement stationnaire, présenté en section 3.3 page 52, est un tel exemple de cas particulier pour lequel la complexité des calculs passe de $\mathcal{O}\left(L^3I^3\right)$ à $\mathcal{O}\left(IL\log\left(L\right)\right)$ et est dominée par des opérations de transformées de Fourier, pour lesquelles il existe plusieurs implémentations optimisées disponibles [3].

Je me place dans cette section dans le même cadre de travail qu'en section 3.3 page 52. Ainsi, le domaine de définition des sources est $\mathbb{T}=\mathbb{Z}^D$ et les observations sont faites sur une grille T de points régulièrement échantillonnés 2.5.1 page 35. Comme c'est souvent le cas, je considèrerai qu'on souhaite estimer les sources aux mêmes points $T'=T$ que les observations. Je supposerai de plus que l'opérateur de tramage $\mathcal{G}$ considéré est connu, tel que défini en section 3.2 page 47.

Dans ces conditions, soit $\tilde{s}$ un groupe de J PGLS indépendants et centrés $\tilde{s}\left(\cdot,j\right)$. Le signal observé n'est pas $\tilde{s}$, mais plutôt un groupe $\tilde{x}$ de I formes d'onde, défini de la même manière qu'en 5.2.3. Du fait de la linéarité à la fois du tramage et de la transformée de Fourier, l'équation de mélange 5.2.3 se traduit à l'identique sur les TFCT $\boldsymbol{x}$ du signal observé $\tilde{\boldsymbol{x}}$, comme cela a déjà été évoqué en 1.3.1 :

$$\forall\left(\boldsymbol{f},\boldsymbol{n}\right),\boldsymbol{x}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right)=A\boldsymbol{s}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right), \tag{5.3.1}$$

où les notations ont déjà été définies en partie 1.3.2 page 9.

Compte tenu de l'indépendance des sources et du fait que pour un PGLS, tous les points $(\boldsymbol{f},\boldsymbol{n})$ sont indépendants (voir section 3.3.2 page 53), l'estimation des vecteurs sources $\boldsymbol{s}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right)$, de dimension $J\times 1$, peut se faire indépendamment pour chaque point $(\boldsymbol{f},\boldsymbol{n})$. De plus, tous les $\left\{\boldsymbol{x}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right)\right\}_{\boldsymbol{f},\boldsymbol{n}}$ sont des vecteurs de dimension $I\times 1$ indépendants entre eux et gaussiens.

2. On verra cependant au chapitre 8 que ce formalisme général peut s'avérer utile dans certaines applications.

3. La plupart des langages de calcul scientifique tels que Python ou Matlab disposent d'implémentations efficaces des transformées de Fourier.

Une fois encore, la méthodologie adoptée pour estimer la distribution *a posteriori* des sources consiste à tout d'abord exprimer la distribution jointe des sources et des mélanges :

$$\begin{bmatrix} \boldsymbol{s}(\boldsymbol{f},\boldsymbol{n},\cdot) \\ \boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \end{bmatrix} \sim \mathcal{N}_c\left(0, \begin{bmatrix} \operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot)) & \operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H \\ A\operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot)) & A\operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H \end{bmatrix}\right), \tag{5.3.2}$$

où $P(\boldsymbol{f},\boldsymbol{n},\cdot)$ est le vecteur de dimension $J \times 1$ qui donne la densité spectrale de puissance des J sources au point $(\boldsymbol{f},\boldsymbol{n})$. En vertu du théorème de Wiener-Khinchin, leur connaissance est équivalente à celle des fonctions de covariance des sources (voir section 2.5.3). Il est alors possible de simplement déduire de 5.3.2 et des résultats de la section 2.2.1 l'expression analytique de la distribution *a posteriori* des sources étant donné le mélange :

$$\boldsymbol{s}(\boldsymbol{f},\boldsymbol{n},\cdot) \mid \boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \sim \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot), K_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot,\cdot)\right), \tag{5.3.3}$$

avec

$$\boldsymbol{\mu}_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot) = \operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H \left(A\operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H\right)^{-1} \boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \tag{5.3.4}$$

et

$$\begin{aligned} K_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot,\cdot) = {} & \operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot)) \\ & - \operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H \left(A\operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H\right)^{-1} A\operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot)) \end{aligned} \tag{5.3.5}$$

Une fois encore, la distribution *a posteriori* des sources étant gaussienne, la valeur estimée qui minimise l'erreur quadratique est simplement donnée par leur moyenne 5.3.4. Contrairement aux résultats de la section précédente faisant intervenir l'inversion d'une grande matrice de dimension $LI \times LI$, le calcul de la moyenne 5.3.4 et de la covariance 5.3.5 *a posteriori* ne fait intervenir pour chaque point $(\boldsymbol{f},\boldsymbol{n})$ que l'inversion de la matrice $A\operatorname{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\, A^H$, de dimension $I \times I$, ce qui constitue une réduction considérable de la complexité de la méthode. De plus, les traitements dans chaque point $(\boldsymbol{f},\boldsymbol{n})$ pouvant être effectués indépendamment, il est possible de largement paralléliser ces calculs. Une fois les moyennes *a posteriori* calculées dans tous les points $(\boldsymbol{f},\boldsymbol{n})$, les sources estimées peuvent être récupérées simplement par transformée de Fourier inverse puis par addition-recouvrement, comme suggéré en section 3.2.3. La complexité totale de ces opérations est dominée par les calculs des transformées de Fourier, menant à un traitement d'une complexité totale [4]de $\mathcal{O}(INF\log(F))$, ce qui constitue une réduction particulièrement remarquable de la complexité initiale $\mathcal{O}\left(L^3I^3\right)$ du calcul dans le cas général. Bien entendu, cette réduction de la complexité de la séparation a un prix, puisqu'elle n'est valide que dans le cas où les sources peuvent être modélisées comme des processus gaussiens localement stationnaires centrés et sont de plus régulièrement échantillonnées. Tel est le cas des signaux audio, qui m'occuperont en parties III et IV.

L'état de l'art [17, 30, 45, 72, 69, 155] se concentre exclusivement sur le cas des PGLS, pour $D = 1$. Comme on le voit, les opérations de séparation correspondantes sont directement applicables à D quelconque, et ce quel que soit le modèle choisi pour les DSP P des sources. C'est en cherchant le modèle temporel correspondant à ces méthodes que je suis parvenu à la formulation plus générale qui fait l'objet de cette partie.

De la même manière qu'en section précédente, les calculs ci-dessus sont valables quel que soit le nombre I de mélanges par rapport au nombre J de sources et mènent donc à des solutions opérationnelles à la fois dans le cas sous-déterminé ($I < J$) et sur-déterminé ($I \geq J$). On peut d'ailleurs remarquer que si A est inversible, 5.3.4 coïncide simplement avec $A^{-1}\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot)$. Dans la littérature sur la séparation de sources, les développements ci-dessus sont désormais classiques dans le cas des séries temporelles ($D = 1$) et pour un ou deux capteurs. On ne peut cependant pas dire qu'il soit largement reconnu, même aujourd'hui, que le même modèle gaussien peut s'appliquer

4. Pour chacune des N trames de chacun des I mélanges, le calcul d'une transformée de Fourier rapide peut se faire en $\mathcal{O}(F\log F)$ opérations si F est une puissance de 2.

quelle que soit la détermination du problème et indépendamment du modèle choisi pour les DSP P des sources. De plus, ce texte est à ma connaissance le premier à établir que les mêmes résultats sont valides quelle que soit la dimension D du domaine de définition des sources. Ce faisant, il établit une connexion intéressante entre les techniques appliquées en traitement des séries temporelles pour la séparation et celles utilisées en interpolation spatiale [44, 32]. Dans ce domaine, il est cependant notable que les signaux ne peuvent que très rarement être considérés comme centrés, leurs fonctions moyenne sont rarement connues et il arrive fréquemment qu'ils ne soient pas échantillonnés de manière régulière. Malgré tout, ce sont les mêmes idées de mélanges de processus gaussiens qui y sont à l'œuvre. D'ailleurs, le modèle de mélange linéaire instantané 5.2.3 a des similarités frappantes avec celui de la *co-régionalisation*, fréquemment utilisé en géostatistiques.

Chapitre 6

Mélanges complexes

Dans ce chapitre, je montre comment le modèle linéaire instantané présenté au chapitre précédent peut être étendu au cas de modèles de mélanges plus complexes. Les deux modèles que je présente ici se comprennent comme autant de manières de définir le lien entre les J signaux sources $\tilde{s}(\cdot, j)$ et leurs images $\tilde{y}(\cdot, \cdot, j)$, dont les mélanges observés sont la somme 5.2.1 :

$$\forall i,\ \tilde{x}(\cdot, i) = \sum_{j=1}^{J} \tilde{y}(\cdot, i, j)\,.$$

Un modèle de mixage permet de préciser le lien entre un processus source et son image. Comme son nom l'indique, il n'est qu'un *modèle* qui rend compte d'une réalité complexe. Lorsque je dirai que les sources sont mixées de manière convolutive ou diffuse, je voudrai bien sûr dire que leur mélange est modélisé en utilisant une approximation convolutive ou diffuse.

Si le modèle linéaire instantané 5.2.2 introduit une relation linéaire très simple entre chaque source et son image, il est possible d'envisager d'autres modèles de mélange. Dans un premier temps, je présente en section 6.1 le modèle de mélange convolutif, classique en traitement du signal audio et fréquemment utilisé en séparation de sources dans ce domaine [155, 38, 50]. L'intérêt de ma présentation par rapport à celles qu'on trouvera dans la littérature réside principalement dans le fait que je présente ce modèle dans un contexte un peu plus général, où le domaine de définition $\mathbb{T}$ des sources est $\mathbb{Z}^D$, et non pas seulement $\mathbb{Z}$. Cependant, il est clair que cette extension ne se fait qu'au prix de très minimes complications des expressions et cela explique la concision de ma présentation de ce modèle classique.

Dans un deuxième temps, je présenterai en section 6.2 le modèle *diffus,* aussi appelé modèle *de rang plein*, récemment proposé par Duong *et al.* [50, 49, 47] dans un contexte de séparation de sources audio. Ce modèle établit une rupture forte avec le modèle convolutif dans la mesure où le lien entre une source et son image n'est plus supposé déterministe, mais probabiliste. En ce sens, il permet une bien plus grande souplesse, au prix d'un plus grand nombre de paramètres. Une fois encore, je présenterai ce modèle dans le cadre plus général des PGLS en dimension quelconque, et je préciserai comment les images des sources peuvent y être récupérées à partir des mélanges.

6.1 Modèle convolutif

6.1.1 Motivations : le cas de l'audio ($\mathbb{T} = \mathbb{Z}$)

Lors de la production d'un morceau de musique à partir des différentes pistes qui le composent, il est extrêmement fréquent d'appliquer aux pistes certains *effets* lors de la procédure de mixage, en plus de leur latéralisation sur le plan stéréo ou dans l'espace spatial du format 5.1. Certains de ces effets, comme l'égalisation, peuvent être modélisés comme un processus de filtrage. Dans ces conditions, le $i^{\text{ème}}$ canal $\tilde{y}(\cdot, i, j)$ de l'image de la source $\tilde{s}(\cdot, j)$, aussi appelé la contribution de la

source j dans le $i^{\text{ème}}$ mélange, est obtenu par :

$$\forall t \in \mathbb{Z},\ \tilde{y}\left(t,i,j\right) = \sum_{\tau=-\infty}^{+\infty} a_{ij}\left(\tau\right) \tilde{s}\left(t-\tau,j\right), \tag{6.1.1}$$

où a_{ij} est la réponse impulsionnelle du filtre de mélange de la source j vers le mélange i. L'opération 6.1.1 est une *convolution*, notée $*$ dans la suite :

$$\forall t \in \mathbb{Z},\ \tilde{y}\left(t,i,j\right) = \left(a_{ij}\left(\cdot\right) * \tilde{s}\left(\cdot,j\right)\right)\left(t\right).$$

La classification des différentes propriétés de l'opération de filtrage est un problème classique de la littérature du traitement du signal. Dans la suite de cet exposé, je considèrerai toujours que les filtres de mélange a_{ij} sont tels que 6.1.1 existe. À cet effet, je me restreindrai au cas simple et suffisant (mais pas nécessaire) des filtres de mélange à *réponse impulsionnelle finie* (RIF), c'est-à-dire pour lesquels

$$\exists H \in \mathbb{N} : \forall \tau \in \mathbb{T},\ |\tau| > H \Rightarrow a_{ij}\left(\tau\right) = 0 \tag{6.1.2}$$

où le plus petit H vérifiant 6.1.2 désigne l'ordre du filtre. Si on suppose en plus que le filtre est *causal*, on a une réponse impulsionnelle nulle pour $\tau < 0$.

Comme on le voit, le cas linéaire instantané 5.2.2 est un cas particulier de mélange convolutif pour lequel $H = 0$.

Le modèle convolutif est assez adéquat pour modéliser un effet d'égalisation appliqué à des sons, et très satisfaisant lorsqu'on cherche à simuler le parcours d'un son dans l'air ou sa propagation à travers des matériaux. Il est également très utile pour rendre compte fidèlement du trajet acoustique d'une source ponctuelle située à une position spatiale quelconque vers chacune des oreilles d'un auditeur. Dans ce contexte, on parle volontiers de filtrage binaural, ou encore de spatialisation. Pour peu que l'ordre H considéré soit suffisamment élevé, il peut aussi être adéquat pour la modélisation de réverbération ou d'échos.

En revanche, il arrive qu'un modèle convolutif ne soit pas satisfaisant. On peut constater en observant son expression mathématique 6.1.1 qu'il s'agit d'une opération linéaire. Or, il arrive fréquemment dans la production musicale que des traitements fortement non-linéaires soient appliqués aux sources. Par exemple, il est extrêmement fréquent d'appliquer sur la voix une *compression*, qui en réduit la dynamique au sein du mélange. De la même manière, il est courant d'appliquer une distorsion à un son de voix ou de guitare pour en modifier le timbre. De tels traitements sont mal modélisés par une simple convolution. Il faut avoir conscience de ce problème lorsqu'on adopte un modèle convolutif pour rendre compte du processus de mixage [1].

D'expérience, je dirais qu'un modèle convolutif est satisfaisant dès lors qu'on cherche à rendre compte d'une opération *naturelle* appliquée à une source sonore ponctuelle comme sa diffusion dans l'air ou sa transmission à travers certains matériaux. Il atteint souvent ses limites dès lors que la source ne peut plus être supposée ponctuelle ou que les traitements considérés sont résolument *artificiels*, comme le cas d'une lourde réverbération, d'une distorsion, d'une compression dynamique, etc.

Cependant, le modèle convolutif reste central dans de nombreuses méthodes de séparation de sources, parce qu'il permet des traitements particulièrement efficaces en termes de complexité calculatoire. On verra d'ailleurs au chapitre 12 que ses performances pour la récupération des images sont souvent très satisfaisantes, même si le mélange est fortement non linéaire. Dans tous les cas, il existe une communauté de chercheurs qui s'attelle aujourd'hui à la mise au point de modèles efficaces permettant de séparer des mélanges non linéaires [202, 113, 2, 38].

1. La modélisation de mixages non linéaires complexes en audio reste encore aujourd'hui un enjeu scientifique ouvert. Le fait que l'article récent de N. Stürmel sur le sujet [201] ait reçu un prix distinctif montre qu'il existe une réelle demande pour des modèles plus fidèles du mixage musical tel qu'accompli par les professionnels.

6.1.2 Séparation de mélanges convolutifs ($\mathbb{T} = \mathbb{Z}^D$)

Dans cette section, je vais m'intéresser au cas général [2] de la séparation de sources définies sur $\mathbb{T} = \mathbb{Z}^D$, où le mixage est convolutif. Comme précédemment, je noterai $\tilde{s}$ un ensemble de J processus gaussiens indépendants définis sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$, dont les fonctions de moyenne et de covariance sont notées $\mu_j(t)$ et $k_j(t,t')$. Une fois encore, les I mélanges $\tilde{x}(\cdot,i)$ sont modélisés comme la somme instantanée des J images $\tilde{y}(\cdot,\cdot,j)$ comme en 5.2.1 page 77. Je ferai à nouveau l'hypothèse que toutes les fonctions moyenne μ_j sont nulles pour alléger les notations. Si tel n'est pas le cas, on peut les prendre en compte facilement comme je l'ai fait en section 5.1.

L'opération de convolution 6.1.1 par laquelle les images sont obtenues à partir des sources dans le cas $\mathbb{T} = \mathbb{Z}$ des séries temporelles peut sans problème être généralisée au cas $\mathbb{T} = \mathbb{Z}^D$ et on a alors :

$$\forall (t,i,j), \tilde{y}(t,i,j) = \sum_{\tau \in \mathbb{T}} a_{ij}(\tau)\, \tilde{s}(t-\tau, j),$$

noté comme en section précédente :

$$\tilde{y}(t,i,j) = (a_{ij}(\cdot) * \tilde{s}(\cdot,j))(t), \tag{6.1.3}$$

où $a_{ij}(\cdot)$ est encore ici appelé la réponse impulsionnelle du filtre de mélange de la source j vers le mélange i. Une fois encore, je supposerai que 6.1.3 existe. A cet effet, je ferai de nouveau l'hypothèse suffisante mais pas nécessaire que a_{ij} est à réponse impulsionnelle finie, en supposant [3] :

$$\exists H \in \mathbb{N} : \forall \tau \in \mathbb{T}, \|\tau\|_\infty > H \Rightarrow a_{ij}(\tau) = 0. \tag{6.1.4}$$

Le plus petit H vérifiant 6.1.4 sera appelé l'ordre du filtre de mélange. Il définit une boule en dehors de laquelle la réponse impulsionnelle a_{ij} est nulle.

En tant que combinaison linéaire de processus gaussiens, chaque image $\tilde{y}(\cdot,\cdot,j)$ est elle-même un processus gaussien défini sur $\mathbb{N}_I \times \mathbb{T}$ à valeur dans $\mathbb{C}$ dont la moyenne est $(a_{ij} * \mu_j)(t) = 0$ et dont la fonction de covariance $k_j((i,t),(i',t'))$ est donnée par :

$$\begin{aligned}
\forall ((i,t),(i',t')), \mathbb{E}\left[\tilde{y}(t,i,j)\, \tilde{y}(t',i',j)^*\right] &= \mathbb{E}\left[\left(\sum_{\tau\in\mathbb{T}} a_{ij}(\tau)\, \tilde{s}(t-\tau,j)\right)\left(\sum_{\tau'\in\mathbb{T}} a_{i'j}(\tau')\, \tilde{s}(t'-\tau',j)\right)^*\right] \\
&= \sum_{\tau,\tau'\in\mathbb{T}} a_{ij}(\tau)\, a_{i'j}(\tau')^*\, k_j(t-\tau, t'-\tau') \\
k_j((i,t),(i',t')) &= \left(\left(a_{ij} a^*_{i'j}(\cdot,\cdot)\right) * k_j(\cdot,\cdot)\right)(t,t'),
\end{aligned} \tag{6.1.5}$$

où $a_{ij}a^*_{i'j}(t,t') = a_{ij}(t)\, a^*_{i'j}(t')$ est le produit de a_{ij} et de $a^*_{i'j}$, défini sur $\mathbb{T}\times\mathbb{T}$. Comme on le voit, la fonction de covariance de chaque image peut s'exprimer en fonction de celle de la source correspondante et des réponses impulsionnelles de ses filtres de mélange.

Ainsi, j'ai établi que si les sources sont J processus gaussiens, l'image $\tilde{y}(\cdot,\cdot,j)$ de chacune est elle-même un processus gaussien dont j'ai donné les fonctions de moyenne et de covariance. Si l'objectif du traitement est de récupérer ces images à partir de l'observation du mélange, le même traitement que celui présenté en section 5.1 peut être utilisé. En effet, le mélange est par définition 5.2.1 la somme instantanée des J images. Il suffit donc d'utiliser leurs fonctions de moyenne et de covariance pour les séparer.

Par contre, si l'objectif est de récupérer non pas les J images $\tilde{y}(\cdot,\cdot,j)$, mais les sources $\tilde{s}(\cdot,j)$, alors il est nécessaire encore une fois d'appliquer la méthodologie déjà rencontrée à chaque fois que j'ai abordé un problème de régression en utilisant des processus gaussiens. Dans la mesure où le cas que j'aborde à présent généralise l'ensemble de ceux vus précédemment en sections 2.2.3, 5.1 et 5.2, il me parait intéressant de le détailler encore.

2. Il est possible d'étendre cette discussion à des domaines de définition $\mathbb{T}$ bien plus généraux que $\mathbb{Z}^D$ mais une telle extension est au-delà de mes compétences actuelles en mathématiques fondamentales. Dans les applications que je considère, le cas de $\mathbb{T} = \mathbb{Z}^D$ est de toute façon suffisant.

3. Pour $\tau = [\tau_1, \ldots, \tau_D]$, vecteur de dimension $D \times 1$, $\|\tau\|_\infty$ désigne la valeur la plus grande des τ_d.

Soient deux ensembles T et T' de L et L' points de $\mathbb{T}$, respectivement. Je suppose qu'une réalisation $\tilde{\boldsymbol{x}}(T,\cdot)$ du mélange multicanal, de dimension $IL\times 1$ est observée sur T et que l'objectif est d'estimer la valeur $\tilde{\boldsymbol{s}}(T',\cdot)$, de dimension $JL'\times 1$ des sources en T'. Avant d'aller plus loin, calculons la covariance $k_j((i,t),t')$ entre une image et la source correspondante :

La séparation des images dans le cas convolutif se ramène à celui de la séparation des sources dans un mélange monocanal instantané en remplaçant les fonctions moyenne et covariance des sources par celles des images 6.1.5.

$$\begin{aligned} k_j((i,t),t') &= \mathbb{E}\left[\tilde{y}(i,t,j)\,\tilde{s}(t',j)^*\right] \\ &= \mathbb{E}\left[\left(\sum_{\tau\in\mathbb{T}} a_{ij}(\tau)\,\tilde{s}(t-\tau,j)\right)\tilde{s}(t',j)^*\right] \\ &= \sum_{\tau\in\mathbb{T}} a_{ij}(\tau)\,k_j(t-\tau,t') \\ &= (a_{ij}(\cdot) * k_j(\cdot,t'))(t). \end{aligned} \tag{6.1.6}$$

Munis des fonctions de covariance k_j des sources et de l'expression 6.1.6, la distribution jointe de vec $\tilde{\boldsymbol{x}}(T,\cdot)$ et de vec $\tilde{\boldsymbol{s}}(T',\cdot)$ est donnée par :

$$\begin{bmatrix} \tilde{\boldsymbol{s}}(T',\cdot) \\ \tilde{\boldsymbol{x}}(T,\cdot) \end{bmatrix} \sim \mathcal{N}\left(0, K_{\text{jointe}}\right), \tag{6.1.7}$$

où la matrice K_{jointe}, de dimension $(JL'+IL)\times(JL'+IL)$ est structurée de la manière suivante :

$$K_{\text{jointe}} = \begin{bmatrix} K(\tilde{\boldsymbol{s}}(T',\cdot),\tilde{\boldsymbol{s}}(T',\cdot)) & K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{s}}(T',\cdot))^H \\ K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{s}}(T',\cdot)) & K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{x}}(T,\cdot)) \end{bmatrix},$$

dont les sous-matrices $K(\tilde{\boldsymbol{s}}(T',\cdot),\tilde{\boldsymbol{s}}(T',\cdot))$, $K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{s}}(T',\cdot))$ et $K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{x}}(T,\cdot))$, de dimensions respectives $JL'\times JL'$, $IL\times JL'$ et $IL\times IL$, sont définies naturellement en fonction de k_j et de 6.1.6. En effet, du fait de l'indépendance des sources, on a par exemple :

$$\mathbb{E}\left[\tilde{\boldsymbol{x}}(t_l,i)\,\tilde{\boldsymbol{s}}(t_{l'},j)^*\right] = k_j((i,t_l),t_{l'}).$$

Comme en section 5.2, on peut alors déterminer la distribution *a posteriori* de $\tilde{\boldsymbol{s}}(T',\cdot)$ étant donné $\tilde{\boldsymbol{x}}(T,\cdot)$ et obtenir les mêmes expressions :

$$\tilde{\boldsymbol{s}}(T',\cdot) \mid \tilde{\boldsymbol{x}}(T,\cdot) \sim \mathcal{N}\left(\boldsymbol{\mu}_{\text{post}}, K_{\text{post}}\right), \tag{6.1.8}$$

avec

$$\boldsymbol{\mu}_{\text{post}} = K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{s}}(T',\cdot))^H\, K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{x}}(T,\cdot))^{-1}\,\text{vec}\,\tilde{\boldsymbol{x}}(T,\cdot) \tag{6.1.9}$$

et

$$\begin{aligned} K_{\text{post}} = {} & K(\tilde{\boldsymbol{s}}(T',\cdot),\tilde{\boldsymbol{s}}(T',\cdot)) \\ & - K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{s}}(T',\cdot))^H\, K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{x}}(T,\cdot))^{-1}\, K(\tilde{\boldsymbol{x}}(T,\cdot),\tilde{\boldsymbol{s}}(T',\cdot)). \end{aligned} \tag{6.1.10}$$

Étant donnés $\{k_j\}_j$, $\{a_{ij}(\cdot)\}_{i,j}$ et $\tilde{\boldsymbol{x}}(T,\cdot)$, l'estimée des sources qui minimise l'erreur quadratique est donc donnée par $\boldsymbol{\mu}_{\text{post}}$ en 6.1.9. Ce calcul est d'une complexité $\mathcal{O}\left(I^3L^3\right)$, qui peut vite devenir prohibitive si les signaux observés sont de grande dimension.

Une fois encore, on voit que le formalisme gaussien permet de récupérer non seulement leurs images, mais aussi les sources, à partir de l'observation d'un mélange convolutif, et ce quel que soit le nombre relatif de sources J par rapport à celui des mélanges.

Il faut malgré tout souligner encore ici que tous les traitements présentés dans cette section se distinguent nettement du cas de la séparation *aveugle* des sources, où très peu d'hypothèses sont faites sur la nature des signaux. Ici au contraire, j'ai supposé qu'elles étaient des processus gaussiens dont les fonctions de covariance sont connues. Une telle configuration peut paraître irréaliste mais on verra qu'il n'en est rien dans le cas de la séparation informée en parties III et IV. Si elles ne sont pas connues, il est encore possible de mettre à profit le formalisme gaussien et on verra bientôt au chapitre 7 que ces paramètres peuvent être appris, si certaines connaissances *a priori* les concernant sont disponibles.

6.1.3 Processus gaussiens localement stationnaires

De la même manière que l'hypothèse de stationnarité locale a permis de grandement simplifier les calculs dans le cas instantané en section 5.3 page 79 si les signaux sont régulièrement échantillonnés, on va voir ici que les mêmes simplifications se produisent dans le cas d'un modèle convolutif. Je suppose donc la même configuration que dans le cas instantané vu en section 5.3 page 79, mais les images des sources sont remplacées par leur version convolutive 6.1.3.

Je vais montrer que sous certaines conditions, le mélange convolutif de processus gaussiens localement stationnaires se traduit dans le domaine fréquentiel par un mélange instantané similaire à 5.3.1 page 79 pour lequel la matrice de mélange A dépend désormais de l'indice de fréquence $\boldsymbol{f}$ considéré. Ces développements, désormais classiques, ont été proposés dans la littérature sur la séparation de mélanges convolutifs dans [164] et repris récemment [155, 163].

Pour commencer, c'est une des propriétés de la transformée de Fourier discrète de transformer une convolution circulaire en multiplication.

Si $\boldsymbol{a}$ et $\boldsymbol{b}$ sont deux vecteurs de dimension $H \times 1$, soit $\boldsymbol{a} \circledast \boldsymbol{b}$ leur convolution circulaire, définie par :

$$(\boldsymbol{a} \circledast \boldsymbol{b})(n) = \sum_{\tau \in \mathbb{N}_H} \boldsymbol{a}(\tau)\, \boldsymbol{b}((n-\tau) \bmod [H]), \tag{6.1.11}$$

où $(n-\tau) \bmod [H]$ est le reste de la division Euclidienne de $n-\tau$ par H (opération *modulo*). Dans ces conditions, le théorème de la convolution circulaire indique que :

$$\mathcal{F}\{\boldsymbol{a} \circledast \boldsymbol{b}\} = \mathcal{F}\{\boldsymbol{a}\} \cdot \mathcal{F}\{\boldsymbol{b}\}.$$

Une propriété largement utilisée en pratique est que si il existe $H_a \ll H$ tel que

$$\forall n > H_a, \boldsymbol{a}(n) = 0,$$

c'est-à-dire si un des vecteurs est bien plus court que l'autre (et ramené à la taille H par ajout de zéros terminaux), alors on a :

$$\mathcal{F}\{\boldsymbol{a} * \boldsymbol{b}\} \approx \mathcal{F}\{\boldsymbol{a}\} \cdot \mathcal{F}\{\boldsymbol{b}\}. \tag{6.1.12}$$

En d'autres termes, la convolution des vecteurs $\boldsymbol{a}$ et $\boldsymbol{b}$ est à peu près équivalente à une multiplication dans le domaine fréquentiel, parce que dans ce cas, elle est très proche d'une convolution circulaire. Les mêmes résultats se montrent en dimension quelconque, où la convolution circulaire 6.1.11 est définie de manière analogue.

Par ailleurs, je vais montrer à présent que sous certaines conditions, on peut exprimer l'expression d'une trame $\mathcal{G}\{\tilde{x}(\cdot, i)\}(t, \boldsymbol{n})$ du $i^{\text{ème}}$ mélange en fonction de celle du tramage $\mathcal{G}\{\tilde{s}(\cdot, j)\}$ des sources. Pour ce faire, je reprends les mêmes notations que celles de la section 3.2.3 page 48 :

$$\begin{aligned}
\forall (i, t, \boldsymbol{n}) \in \mathbb{N}_I \times \mathbb{T}_0 \times \mathcal{N}_{\mathcal{G}},\ \mathcal{G}\{\tilde{x}(\cdot, i)\}(t, \boldsymbol{n}) &= \mathcal{G}\left\{\sum_{j=1}^{J} a_{ij}(\cdot) * \tilde{s}(\cdot, j)\right\}(t, \boldsymbol{n}) \\
&= \sum_{j=1}^{J} \mathcal{G}\{a_{ij}(\cdot) * \tilde{s}(\cdot, j)\}(t, \boldsymbol{n}) \\
&= \sum_{j=1}^{J} g(t)\,(a_{ij}(\cdot) * \tilde{s}(\cdot, j))(t + \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho})) \\
&= \sum_{j=1}^{J} g(t) \sum_{\tau \in \mathbb{T}} a_{ij}(\tau)\, \tilde{s}(t + \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho}) - \tau, j) \\
&= \sum_{j=1}^{J} \sum_{\tau \in \mathbb{T}} a_{ij}(\tau)\, (g(t)\, \tilde{s}(t - \tau + \boldsymbol{n} \cdot \boldsymbol{L}_0 \cdot (1 - \boldsymbol{\rho}), j))
\end{aligned}$$

En observant cette dernière expression et en la comparant à la définition du fenêtrage 3.2.6 page 49, on remarque que si on a :

$$\forall (t, \tau), g(t) = g(t - \boldsymbol{\tau}), \tag{6.1.13}$$

autrement dit, *si g peut être considérée comme constante* sur le support de la réponse impulsionnelle de a_{ij}, alors on a :

$$\forall (i, t, \boldsymbol{n}) \in \mathbb{N}_I \times \mathbb{T}_0 \times \mathcal{N}_{\mathcal{G}}, \mathcal{G}\{\tilde{x}(\cdot, i)\}(t, \boldsymbol{n}) \approx \sum_{j=1}^{J} (a_{ij} * \mathcal{G}\{\tilde{s}(\cdot, j)\}(\cdot, \boldsymbol{n}))(t), \tag{6.1.14}$$

où l'approximation survient du fait de l'inexactitude de l'expression 6.1.13. Pour peu que l'ordre H de tous les filtres de mélange soit suffisamment petit devant la taille du domaine de définition $\mathbb{T}_0$ de chaque trame, l'approximation 6.1.14 est valide, ce qui signifie que le tramage d'une convolution peut être approximé par la convolution du tramage.

Enfin, si H est suffisamment petit, on a vu que l'approximation 6.1.12 est également valide, ce qui signifie que la TFCT $\boldsymbol{x}$ des mélanges est alors donnée par :

$$\forall (\boldsymbol{f}, \boldsymbol{n}), \boldsymbol{x}(\boldsymbol{f}, \boldsymbol{n}, \cdot) \approx A(\boldsymbol{f}) \boldsymbol{s}(\boldsymbol{f}, \boldsymbol{n}, \cdot), \tag{6.1.15}$$

où

$$[A(\boldsymbol{f})]_{i,j} = \mathcal{F}_D\{a_{ij}\}(\boldsymbol{f}) \tag{6.1.16}$$

est une matrice de mélange dépendante de la fréquence.

On voit que sous certaines hypothèses, le cas du mélange convolutif peut se ramener à celui des mélanges instantanés, où la matrice de mélange est différente selon la fréquence. Ce constat, désormais classique en séparation de signaux audio [164, 155], est d'une importance pratique considérable. Il signifie en effet que :

> la séparation de sources mélangées de manière convolutive peut être accomplie exactement de la même manière qu'en section 5.3, en remplaçant la matrice de mélange A par $A(\boldsymbol{f})$ défini en 6.1.16. Le cas linéaire instantané ($H = 0$) est un cas particulier de 6.1.16, pour lequel la matrice de mélange est constante en fonction de la fréquence.

Dans le cas convolutif, si les filtres de mélanges sont suffisamment courts et si la fenêtre de pondération utilisée lors du fenêtrage est constante ou évolue lentement à l'échelle des filtres de mélange, alors le mélange reste instantané pour les TFCT.

Il est clair que l'approximation 6.1.15 repose sur deux hypothèses assez restrictives. La première est que les filtres de mélange ont une réponse impulsionnelle suffisamment courte [4]. La deuxième est l'utilisation d'une fenêtre d'analyse rectangulaire pour le tramage, ou tout au moins d'une fenêtre qui varie peu sur le support de la réponse impulsionnelle des filtres de mélange [5]. Une fenêtre rectangulaire ne permet pas de récupérer des signaux lisses par addition-recouvrement 3.2.9 page 49. En effet, elle introduit des discontinuités entre les trames au moment de la reconstruction et il devient nécessaire pour la synthèse des signaux d'utiliser une méthode faisant intervenir une fenêtre de synthèse lisse, différente de celle d'analyse. La procédure d'addition-recouvrement utilisée est alors 3.2.10.

De manière à prendre en compte le fait que l'équation de mélange 6.1.15 n'est qu'une approximation, Certains auteurs proposent [155, 50] de ne pas prétendre à son exactitude dans les calculs, mais de plutôt la remplacer par :

$$\forall (\boldsymbol{f}, \boldsymbol{n}), \boldsymbol{x}(\boldsymbol{f}, \boldsymbol{n}, \cdot) = A(\boldsymbol{f}) \boldsymbol{s}(\boldsymbol{f}, \boldsymbol{n}, \cdot) + \boldsymbol{\epsilon}(\boldsymbol{f}, \boldsymbol{n}, \cdot), \tag{6.1.17}$$

4. L'approximation est correcte pour des filtres d'un support environ 5 fois plus petit que celui des trames.

5. Je remercie Roland Badeau et Cédric Févotte de m'avoir aidé à l'identification précise des conditions nécessaires pour que 6.1.15 soit vérifié.

où $\boldsymbol{\epsilon}(\boldsymbol{f},\boldsymbol{n},\cdot)$ est un bruit additif gaussien, dont la matrice de covariance $K_\epsilon(\boldsymbol{f},\boldsymbol{n})$, de dimension $I\times I$, est supposée connue[6]. Dans ces conditions, la matrice de covariance du mélange devient

$$K(\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot)) = A(\boldsymbol{f})\,\mathrm{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\,A(\boldsymbol{f})^H + K_\epsilon(\boldsymbol{f},\boldsymbol{n}), \tag{6.1.18}$$

et la distribution *a posteriori* des sources $\boldsymbol{s}(\boldsymbol{f},\boldsymbol{n},\cdot)$ est donnée par :

$$\boldsymbol{s}(\boldsymbol{f},\boldsymbol{n},\cdot) \mid \boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \sim \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot), K_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot,\cdot)\right), \tag{6.1.19}$$

avec

$$\boldsymbol{\mu}_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot) = \mathrm{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\,A(\boldsymbol{f})^H\,K(\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))^{-1}\,\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \tag{6.1.20}$$

et

$$\begin{aligned} K_{\text{post}}(\boldsymbol{f},\boldsymbol{n},\cdot,\cdot) &= \mathrm{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot)) \\ &\quad - \mathrm{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot))\,A(\boldsymbol{f})^H\,K(\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))^{-1}\,A(\boldsymbol{f})\,\mathrm{diag}(P(\boldsymbol{f},\boldsymbol{n},\cdot)) \end{aligned} \tag{6.1.21}$$

Le terme de bruit additif dans 6.1.17 ne correspond pas nécessairement à un bruit réellement ajouté lors du mixage. Il peut ne correspondre qu'à la prise en compte d'une incertitude sur l'équation 6.1.15 ou à une erreur de modélisation.

Puisque leur distribution *a posteriori* 6.1.19 est gaussienne, les sources sont estimées aux moindres carrés par leur moyenne *a posteriori* 6.1.20. Une fois encore, ces conclusions sont valides quel que soit le nombre I de mélanges et le nombre J de sources et la dimension D du domaine de définition des sources. Il est intéressant de constater que l'introduction d'un bruit additif dans le modèle 6.1.17 permet souvent de garantir l'inversibilité de la matrice de covariance du mélange et ainsi de résoudre les problèmes numériques posés par un éventuel mauvais conditionnement, ce qui arrive fréquemment car de nombreuses valeurs de $A(\boldsymbol{f})$ sont très faibles.

6.2 Modèle diffus

6.2.1 Motivations

L'approximation par laquelle un mélange convolutif 6.1.1 se traduit par un mélange linéaire 6.1.17 des TFCT repose sur des hypothèses assez restrictives. Non seulement l'hypothèse même d'un mélange convolutif peut être discutable, comme on l'a vu en section 6.1.1, mais cette simplification n'est valable que pour des filtres de mélange dont la réponse impulsionnelle est d'un ordre petit devant la taille des trames. Or, il est fréquent dans de nombreuses applications pratiques que les filtres de mélange n'aient pas une réponse impulsionnelle courte. C'est par exemple le cas en traitement du signal audio lors de la présence d'une réverbération. De tels filtres ont fréquemment des réponses impulsionnelles d'une durée proche de la seconde, ce qui est bien plus long que la longueur d'une trame, habituellement de l'ordre de 50ms.

La prise en compte 6.1.18 d'une incertitude dans le modèle n'est souvent qu'un pis-aller dans la mesure où si la covariance $K_\epsilon(\boldsymbol{f},\boldsymbol{n})$ du bruit additif devient importante, elle éclipsera la contribution des sources dans le calcul de la covariance 6.1.18 des mélanges, baissant d'autant la qualité de l'estimation. Pour qu'un tel modèle de bruit soit efficace, l'expérience montre qu'il est important qu'il corresponde à une réelle source de bruit, ou bien qu'il soit de faible variance.

Dans ces conditions, N. Duong *et al.* ont introduit récemment pour la séparation de sources [50, 49, 48, 47] un modèle de mélange significativement plus expressif, issu de la communauté de l'acoustique probabiliste des salles [93] qui permet de prendre en compte des temps de réverbération

6. On verra au chapitre 7 qu'elle peut être apprise à partir des observations, comme les autres paramètres du modèle.

conséquents et des sources non ponctuelles. Ce modèle porte le nom de modèle gaussien de rang plein. Dans cet exposé, je l'appellerai aussi modèle *diffus.* Comme on va le voir, il admet le mixage convolutif 6.1.17 comme un cas particulier.

Toujours en adoptant les notations des PGLS de la section 5.3, le modèle diffus suppose que la covariance $K\left(\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right),\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right)\right)$ d'une image à un point $(\boldsymbol{f},\boldsymbol{n})$ est donnée par :

$$K\left(\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right),\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right)\right)=P\left(\boldsymbol{f},\boldsymbol{n},j\right)R_{j}\left(\boldsymbol{f}\right), \tag{6.2.1}$$

où $R_j\left(\boldsymbol{f}\right)$ est une matrice définie positive de dimension $I\times I$, appelée *matrice de covariance spatiale* de la source j pour l'index de fréquence $\boldsymbol{f}$. $P\left(\boldsymbol{f},\boldsymbol{n},j\right)$ désigne toujours la densité spectrale de la source j au point $(\boldsymbol{f},\boldsymbol{n})$.

Il est fréquent que des filtres de mélange réels ne respectent pas les hypothèses requises pour les traitements présentés en section 6.1.3.

Dans le domaine de la séparation de sources audio, la plupart des études dont j'ai connaissance qui font intervenir le modèle de rang plein [47, 50, 49, 52, 48, 51, 163] posent 6.2.1 en justifiant cette hypothèse par l'invocation de travaux précurseurs en acoustique statistique portant sur l'étude de longues réverbérations [93]. Ce modèle est de plus exprimé dans le domaine de la TFCT, ce qui revient à le restreindre au cas des PGLS, alors que les idées qui le sous-tendent sont en fait assez générales. Dans la section suivante, je propose de montrer comment le modèle diffus peut s'expliquer par l'abandon de l'hypothèse de *ponctualité* des sources.

6.2.2 Formalisation

Jusqu'à présent, j'ai toujours supposé qu'un signal source était l'unique réalisation d'un processus gaussien défini sur $\mathbb{T}$ et à valeurs dans $\mathbb{C}$. Dans ce cas, on parle de source *ponctuelle*. Cependant, il est possible qu'en réalité, ce ne soit pas une seule réalisation de cette source qui soit observée par le biais de son image, mais plusieurs. Dans ce cas, on parlera de sources *diffuses* et l'image d'une source s'exprime alors comme la superposition de toutes les contributions provenant de ses différentes réalisations, toutes supposées indépendantes.

Soit $\tilde{s}\left(\cdot,j\right)$ un des processus sources. Notons $\left\{\tilde{\boldsymbol{s}}\left(\cdot,q,j\right)\right\}_{q=1,\ldots,Q}$ un ensemble de Q réalisations de $\tilde{s}\left(\cdot,j\right)$, toutes indépendantes et données sur tout $\mathbb{T}$. Pour un ensemble T de points de $\mathbb{T}$, le modèle diffus revient à supposer que l'image $\tilde{\boldsymbol{y}}\left(T,\cdot,j\right)$ de la source est la superposition des mélanges convolutifs des Q réalisations $\tilde{\boldsymbol{s}}\left(\cdot,q,j\right)$, par des filtres de mélanges $f_{iq,j}\left(\tau\right)$:

$$\forall t\in T,\tilde{\boldsymbol{y}}\left(t,i,j\right)=\sum_{q=1}^{Q}\left(f_{iq,j}\left(\cdot\right)*\tilde{\boldsymbol{s}}\left(\cdot,q,j\right)\right)\left(t\right). \tag{6.2.2}$$

Dans le cas d'une source sonore, la structure vibrante ne se résume pas à un unique point de l'espace et donc à une unique réalisation du processus source. Au contraire, elle correspond à tout un ensemble de telles réalisations, chacune mixée d'une manière différente puisque située en un endroit particulier de l'espace.

Dans le cas où la source $\tilde{s}\left(\cdot,j\right)$ est un PGLS et si on fait l'hypothèse que tous les filtres de mélanges $f_{iq,j}\left(\tau\right)$ sont suffisamment courts, l'approximation 6.1.15 est valable et le modèle 6.2.2 peut s'exprimer dans le domaine de la TFCT comme :

$$\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},i,j\right)=\sum_{q=1}^{Q}F_{iq,j}\left(\boldsymbol{f}\right)\boldsymbol{s}\left(\boldsymbol{f},\boldsymbol{n},q,j\right), \tag{6.2.3}$$

où $\boldsymbol{s}\left(\cdot,\cdot,q,j\right)$ est la TFCT de la $q^{\text{ème}}$ réalisation de $\tilde{s}\left(\cdot,j\right)$, indépendante des autres. $F_{iq,j}\left(\boldsymbol{f}\right)$ donne la réponse en fréquence du filtre $f_{iq,j}\left(\tau\right)$, à l'indice $\boldsymbol{f}$. Si on regroupe tous les $\left\{F_{iq,j}\left(\boldsymbol{f}\right)\right\}_{i,q}$ dans la matrice $F_j\left(\boldsymbol{f}\right)$, de dimension $I\times Q$ et qu'on note $F_{\cdot q,j}\left(\boldsymbol{f}\right)$ sa $q^{\text{ème}}$ colonne, on peut calculer simplement la matrice de covariance $K\left(\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right),\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right)\right)$ de l'image de la source j prise

au point $(\boldsymbol{f}, \boldsymbol{n})$:

$$\begin{aligned} K\left(\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right),\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right)\right) &= \mathbb{E}\left[\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right)\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right)^{H}\right] \\ &= \mathbb{E}\left[\left(\sum_{q=1}^{Q} F_{\cdot q,j}\left(\boldsymbol{f}\right)\boldsymbol{s}\left(\boldsymbol{f},\boldsymbol{n},q,j\right)\right)\left(\sum_{q'=1}^{Q} F_{\cdot q',j}\left(\boldsymbol{f}\right)\boldsymbol{s}\left(\boldsymbol{f},\boldsymbol{n},q',j\right)\right)^{H}\right] \\ &= P\left(\boldsymbol{f},\boldsymbol{n},j\right)\sum_{q=1}^{Q} F_{\cdot q,j}\left(\boldsymbol{f}\right)F_{\cdot q,j}\left(\boldsymbol{f}\right)^{H} \qquad (6.2.4) \\ &= P\left(\boldsymbol{f},\boldsymbol{n},j\right)\underbrace{F_{j}\left(\boldsymbol{f}\right)F_{j}\left(\boldsymbol{f}\right)^{H}}_{R_{j}(\boldsymbol{f})}. \qquad (6.2.5) \end{aligned}$$

Le résultat de la ligne 6.2.4 s'obtient si on suppose que toutes les réalisations sont indépendantes et ont la même DSP. Comme on le voit, on retombe bien avec 6.2.5 sur le modèle diffus 6.2.1. On peut remarquer en outre que cette définition de la matrice de covariance spatiale garantit sa positivité.

Plusieurs remarques peuvent être faites sur ces résultats. Tout d'abord, si on ne considère qu'une seule réalisation des sources ($Q = 1$), on voit que le modèle diffus se confond avec le modèle convolutif. Cela parait naturel puisque ce dernier est restreint aux sources ponctuelles. Plus précisément, on a alors :

$$R_{j}\left(\boldsymbol{f}\right) = A_{j}\left(\boldsymbol{f}\right)A_{j}\left(\boldsymbol{f}\right)^{H}, \qquad (6.2.6)$$

où $A_{j}\left(\boldsymbol{f}\right)$ est la $j^{\text{ème}}$ colonne de la matrice de mélange $A\left(\boldsymbol{f}\right)$.

Ensuite, il faut souligner que chaque colonne de la matrice $F_{j}\left(\boldsymbol{f}\right)$ donne la répartition de la réalisation correspondante vers les I canaux de l'image. Si certaines de ces colonnes sont colinéaires, cela signifie que les points correspondants de la source proviennent de la même direction. A l'opposé, si elles sont indépendantes et se répartissent uniformément sur $\mathbb{C}^{I}$, alors la source proviendra de toutes les directions, et $R_{j}\left(\boldsymbol{f}\right)$ tendra vers la matrice identité. On peut interpréter de cette manière l'ajout d'un bruit additif 6.1.17 dans l'équation de mélange du modèle ponctuel.

Ces considérations sur le modèle diffus peuvent être considérablement étendues. Pour commencer, il n'est pas nécessaire de supposer que les sources sont des PGLS. L'idée de considérer l'image comme la superposition de contributions provenant de plusieurs réalisations indépendantes de la source ne dépend pas du modèle particulier choisi pour les sources. Par ailleurs, il est tout à fait possible d'élargir le cas d'un nombre fini de sources ponctuelles Q au cas d'une source présentant une densité de présence dans l'espace. Quoiqu'il en soit, ces extensions peuvent être abordées de manière élégante en utilisant le formalisme des processus gaussiens.

6.2.3 Séparation dans le cas diffus

On a vu que le modèle diffus se traduit par l'hypothèse que le signal image est la résultante de contributions provenant de plusieurs réalisations du processus source. Si les filtres de mélange $F_{j}\left(\boldsymbol{f}\right)$ de ces réalisations, tels que définis en 6.2.3 sont connus, récupérer chacune d'entre elles revient à procéder à une séparation dans le cas ponctuel tel qu'évoqué en section 6.1.3, à la seule différence que les DSP des sources à séparer sont les mêmes et qu'il faut utiliser les colonnes de $F_{j}\left(\boldsymbol{f}\right)$ en lieu et place des filtres de mélanges $A_{j}\left(\boldsymbol{f}\right)$.

Cependant, on considère la plupart du temps dans le cas d'un mélange diffus que l'objectif de la séparation de sources est la récupération des seules images [47, 50, 51, 49, 52, 48]. Comme nous l'avons déjà vu en section 6.1.2, cela se fait très simplement si on se rappelle que le mélange est par définition la simple somme des images 5.2.1, ce qui se traduit dans le cas des PGLS par :

$$\boldsymbol{x}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right) = \sum_{j=1}^{J}\boldsymbol{y}\left(\boldsymbol{f},\boldsymbol{n},\cdot,j\right). \qquad (6.2.7)$$

Puisqu'on connaît la matrice de covariance $R_j(\boldsymbol{f})$ de chacune des images, leur distribution *a posteriori* étant donné le mélange est :

$$\begin{bmatrix} \boldsymbol{y}(\boldsymbol{f},\boldsymbol{n},\cdot,1) \\ \vdots \\ \boldsymbol{y}(\boldsymbol{f},\boldsymbol{n},\cdot,J) \end{bmatrix} \mid \boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \sim \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}}, K_{\text{post}}\right), \tag{6.2.8}$$

avec

$$\boldsymbol{\mu}_{\text{post}} = \begin{bmatrix} P(\boldsymbol{f},\boldsymbol{n},1)\,R_1(\boldsymbol{f}) \\ \vdots \\ P(\boldsymbol{f},\boldsymbol{n},J)\,R_J(\boldsymbol{f}) \end{bmatrix} \left(\sum_{j=1}^{J} P(\boldsymbol{f},\boldsymbol{n},j)\,R_j(\boldsymbol{f})\right)^{-1} \boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot) \tag{6.2.9}$$

et

$$\begin{aligned} K_{\text{post}} = &\begin{bmatrix} P(\boldsymbol{f},\boldsymbol{n},1)\,R_1(\boldsymbol{f}) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & P(\boldsymbol{f},\boldsymbol{n},J)\,R_J(\boldsymbol{f}) \end{bmatrix} \\ &- \begin{bmatrix} P(\boldsymbol{f},\boldsymbol{n},1)\,R_1(\boldsymbol{f}) \\ \vdots \\ P(\boldsymbol{f},\boldsymbol{n},J)\,R_J(\boldsymbol{f}) \end{bmatrix} \left(\sum_{j=1}^{J} P(\boldsymbol{f},\boldsymbol{n},j)\,R_j(\boldsymbol{f})\right)^{-1} \\ &\times \left[\left(P(\boldsymbol{f},\boldsymbol{n},1)\,R_1(\boldsymbol{f})^H\right), \quad \ldots \quad , \left(P(\boldsymbol{f},\boldsymbol{n},J)\,R_J(\boldsymbol{f})^H\right) \right]. \end{aligned} \tag{6.2.10}$$

Comme d'habitude, les images recherchées sont estimées aux moindres carrés par $\boldsymbol{\mu}_{\text{post}}$ défini en 6.2.9.

6.3 Conclusion

Dans ce chapitre, j'ai présenté deux extensions successives du modèle simple de mixage linéaire instantané, vu au chapitre précédent. La première est le modèle convolutif et la deuxième le modèle diffus. J'ai montré comment les sources, entendues comme des réalisations de processus gaussiens, peuvent être estimées à partir de l'observation du mélange dans le cas convolutif et comment leurs images peuvent être séparées dans le cas diffus.

Ma présentation de ces modèles complexes de mixage ne s'est pas restreinte au seul cas des processus gaussiens localement stationnaires, comme c'est d'habitude le cas dans la littérature. Puisqu'il est cependant très utile dans les applications, je l'ai aussi traité en détail, toujours pour des sources définies sur $\mathbb{Z}^D$ et non pas seulement pour des séries temporelles ($D = 1$) comme on le fait habituellement [7]. J'ai de plus montré que le modèle diffus se traduit en pratique par l'abandon de l'hypothèse de source ponctuelle et je trouve que cette interprétation permet de mieux en saisir la portée. Dans tous les cas, j'ai fourni l'expression complète de la distribution *a posteriori* des signaux recherchés.

Comme on le voit, le formalisme gaussien offre une manière simple et efficace de séparer les sources ou leurs images à la condition de connaître leurs fonctions de moyenne et de covariance [8] ainsi que le procédé de mixage. Si les valeurs précises de ces paramètres sont inconnues, il est fréquent dans les applications pratiques d'avoir néanmoins à leur sujet certaines connaissances *a priori*. On va voir à présent qu'il existe des techniques pour estimer ces paramètres à partir de la seule observation des mélanges.

7. Il est cependant clair que cette prise en compte d'une dimension D quelconque du domaine de définition des sources reste presque transparente du point de vue des expressions. Elle ne se traduit à vrai dire que par l'utilisation de vecteurs de dimension $D \times 1$ pour désigner les indices de fréquences $\boldsymbol{f}$ et de trames $\boldsymbol{n}$.

8. En vertu du théorème de Wiener-Khinchin, connaître les DSP de PGLS est équivalent à connaître leurs fonctions de covariance.

Chapitre 7

Apprentissage des paramètres

7.1 Séparation aveugle, séparation informée

Lors de ma présentation des techniques de séparation mettant en œuvre le formalisme gaussien aux chapitres 5 et 6, j'ai toujours supposé connues les fonctions de moyenne et de covariance des sources, ainsi que les filtres de mélange dans le cas convolutif ou encore les matrices de covariance spatiales dans le cas diffus. Comme on l'a vu, ces connaissances permettent de procéder à la séparation des signaux d'une manière simple et directe.

Cependant, une telle hypothèse peut paraître extrêmement forte. Dans le domaine de la séparation de sources, il est plus fréquent de se concentrer sur la configuration *aveugle* [38], bien plus difficile et déjà évoquée en section 1.2.3, pour laquelle on ne suppose généralement qu'une connaissance très restreinte sur les signaux à séparer. Lorsqu'aucune connaissance sur les sources n'est disponible, on considère souvent qu'il est impossible de séparer des processus gaussiens.

Pourtant, le seul résultat négatif en la matière concerne les mélanges de processus gaussiens indépendants et identiquement distribués (i.i.d.), c'est-à-dire dont les fonctions de covariance $k_j(t, t')$ sont données par :

$$k_j(t, t') = \delta_{tt'}\sigma_j^2. \tag{7.1.1}$$

En effet, en utilisant des résultats établis par Darmois en 1953 [41], Comon a démontré qu'il est impossible de séparer des sources gaussiennes i.i.d. [37, 115, 39]. Partant de ce constat, une partie de l'effort de recherche dans le domaine de la séparation de sources s'est porté sur le cas de sources i.i.d. non gaussiennes, séparables en théorie. Pour des mélanges déterminés ou sur-déterminés, l'Analyse en Composantes Indépendantes (ACI [114, 109, 38]) a été proposée pour séparer des sources par l'exploitation des moments d'ordre supérieur [1]. Cette technique, ainsi que toutes les variantes et extensions qui en ont été proposées ont obtenu de grands succès dans de nombreux domaines applicatifs [38].

Cependant, ni les résultats de Darmois, ni ceux de la communauté de l'ACI ne sont incompatibles avec une hypothèse de gaussiannité faite sur les sources. En effet, ce ne sont que les sources gaussiennes i.i.d. qui ne sont pas séparables : il est possible en théorie de séparer des processus gaussiens, pour peu que leurs fonctions de covariance soient distinctes et différentes [2] de 7.1.1. Dans le cadre de mélanges déterminés ou sur-déterminés, Belouchrani *et al.* ont par exemple proposé l'algorithme SOBI (Second Order Blind Identification [16]), qui permet de procéder à la séparation de processus gaussiens colorés, c'est-à-dire stationnaires au sens large et dont les échantillons sont corrélés.

Dans le cas d'une séparation de sources sous-déterminée, on a vu en section 1.2.3 page 3 que la seule identification de la matrice de mélange ne suffit plus, rendant inopérantes les techniques classiques. Il est alors nécessaire pour procéder à la séparation de faire sur les sources des hypothèses

1. Pour une variable aléatoire x de moyenne μ et de variance σ^2, les moments d'ordre supérieur correspondent à $\mathbb{E}[(x-\mu)^p]$ pour $p > 2$. Si x est gaussienne, ils sont tous fonction de μ et σ et n'offrent pas d'information supplémentaire.

2. Plus précisément, la séparation peut se faire si une source au maximum est gaussienne de fonction de covariance 7.1.1.

supplémentaires. Par exemple, pour ce qui est des processus gaussiens, même colorés, on ne peut plus séparer les sources avec la seule matrice de mélange. En effet, en l'absence d'information, n'importe quelles fonctions de covariance qui s'ajoutent pour former la covariance du mélange fourniront des candidats valides pour la séparation et il y aura ainsi une multitude indénombrable de manières possibles de séparer le mélange selon le formalisme gaussien.

Pour illustrer cette situation, imaginons un instant un unique mélange $\tilde{x}(t)$, qu'on suppose être la somme instantanée de J processus gaussiens $\tilde{s}(\cdot, j)$ indépendants, dont on ne connaît pas les fonctions de covariance [3]. On suppose simplement que les sources $\tilde{s}(\cdot, j)$, tout comme le mélange, sont des PGLS. Comme on l'a vu en section 2.5.3, la DSP $P_x(\boldsymbol{f}, \boldsymbol{n})$ du mélange est donc la somme de celles des sources [4] $P_s(\boldsymbol{f}, \boldsymbol{n}, j)$:

$$\forall (\boldsymbol{f}, \boldsymbol{n}),\ P_x(\boldsymbol{f}, \boldsymbol{n}) = \sum_{j=1}^{J} P_s(\boldsymbol{f}, \boldsymbol{n}, j). \tag{7.1.2}$$

Muni de ce modèle 7.1.2 pour la DSP du mélange, on peut chercher à en déterminer les paramètres $P_s^\star$ les plus vraisemblables en utilisant le formalisme présenté en section 4.1 page 57 :

$$P_s^\star = \underset{P_s}{\operatorname{argmin}} \sum_{(\boldsymbol{f}, \boldsymbol{n})} d_0\left(v_x(\boldsymbol{f}, \boldsymbol{n}) \mid \sum_{j=1}^{J} P_s(\boldsymbol{f}, \boldsymbol{n}, j)\right), \tag{7.1.3}$$

où $v_x(\boldsymbol{f}, \boldsymbol{n})$ est le spectrogramme du mélange. C'est à ce moment de la procédure qu'on se retrouve coincé. En effet, n'importe quelle configuration de P_s telle que :

$$\forall (\boldsymbol{f}, \boldsymbol{n}),\ v_x(\boldsymbol{f}, \boldsymbol{n}) = \sum_{j=1}^{J} P_s(\boldsymbol{f}, \boldsymbol{n}, j)$$

minimise 7.1.3 et le problème n'a donc pas de solution unique. On peut faire le même constat d'échec du modèle gaussien pour la séparation aveugle sans faire l'hypothèse que les sources et les mélanges sont des PGLS. Une fois encore, on se retrouvera comme en 7.1.3 avec plus de paramètres à estimer que d'observations puisqu'il s'agira alors d'estimer des covariances $k_j(t, t')$ quelconques à partir de l'observation des $\tilde{x}(t)$.

Comme on le voit, ce que j'ai jusqu'ici présenté comme une force du modèle gaussien, à savoir le fait qu'à partir de sources gaussiennes, on obtient un mélange gaussien, a été surtout senti comme sa principale faiblesse. Dit autrement, le caractère gaussien a surtout été vu comme un état "absorbant" duquel l'ajout de nouvelles sources gaussiennes ne fait pas sortir et à partir duquel il est impossible à l'aveugle de distinguer les contributions de plusieurs sources gaussiennes. Pour toutes ces raisons, les approches aveugles se sont focalisées sur d'autres modèles, permettant une meilleure identifiabilité.

Sans faire d'hypothèses sur les fonctions de covariance et de moyenne des sources, c'est-à-dire dans le cas *aveugle*, on ne peut pas utiliser le formalisme gaussien pour la séparation.

Cependant, s'il existe bien des cas de figure où on n'a réellement aucune information sur les sources hormis la certitude que le signal observé est la somme de composantes indépendantes, il est en fait plutôt fréquent de disposer de certaines connaissances *a priori* sur les signaux mélangés. Je donne ci-dessous quelques exemples typiques de ce cas de figure :

- Il est fréquent de savoir dans une application de débruitage que le signal recherché est relativement lisse et mélangé avec un bruit additif blanc. Comme on l'a vu en section 2.3, ce genre de connaissance peut très bien se traduire par le choix d'une *famille* de fonctions de covariance pour les deux signaux. Dans cet exemple, on peut choisir la famille EC définie en 2.3.2 page 31 pour modéliser la première source et voir la deuxième comme un bruit additif de fonction de covariance $\sigma^2 \delta_{tt'}$.

3. Je suppose pour simplifier la discussion que les moyennes sont supposées connues et nulles.

4. Cela est dû, comme en section 2.5.4 page 39, au fait que les $P_s(\boldsymbol{f}, \boldsymbol{n}, j)$ sont leurs variances pour le point $(\boldsymbol{f}, \boldsymbol{n})$ et s'ajoutent pour former la variance $P_x(\boldsymbol{f}, \boldsymbol{n})$ du mélange.

– En audio, hormis le fait que les signaux sources sont très souvent correctement modélisés comme des PGLS, il est clair qu'ils sont susceptibles de présenter de fortes redondances, justifiant le choix d'un modèle paramétrique tel que le modèle NTF vu en section 4.3. Une fois encore, le problème de l'estimation des fonctions de covariance des sources, ici des DSP, devient celui de l'estimation d'un nombre bien plus restreint d'hyperparamètres.
– Les connaissances peuvent également porter sur le procédé de mélange, comme le modèle diffus 6.2.1 page 90, qu'on justifie par exemple par des considérations physiques [50, 49]. Dans ce cas, même si les fonctions de covariance des sources sont inconnues, l'estimation des paramètres du modèle est possible, grâce au couplage que le mixage introduit entre les différents canaux du mélange [5]. Bien entendu, le problème délicat de la convergence des algorithmes d'estimation y demeure centrale.
– Il est possible d'imaginer de nombreux autres types d'informations disponibles sur les signaux à séparer, sous la forme de données annexes qui y sont liées et qu'on peut voir comme fournissant autant de *variables explicatives* au phénomène complexe qu'on cherche à séparer. En audio, un thème de recherche porteur est par exemple la prise en compte de partitions pour aider la séparation [80, 95, 101, 100, 191]. Il est également envisageable d'utiliser des bases de données d'apprentissage pour apprendre certains modèles à utiliser lors de la séparation [162]. En particulier, on peut chercher à séparer un morceau en utilisant de nouvelles pistes séparées produites par des musiciens qui en ont fait une reprise [81].
– Enfin, il est possible de faire intervenir un utilisateur dans le processus de séparation [156, 57, 77], capable de remédier aux insuffisances d'un modèle et de l'orienter vers une solution de meilleure qualité. Développer des formalismes capables de faire intervenir une interaction homme-machine est un enjeu fort dans certains domaines comme dans le traitement du signal sonore, où les professionnels sont très demandeurs de techniques paramétrables sur lesquelles ils peuvent agir pour obtenir de meilleurs résultats.

Lors de mon travail à Audionamix, j'ai appris qu'il est souvent plus efficace d'inclure de bonnes connaissances *a priori* pour la séparation que d'utiliser un modèle plus complexe.

D'une manière générale, il parait clair aujourd'hui [214, 163] que toute prise en compte d'information disponible conduit à un gain en performance. On parle alors volontiers de séparation *informée*.

En tout état de cause, s'il s'avère inefficace dans le cas aveugle sous déterminé et i.i.d., le formalisme gaussien que je présente prend tout son sens dans le cas informé.

7.2 Formalisation

Le point commun de tous les exemples que j'ai donnés plus haut peut se comprendre dans le formalisme gaussien comme permettant de restreindre le nombre des possibilités pour les fonctions de moyenne et de covariance des sources. Pour la suite de cet exposé, je considèrerai le cas simple où les fonctions moyennes sont supposées connues. Un traitement similaire faisant intervenir des fonctions moyennes dont on suppose une forme paramétrique est fréquent dans le domaine des géostatistiques [44, 32], mais je ne le considèrerai pas dans mon exposé.

Dans une séparation *semi-informée*, les paramètres de modèles gaussiens sont remplacés par des représentants de modèles paramétriques.

De manière formelle, je dirai que l'apport d'une information *a priori* permet de remplacer la fonction de covariance quelconque $k_j(t,t')$ d'une source par un membre $k_j(t,t' \mid \theta_j)$ d'une certaine famille paramétrique connue. De cette manière, le problème de l'estimation des k_j devient celui, considérablement mieux posé en général, de l'estimation des paramètres θ_j. Dans l'exemple de séparation représenté en figure 5.1 page 77, seuls 5 hyperparamètres sont nécessaires pour séparer des centaines d'observations. J'utiliserai le terme de séparation

5. Dans la mesure où la seule hypothèse faite dans [49] sur les sources est qu'elles sont des PGLS, cette technique peut bien se ranger dans les méthodes *aveugles*, à l'instar de l'ACI qui suppose aussi connu le type de mélange.

de sources *semi-informée* pour désigner ce cas de figure où l'information permet seulement de restreindre la généralité des modèles par lesquels on rend compte des sources, par contraste avec le cas de la séparation *fortement informée*, que j'introduirai plus tard aux parties III et IV, pour lequel les paramètres peuvent en plus être appris à partir des sources elles-mêmes.

Dans le cas d'un mélange instantané et unique ($I = 1$) de PGLS par exemple, si on suppose qu'une information *a priori* est disponible qui permet de supposer que les DSP des sources sont membres de J familles paramétriques $\mathcal{P}_j$, le problème insoluble 7.1.3 devient celui de la recherche de $\theta^\star = \{\theta_1^\star, \ldots, \theta_J^\star\}$ tel que [6] :

$$\theta^\star = \underset{\theta}{\operatorname{argmin}} \sum_{(\boldsymbol{f},\boldsymbol{n})} d_0 \left(v_x\left(\boldsymbol{f},\boldsymbol{n}\right) \mid \sum_{j=1}^{J} \mathcal{P}_j\left(\boldsymbol{f},\boldsymbol{n} \mid \theta_j\right) \right), \tag{7.2.1}$$

qui peut avoir une solution, en fonction de la nature des modèles $\mathcal{P}_j$. Il est remarquable que la plupart des méthodes basées sur des modèles gaussiens en séparation de sources mono-capteur [18, 17, 30, 75, 55, 56, 101] se concentrent sur un problème du type de 7.2.1, avec des choix particuliers pour les modèles de sources $\mathcal{P}_j$.

Dans le cas de mélanges plus complexes ou lorsque plusieurs mélanges sont observés ($I > 1$), l'information disponible peut également concerner la nature du processus de mixage. Elle peut tout d'abord permettre de trancher entre les différents types de modèles de mixage présentés aux chapitres 5 et 6. Ensuite, dans le cas convolutif, il est fréquent de supposer que les sources ont des directions spatiales disjointes mais stables à la fois sur l'ensemble des indices fréquentiels $\boldsymbol{f}$ et des trames $\boldsymbol{n}$, ce qui permet de les séparer. De tels *a priori* sont en effet équivalents à fournir pour chaque source un vecteur de direction $\boldsymbol{d}_j \in \mathbb{C}^I$ auquel tous les $A_j\left(\boldsymbol{f}\right)$ sont colinéaires, réduisant d'autant le nombre de paramètres à estimer. De la même manière, le modèle diffus permet d'inclure une incertitude supplémentaire sur cette direction d'une source, par l'introduction d'un paramètre de *diffusion*, caractérisant l'écart attendu de l'observation à cette direction centrale [47, 49, 50]. La simple hypothèse que de telles directions existent permet à elle seule de rendre l'estimation des paramètres du modèle plus facile.

Une extension prometteuse de ce formalisme serait de ne pas considérer qu'un *a priori* permet de remplacer les fonctions de covariance par un modèle paramétrique, mais plutôt qu'elles sont la réalisation de processus aléatoires dont on cherche alors les hyperparamètres à partir des observations [45].

Comme on le voit, la présence d'une information peut souvent se traduire dans le cas gaussien par une réduction importante du nombre de paramètres à estimer et d'ainsi sortir de l'impasse de la situation aveugle. Il est très fréquent qu'une telle information soit disponible et l'objet de ce travail est de mettre en lumière le fait que le formalisme gaussien est idéal pour prendre en compte un très large ensemble de connaissances dans les modèles de séparation.

La présentation faite du formalisme de séparation par l'utilisation de processus gaussiens aux chapitres 5 et 6 a visé à montrer que si les paramètres du modèle sont connus, la séparation se fait de manière intégrée et efficace.
Par conséquent, le formalisme gaussien présente une disjonction complète entre le problème de l'estimation des paramètres des sources et celui de leur séparation étant donné le mélange. Cette disjonction est une des propriétés très avantageuses de l'approche.
Je vais maintenant donner la méthodologie générale par laquelle ces paramètres peuvent être estimés à partir du mélange.

7.3 Estimation des paramètres à partir du mélange

Pour peu qu'une information *a priori* permette de remplacer les fonctions de covariance k_j par des représentants de familles paramétriques $k_j\left(t, t' \mid \theta_j\right)$ dont le nombre de paramètres est suffisamment faible, l'ensemble des modèles présentés aux chapitres 5 et 6 pour la séparation de

6. Les familles paramétriques $\mathcal{P}_j$ ne sont pas nécessairement différentes.

sources peut faire l'objet d'une estimation des paramètres étant donné le mélange seulement. Dans ce cas, je parlerai de séparation *semi-informée*. La méthodologie dans ce but est toujours la même.

Dans le cas semi-informé, l'information annexe permet de réduire le nombre de paramètres à apprendre, mais leur estimation se fait à partir des mélanges.

Comme je l'ai souligné à de nombreuses reprises, tous les modèles de mixage que j'ai présentés produisent des mélanges qui sont des processus gaussiens si tel est le cas des sources. Si cette propriété, on l'a vu, est un inconvénient dans le cas aveugle, elle est intéressante dans le cas semi-informé. Pour chaque modèle de mixage, j'ai donné l'expression explicite de la matrice de covariance des mélanges. Les entrées de cette matrice de covariance dépendent à la fois des fonctions de covariance des sources et des paramètres de mixage. Dans le le cas où ces fonctions dépendent d'un certain lot Θ d'hyperparamètres, la matrice de covariance du mélange, que je noterai K_Θ, dépendra donc également de ces paramètres.

On peut appliquer exactement à l'identique le formalisme vu en section 2.4.2 page 34 pour l'apprentissage de Θ, appliqué cette fois au cas des mélanges $\tilde{x}$. Ainsi, on cherchera le lot $\Theta^\star$ de paramètres qui maximisera la vraisemblance des observations :

$$\Theta^\star = \underset{\Theta}{\operatorname{argmax}}\; p\left(\tilde{\boldsymbol{x}}\left(T,\cdot\right) \mid \Theta\right). \tag{7.3.1}$$

Si une distribution *a priori* $p\left(\Theta\right)$ est en outre disponible, le problème peut s'exprimer également comme un problème de maximum *a posteriori* :

$$\Theta^\star = \underset{\Theta}{\operatorname{argmax}}\; p\left(\tilde{\boldsymbol{x}}\left(T,\cdot\right) \mid \Theta\right) p\left(\Theta\right). \tag{7.3.2}$$

Quoiqu'il en soit, le cadre gaussien permet de donner à ces expressions une forme analytique, puisque la vraisemblance $p\left(\tilde{\boldsymbol{x}}\left(T,\cdot\right) \mid \Theta\right)$ des mélanges s'exprime en fonction de leur matrice de covariance K_Θ. Dans ces conditions, le problème 7.3.1 se traduit par [7] (voir l'équation 2.4.3 page 34) :

$$\Theta^\star = \underset{\Theta}{\operatorname{argmin}}\; \frac{1}{2}\left[\tilde{\boldsymbol{x}}\left(T,\cdot\right)^{H} K_\Theta \tilde{\boldsymbol{x}}\left(T,\cdot\right) + \ln\left|K_\Theta\right| + L\ln 2\pi\right], \tag{7.3.3}$$

auquel s'ajoute un terme $-\ln p\left(\Theta\right)$ dans le cas d'un maximum *a posteriori* 7.3.2.

Comme on le voit, l'estimation des paramètres Θ du modèle peut simplement être vue comme un problème d'optimisation dont la fonction de coût est donnée par la vraisemblance des mélanges [214, 29]. Il existe de nombreuses études portant sur l'estimation des hyperparamètres d'un modèle gaussien. Le lecteur intéressé pourra avec profit se référer à [178, 140, 185] pour des résumés synthétiques du sujet. En fonction de la nature des familles paramétriques $k_j\left(t,t' \mid \theta_j\right)$ et du modèle de mixage choisi, le problème 7.3.3 peut avoir ou pas une solution analytique ou encore disposer de propriétés de convexité intéressantes. Il faut ici remarquer que l'ensemble des fonctions de covariances présentées en section 2.3 ainsi que des modèles résumés au chapitre 3 peuvent être utilisés pour modéliser des sources, ainsi que toutes celles qu'on peut trouver dans la littérature [1, 124, 199, 139, 145, 206, 198, 197, 176].

Dans le but de résoudre le problème 7.3.3 dans le contexte particulier de la séparation de sources, il est également fréquent [50, 72, 155, 47] d'introduire explicitement les sources $\tilde{\boldsymbol{s}}\left(T,\cdot\right)$ ou les composantes de leurs images $\tilde{\boldsymbol{y}}\left(T,\cdot,j\right)$ comme autant de *variables latentes*, en particulier dans le cas où un modèle complexe de mixage est utilisé. En effet, le cadre probabiliste devient alors propice à l'application de l'algorithme d'Espérance-Maximisation (EM) [42], qui permet de trouver un minimum (local)) à 7.3.3.

Dans le cas très étudié en audio où les sources sont des PGLS, le problème admet souvent une expression plus simple, comme on l'a vu en 7.2.1 pour un unique mélange instantané ($I = 1$). Des exemples remarquables d'études pour la séparation de la voix rentrant dans ce cas de figure ont été proposés par Durrieu *et al.* et peuvent être trouvés dans [55, 56].

7. Comme pour 2.4.3 page 34, l'expression 7.3.3 est valide pour des processus à valeurs réelles. Pour des processus à valeur dans $\mathbb{C}$, il faut remplacer $L\ln 2\pi$ par $2L\ln\pi$ dans 7.3.3.

La conjonction d'un modèle de factorisation de DSP et du modèle de mélange convolutif a également fait l'objet de travaux célèbres [155, 72], tandis que l'utilisation du même modèle NTF de sources combiné à un processus de mixage diffus constitue ce qu'on peut aujourd'hui considérer comme l'état de l'art dans le domaine de la séparation de sources audio [7, 163].

La conjonction de l'utilisation du formalisme gaussien et d'informations *a priori* rend possible l'utilisation d'un vaste outillage probabiliste permettant d'estimer les hyperparamètres des sources et de mixage à partir du mélange.

Enfin, la prise en compte d'informations annexes comme la partition des signaux présents dans des mélanges musicaux a elle aussi fait l'objet de travaux prometteurs dans le cadre gaussien [101]. Elle se traduit souvent par l'utilisation d'un modèle NTF comme celui présenté en section 4.3, pour lequel chaque composante k correspond à une note, dont la partition permet de spécifier les activations en fonction des trames [65, 64]. Des modèles plus complets permettent de s'affranchir de la nécessité d'avoir un alignement parfait du morceau considéré avec sa partition [191].

Ainsi, il existe un nombre important de modèles considérés dans la littérature qui peuvent se comprendre comme des cas particuliers de séparation semi-informée de processus gaussiens. Dans ces modèles, les fonctions de covariance des sources sont supposées être membres de familles paramétriques, dont les paramètres sont estimés par maximisation de la vraisemblance des mélanges. En fonction du processus de mixage considéré et des familles particulières de fonctions de covariance, la procédure d'optimisation présente certaines simplifications qui sont exploitées pour aboutir à une estimation efficace.

Dans le cadre du traitement du signal audio, le formalisme gaussien est à ma connaissance le seul à permettre une séparation efficace sur des morceaux entiers, composés de dizaines de millions d'échantillons.

Dans la plupart des cas, les études se restreignent au cas de mélanges de PGLS pour $D = 1$. Le formalisme général proposé permet de réunir la plupart de ces approches comme autant de manifestations d'une stratégie commune. On constate donc que la séparation semi-informée prend une importance de plus en plus considérable dans la littérature, puisqu'elle permet d'inclure des connaissances dans des problèmes d'une grande complexité, qu'aucune approche aveugle n'avait permis d'aborder jusqu'à présent.

Quoi qu'il en soit, une fois les paramètres $\Theta^\star$ des sources et du mélange estimés, il suffit pour procéder à la séparation de remplacer les expressions des fonctions de covariance et celles des paramètres de mixage par les représentants $k\left(t, t' \mid \theta_j\right)$ et $A_j\left(\boldsymbol{f}\right)$ ou $R_j\left(\boldsymbol{f}\right)$ trouvés et de procéder au filtrage de Wiener correspondant. Cette procédure de séparation bénéficie en outre d'un caractère d'optimalité, puisque c'est elle qui minimise l'erreur quadratique des estimées étant donnés le mélange et le modèle choisis.

Chapitre 8

Applications semi-informées

Dans cet exposé, j'ai présenté le formalisme gaussien pour la séparation de sources. Dans ce but, j'ai montré comment les sources ou leurs images peuvent être récupérées efficacement à partir des mélanges, pour peu que leurs fonctions de moyenne et de covariance soient connues. S'il arrive fréquemment que ces fonctions ne soient pas connues précisément, on a vu au chapitre 7 qu'une connaissance *a priori* à leur sujet est néanmoins souvent disponible, ce qui permet de les approcher comme des membres de familles paramétriques. Dans ces conditions, il devient possible d'estimer la valeur des paramètres qui les caractérisent à partir de la seule observation des mélanges et donc de procéder à la séparation. De manière à le distinguer du cas de la séparation aveugle, ou aucune connaissance *a priori* n'est supposée sur les sources, j'ai appelé ce cas de figure le cas semi-informé.

Dans la mesure où le cas d'application que j'ai décidé de développer en détail est celui de la séparation informée, présentée plus loin, je n'évoque ici que deux applications de la séparation semi-informée.

Le cas semi-informé est important dans les applications et il a donné lieu à de très nombreuses études dont j'ai mentionné certaines au chapitre 7. En effet, il permet d'estimer des sources à partir de leur mélange si certaines connaissances à leur sujet sont disponibles. Dans un souci de concision, j'ai décidé de ne me concentrer dans cet exposé que sur deux des cas particuliers de séparation semi-informée que j'ai abordés au cours de mon doctorat. Dans le premier, qu'on trouvera en section 8.1, je montre comment il est possible de très simplement procéder à la séparation des sons percussifs d'enregistrements musicaux en utilisant le modèle gaussien. Dans le deuxième, que je présente en section 8.2, on verra qu'une séparation par processus gaussiens permet l'analyse de signaux complexes de captation de mouvements.

À ces études, il faudrait rajouter celles effectuées en partenariat avec Zafar Rafii [138], dans laquelle nous procédons à la séparation de la voix d'enregistrements musicaux, en faisant certaines hypothèses sur les DSP des signaux à séparer, ainsi que celles menées avec Benoit Fuentes [78] sur la séparation supervisée.

8.1 Séparation de rythmiques

8.1.1 Problématique

Dans cette section, j'applique le formalisme gaussien pour la séparation de sources au cas de la séparation des sons de rythmiques d'enregistrements musicaux polyphoniques. Par conséquent, les signaux considérés sont régulièrement échantillonnés et leur domaine de définition est $\mathbb{T} = \mathbb{Z}$.

La séparation des signaux percussifs d'enregistrements polyphoniques est une tâche difficile qui a déjà donné lieu à plusieurs études [99, 82, 73, 48]. Son objectif est de parvenir à séparer les rythmiques des autres sons présents dans un morceau de musique. Ce faisant, elle rend possible certains traitements, comme le rehaussement de la section rythmique d'un morceau ou bien l'application d'effets audio uniquement sur l'accompagnement mélodique. Je supposerai que l'objectif du traitement est la séparation de deux images : la première correspondra aux sources rythmiques, la deuxième aux sources non rythmiques.

En section 8.1.2, je présente le modèle par lequel j'ai abordé ce problème et je le mets brièvement en perspective par rapport aux autres méthodes de l'état de l'art. En section 8.1.3, je donne les résultats d'une étude comparative avec la technique proposée par GILLET et RICHARD dans [82].

8.1.2 Modèle

Les I mélanges observés $\tilde{x}(t,\cdot)$ sont supposés être formés par mélange linéaire instantané d'un groupe $\tilde{s}$ de $J_p + J_m$ processus gaussiens indépendants. Alors que les premières J_p sources correspondent aux sons percussifs, les autres J_m sources correspondent aux sons non-percussifs, que j'appellerai *mélodiques*. Ces sources sont mixées pour obtenir un mélange multicanal $\tilde{x}$ observé :

$$\forall (t,i), \tilde{x}(t,\cdot) = \tilde{s}(t,\cdot) A^\top$$

où A est la matrice de mélange (réelle) des sources, de dimension $I \times (J_p + J_m)$. Comme cela est très souvent valide pour des signaux audio, je suppose que toutes les sources de $\tilde{s}$ sont des PGLS. Ainsi, les résultats de la section 5.3 page 79 s'appliquent et on a :

$$\forall (f,n), \boldsymbol{x}(f,n,\cdot) = A\boldsymbol{s}(f,n,\cdot).$$

Dans ces conditions, la DSP P_x du mélange est obtenue facilement par :

$$\forall (f,n,i), P_x(f,n,i) = \sum_{j=1}^{J_p+J_m} A_{ij}^2 P_s(f,n,j).$$

où P_s désigne la DSP des sources. Comme on le voit, la DSP du mélange est simplement obtenue comme une combinaison linéaire de celle des sources, comme on pouvait s'y attendre compte tenu du modèle de mélange linéaire instantané choisi.

Dans un cadre de séparation semi-informée, je vais à présent choisir un modèle pour la DSP des sources. Dans cette étude, je choisis pour chaque source un modèle très simple : je suppose sa DSP constante d'une trame à l'autre, à un coefficient amplificateur près qui dépend de la trame considérée. L'idée sous-jacente est de considérer chaque source comme un son qui présente un timbre stable, mais qui est activé ou pas au cours du morceau. Je pose ainsi :

$$\forall (f,n,j), P_s(f,n,j) = W(f,j) H(n,j), \tag{8.1.1}$$

avec W et H deux matrices non-négatives de dimension $F \times (J_p + J_m)$ et $N \times (J_p + J_m)$, où je rappelle que F et N désignent respectivement le nombre d'indices fréquentiels et de trames des TFCT des signaux. Par conséquent, la DSP des mélanges est donnée par :

$$\forall (f,n,i), P_x(f,n,i) = \sum_{j=1}^{J_p+J_m} A_{ij}^2 W(f,j) H(n,j). \tag{8.1.2}$$

Si on pose

$$Q_{ij} = A_{ij}^2,$$

on reconnaît en 8.1.2 un modèle NTF pour le mélange, identique à celui présenté en section 4.3 et adopté dans de nombreuses études [100, 75, 69, 155]. Son apprentissage peut donc être mené en utilisant sur le mélange l'algorithme 4.1, présenté en section 4.3.2.

Comme on le voit, aucune distinction particulière n'a pour l'instant été faite entre les sources percussives et les sources mélodiques, auxquelles j'ai assigné le même modèle. C'est sur ce point que peuvent se distinguer les différentes études qui adoptent un modèle NTF pour effectuer la séparation de rythmiques [82, 99, 131]. Alors que dans [99], HÉLEN et VIRTANEN effectuent d'abord une décomposition NMF et regroupent *a posteriori* les composantes rythmiques par classification des formes spectrales W obtenues par rapport à celles apprises sur une base de données, GILLET et RICHARD démontrent dans [82] de meilleures performances avec le même genre d'approche, à la différence que l'algorithme NMF est initialisé en utilisant des spectres appris sur une base de données d'enregistrements percussifs [1] plutôt qu'aléatoirement. Dans tous les cas, les composantes correspondant aux sources mélodiques sont initialisées aléatoirement.

1. Telle qu'ENST DRUMS `www.tsi.telecom-paristech.fr/aao/?p=152`

Alors que la plupart des études de l'état de l'art mettant en œuvre un modèle NMF se sont plutôt focalisées sur l'utilisation d'un *a priori* concernant les formes spectrales utilisées, j'ai constaté qu'il est souvent plus efficace de disposer d'une bonne initialisation des paramètres d'activation H. Depuis la publication [131] correspondant au modèle que je présente ici, plusieurs études [156, 65, 64] ont fait le même constat.

Dans cette expérience, plutôt que de me focaliser sur la matrice W, qui contient les formes spectrales des sources, très variables d'un morceau à l'autre, je me suis plutôt intéressé à la signification des entrées de la matrice H de leurs activations. L'entrée $H(n, j)$ peut être comprise comme le gain de la source j pour la trame n. Pour une source percussive, elle est donc liée à la présence d'une percussion dans cette trame. Une bonne manière d'initialiser cette valeur est de simplement utiliser un *détecteur d'onsets*, tel que celui présenté dans [58]. En effet, un détecteur d'onsets est un algorithme capable d'identifier où une transition soudaine est présente, comme une percussion dans un contexte musical. Pour ce faire, il produit pour chaque trame une valeur comprise entre 0 et 1 appelée fonction d'onset, que j'ai utilisée comme initialisation pour les activations H des sources percussives dans l'algorithme 4.1. Puisque l'algorithme de [58] est capable de procéder à une détection d'onsets selon plusieurs bandes de fréquence, j'ai mis à profit cette propriété pour initialiser différemment des sources correspondant à des percussions graves et à des percussions aigües. Les activations des sources mélodiques, ainsi que les formes spectrales W de toutes les sources, sont initialisées aléatoirement. En figure 8.1, j'ai représenté un exemple de fonction de détection renvoyée par l'algorithme [58] et utilisé pour l'initialisation de la NTF.

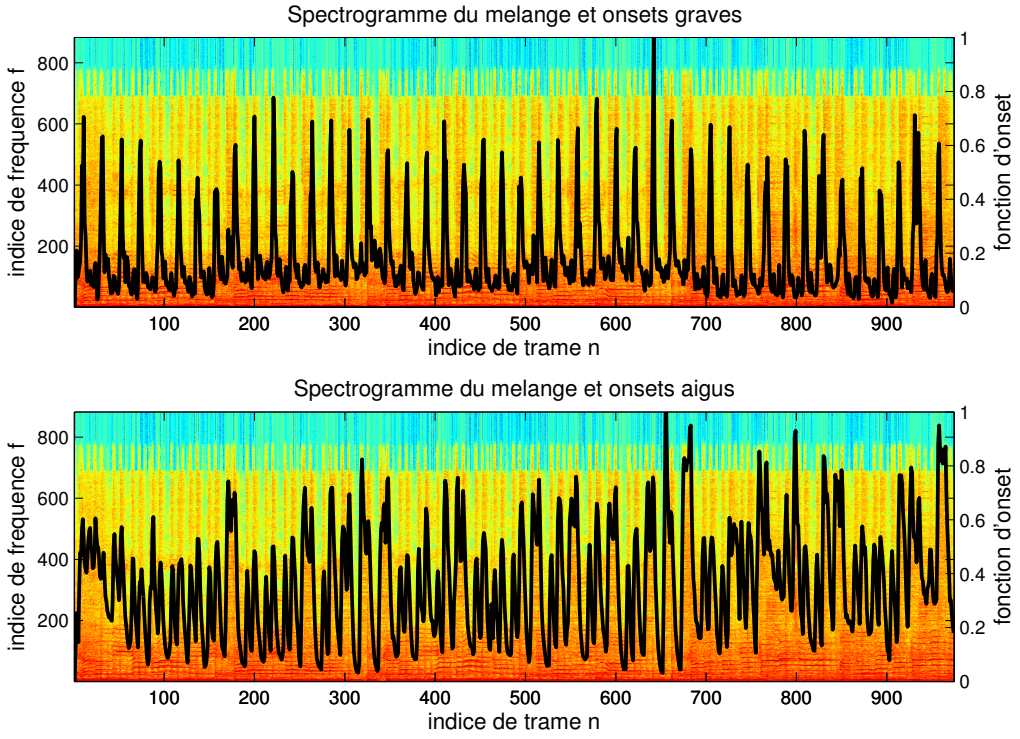

FIGURE 8.1: Exemples de détections d'onsets. Pour un fichier sonore monophonique donné, j'ai superposé le spectrogramme du signal avec la fonction d'onset retournée par l'algorithme [58]. En haut, la détection est faite sur la bande de fréquence [20 ; 300]Hz. En bas, pour la bande [10 ; 15]kHz. Ces valeurs sont utilisées pour l'initialisation du modèle NTF 8.1.2.

Une fois les paramètres du modèle NTF appris, il ne reste plus qu'à utiliser les résultats de la section 6.2.3 page 91 pour procéder à la séparation. En effet, notre objectif est de récupérer la

somme des images des J_p sources percussives. On obtient des images $\widehat{\boldsymbol{y}}\left(f,n,\cdot,\mathbb{N}_{J_P}\right)$ estimées dont l'expression est :

$$\widehat{\boldsymbol{y}}\left(f,n,\cdot,\mathbb{N}_{J_P}\right)=\left[\begin{array}{c}P_s\left(f,n,1\right)A_1A_1^\top\\ \vdots\\ P_s\left(f,n,J_p\right)A_{J_p}A_{J_p}^\top\end{array}\right]\left(\sum_{j=1}^{J_p+J_m}P_s\left(f,n,1\right)A_jA_j^\top\right)^{-1}\boldsymbol{x}\left(f,n,\cdot\right).$$

En utilisant l'expression 8.1.1 des DSP P_s des sources et en remplaçant A_{ij} par $\sqrt{Q_{ij}}$, l'estimée $\widehat{\boldsymbol{y}_p}\left(\cdot,\cdot,i\right)$ de la TFCT du $i^{\text{ème}}$ canal de la somme des images des sources percussives peut donc se réécrire de manière plus concise :

$$\widehat{\boldsymbol{y}_p}\left(\cdot,\cdot,i\right)=\frac{W_{1\cdots J_p}\mathrm{diag}\left(Q_{i,1\cdots J_p}\right)H_{1\cdots J_p}^\top}{W\mathrm{diag}\left(Q_{i,\cdot}\right)H^\top}\cdot\boldsymbol{x}\left(\cdot,\cdot,i\right), \tag{8.1.3}$$

où $M_{1\cdots P}$ désigne la sous-matrice formée des P premières colonnes d'une matrice M quelconque, tandis que $Q_{i,1\cdots J_P}$ désigne le vecteur de dimension $1\times J_p$, formé des J_p premières entrées de la $i^{\text{ème}}$ ligne de Q. Comme on le voit, l'expression 8.1.3 est très simple à calculer et fait intervenir $\mathcal{O}\left(FN\right)$ opérations. Les estimées temporelles $\tilde{y}_p\left(t,\cdot\right)$ de l'image percussive s'obtiennent alors facilement par TFCT inverse.

Si j'ai présenté les études [99, 82] comme mettant en œuvre le formalisme gaussien, ce n'est pas ainsi que le problème y est formulé, mais plutôt sur une base heuristique.

8.1.3 Résultats

La méthode proposée a été testée sur 10 extraits d'une durée de 30 secondes, échantillonnés à 44.1kHz et tirés de la base de données QUASI pour la séparation de sources [2]. Ces extraits sont de plusieurs genres musicaux différents : pop, electropop, rock, reggae et bossa. Pour chacun d'entre eux, les pistes séparées sont connues et utilisées à des fins d'évaluation des résultats de séparation. Cependant, l'algorithme de séparation n'a accès qu'à leur mélange. De manière à pouvoir appliquer l'algorithme [82] sur les mêmes données, j'ai considéré le cas d'un mélange monophonique pour les évaluations ($I=1$). En moyenne, l'amplitude relative des sources percussives dans le mélange par rapport aux sources mélodiques a été défini à -6dB.

Pour l'évaluation, j'ai comparé les performances de la méthode proposée avec celle de [82], en utilisant les métriques de BSSEval [212], désormais classiques dans le domaine de la séparation de sources audio. Ces métriques incluent le rapport Source à Distorsion (SDR), le rapport Source à Artéfacts (SAR) et enfin le rapport Source à Interférence (SIR), sur lesquels je reviendrai plus tard en section 12.1.2 page 150. Pour l'heure, il me suffit de mentionner qu'elles peuvent être comprises comme différents types de rapports signal à bruit des sources estimées par rapport aux sources originales. Plus les scores sont élevés, plus la séparation est bonne. Les résultats de cette évaluation sont donnés sur la figure 8.2.

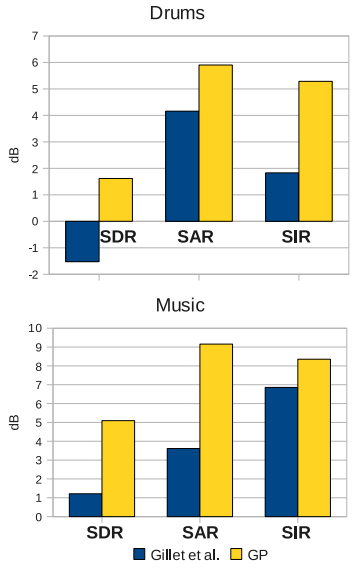

FIGURE 8.2: Évaluation de la qualité de la séparation pour l'extraction de la piste de rythmique (en haut) et de la piste mélodique (en bas). Plus les scores sont élevés, plus le résultat est bon (d'après [131]).

Sur la figure 8.2, on peut constater que l'approche proposée parvient très bien à séparer le signal percussif, dans un nombre considérable de genres différents de musique. En outre, elle permet de récupérer correctement le signal mélodique. Une propriété intéressante de cette technique est qu'elle

2. http://www.tsi.telecom-paristech.fr/aao/en/2012/03/12/quasi/

est assez rapide, puisqu'elle permet d'effectuer à la fois l'estimation des paramètres et la séparation en 30s environ pour un extrait de 30s, alors que 300s sont nécessaires pour [82]. J'ai diffusé une implémentation en MATLAB et en PYTHON de cette méthode, ainsi qu'une interface utilisateur permettant de la mettre en œuvre facilement.

8.2 Analyse de mouvements de danse

8.2.1 Problématique

Le Grand Challenge Huawei/3DLife [3] porte sur un scénario d'enseignement de la danse, où le cours est donné en ligne par un professeur professionnel de salsa. Il est accompagné d'une base de données multimodale très riche, composée d'enregistrements synchronisés provenant d'un réseau de caméras et de microphones, d'une captation par une MICROSOFT KINECT, de capteurs d'inertie fixés aux sujets, etc. [62]. Les mouvements du professeur et des élèves doivent être analysés automatiquement et affichés sous la forme d'avatars dans un environnement en ligne, permettant une interaction entre les participants.

Dans ce scénario, un objectif important des traitements est la reconnaissance et l'analyse des pas de danse effectués par un élève. L'accomplissement de cette tâche permet au professeur de détecter d'éventuelles erreurs ainsi que de suggérer des améliorations. Il est possible d'aborder ce problème comme la décomposition de la chorégraphie observée en mouvements élémentaires, de manière à simplifier sa classification dans une base de données de chorégraphies de référence. Certaines études reliées à cette problématique ont été menées dans la littérature. Dans [190, 189], les auteurs introduisent une segmentation des mouvements basée sur l'analyse de la musique jouée lors de leur exécution. Cette segmentation produit une séquence de *mouvements élémentaires*. Dans [143], une représentation fréquentielle du mouvement est obtenue par le biais d'une analyse en ondelettes. Enfin, une décomposition en valeurs propres est utilisée dans [43] qui a pour objectif la décomposition hiérarchique du mouvement de danse.

Dans [133], nous nous sommes attachés à montrer que le formalisme gaussien pour la séparation de sources semi-informée fournit un angle d'approche intéressant pour la décomposition d'un mouvement en composantes élémentaires. Il est en effet possible d'interpréter une chorégraphie observée comme un mélange de composantes indépendantes et il est alors légitime de chercher à estimer ces composantes par des techniques de séparation de sources. On verra que le formalisme gaussien permet très facilement d'incorporer dans le modèle des connaissances *a priori* sur ces composantes à extraire, rendant possible la séparation d'un phénomène particulier.

Le reste de cette section est structuré de la manière suivante. Pour commencer, je présente en section 8.2.2 les données sur lesquelles nous avons cherché à effectuer une séparation. Ensuite, je précise en section 8.2.3 le formalisme gaussien par lequel nous avons abordé le problème. Enfin, je donne quelques résultats préliminaires en section 8.2.4, où on constate que l'approche proposée permet de correctement identifier des mouvements élémentaires à partir de la captation KINECT.

3. http://www.acmmm12.org/3dlife-huawei-challenge-realistic-interaction-in-onlinevirtual-environments/

8.2.2 Données articulatoires 3D

Les images de profondeur enregistrées par la Microsoft Kinect sont exploitées par le biais de la bibliothèque OpenNI SDK2 [4]. Cette API [5] implémente une fonction de suivi qui permet d'obtenir la position au cours du temps de $P = 17$ articulations du danseur, dont la tête, le cou, le torse, les épaules, les coudes, les poignets, les hanches, les genoux et les pieds. Sur la figure 8.3, j'ai affiché sous forme de squelette la position de toutes ces articulations pour une trame donnée d'un des enregistrements du corpus. C'est l'ensemble de ces données articulatoires en 3D que nous avons utilisé pour l'analyse du mouvement.

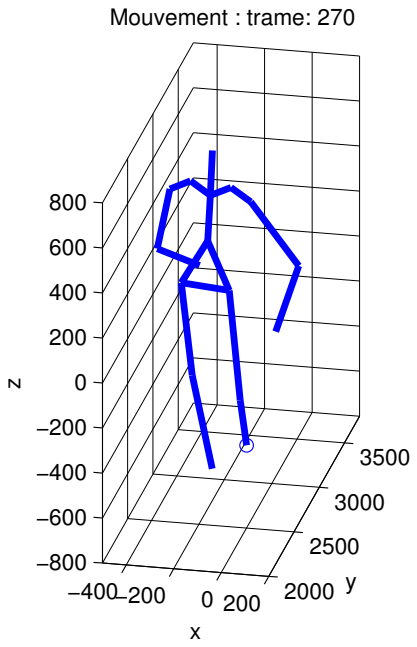

Figure 8.3: Position des 17 articulations d'un danseur pour une trame donnée.

Dans le cadre général que je me suis fixé dans ma présentation des processus gaussiens en partie I, les signaux étudiés sont des fonctions, définies sur un espace quelconque et à valeurs dans $\mathbb{C}$. Dans notre cas particulier, les données articulatoires observées donnent, pour chaque trame et chacune des P articulations sa position dans l'espace. Ainsi, on peut définir le signal $\tilde{z}$ observé comme une fonction définie sur

$$\mathbb{T} = \underbrace{\mathbb{N}}_{\text{trame}} \times \underbrace{\mathbb{N}_P}_{\text{articulation}} \times \underbrace{\mathbb{N}_3}_{\text{dimension}} \tag{8.2.1}$$

et à valeurs dans $\mathbb{R}$. Ainsi, $\tilde{x}\left((230, 8, 3)\right)$ donnera l'élevation ($3^{\text{ème}}$ coordonnée z) de la $8^{\text{ème}}$ articulation pour la $230^{\text{ème}}$ trame, c'est-à-dire pour le point $(230, 8, 3)$ de $\mathbb{T}$. Un point de $\mathbb{T}$ sera désigné indifféremment t ou (n, a, d).

Pour chaque enregistrement du corpus, on dispose d'une observation $\tilde{\boldsymbol{z}}\left(T\right)$ composé de N trames, pour lesquelles on a la position spatiale de toutes les articulations. On a donc $L = 3NP$.

Le travail présenté ici a été effectué en collaboration avec Angélique Drémeau et Slim Essid, que je remercie de m'avoir impliqué dans cette étude. Le contenu de cette section est largement inspiré de notre article commun [133].

8.2.3 Modèle

À l'équilibre, un danseur a une position qui dépend de sa morphologie et qui est indépendante du temps. Je noterai $\tilde{x}_0\left(a, d\right)$ cette position, qui donne l'emplacement spatial au repos de chaque articulation. Lors du mouvement, on peut voir le signal $\tilde{z}$ observé comme la somme de la position d'équilibre et d'un terme de *mouvement* $\tilde{x}$:

$$\forall\left(n, a, d\right) \in \mathbb{T},\ \tilde{z}\left(n, a, d\right) = \tilde{x}_0\left(a, d\right) + \tilde{x}\left(n, a, d\right). \tag{8.2.2}$$

En pratique, pour une observation $\tilde{\boldsymbol{z}}$ donnée, $\tilde{x}_0$ est pris comme la moyenne sur l'ensemble des trames de la position de chaque articulation :

$$\tilde{x}_0\left(a, d\right) = \mathbb{E}_n\left[\tilde{\boldsymbol{z}}\left(n, a, d\right)\right],$$

ce qui permet de considérer, après centrage des données, qu'on observe directement le terme de mouvement $\tilde{x}$.

4. `www.openni.org`
5. Application Programming Interface

Le modèle simple que nous avons proposé dans [133] revient à considérer que $\tilde{x}$ est la somme de J composantes latentes $\tilde{s}$:

$$\forall t \in \mathbb{T},\ \tilde{x} = \sum_{j=1}^{J} \tilde{s}\,(t, j)\,. \tag{8.2.3}$$

Autrement dit, le mouvement observé est modélisé comme la résultante de J mouvements élémentaires qui s'ajoutent pour former l'observation. Dans le contexte de l'analyse d'une chorégraphie, cette hypothèse se justifie en disant que des mouvements rythmiques à différentes échelles de temps se superposent pour former le mouvement final. Dans l'exemple d'application que nous avons envisagé, nous avons cherché à extraire $J = 4$ composantes latentes de l'observation. La première correspond aux variations lentes de la position du danseur autour de l'équilibre. La deuxième et troisième correspondent à des mouvements périodiques à long terme et à court terme respectivement, tandis que la dernière est imprévisible temporellement et peut donc être vue comme un terme d'hésitation.

Contrairement au cas audio, les signaux extraits ne sont pas réellement émis par des sources. Ce sont plutôt des *composantes latentes*, utilisées à des fin d'analyse des signaux.

Les J composantes latentes sont supposées être des processus gaussiens indépendants, de moyennes nulles et de fonctions de covariance $k_j\,(t, t')$, connues. Munis de ces informations, les résultats de la section 5.1 peuvent être utilisés directement, en prenant $T' = T$, pour obtenir une estimation des composantes pour tous les points T observés.

La définition des fonctions de covariance des composantes est le seul élément encore manquant pour pouvoir appliquer le formalisme gaussien à ce problème de séparation de sources. Dans ce travail assez préliminaire, les fonctions de covariance considérées sont séparables, comme défini en section 2.3 page 28 et sont données par :

$$k_j\left((n, a, d), (n', a', d')\right) = k_j^{\text{temp}}\,(n, n')\,k_j^{\text{art}}\left((a, d), (a', d')\right),$$

où k_j^{temp} est une covariance *temporelle*, qui ne dépend que de l'écart de trame entre les points considérés, tandis que k_j^{art} est une covariance *articulatoire*, qui ne dépend que des deux articulations et dimensions considérées.

La covariance temporelle choisie pour toutes les composantes est la fonction pseudo-périodique 2.3.5 page 33, représentée en figure 2.6 et dont je rappelle ci-dessous l'expression :

$$k_j^{\text{temp}}\,(n, n') = \sigma_j^2 \exp\left(-\frac{2\sin^2\frac{\pi(n-n')}{T_j}}{l_j^2} - \frac{(n-n')^2}{2\lambda_j^2}\right).$$

Comme on l'a vu en section 2.3.4, cette fonction permet de modéliser des signaux pseudo-périodiques de période T_j, où l_j correspond à un paramètre de stabilité au sein de chaque période, tandis que λ_j est une longueur caractéristique, telle que deux points séparés de plus de $2\lambda_j$ trames peuvent être considérés comme indépendants. Enfin, σ_j est un paramètre d'énergie, donnant l'amplitude attendue de chaque composante.

- La première composante $j = 1$ rend compte des mouvements lents du danseur autour de son point d'équilibre et on ne lui suppose pas de périodicité particulière. Par conséquent, nous avons choisi $T_1 = \infty$ et $l_1 = \infty$ pour annuler la composante périodique et se ramener à la fonction EC 2.2.13 page 26. Par contre, la longueur caractéristique λ_1 est prise assez grande, de manière à forcer le signal correspondant à avoir une corrélation à long terme. Puisque la fréquence d'échantillonnage de la Kinect est de 30 trames par secondes, une valeur $\lambda_1 = 70$ est adéquate.
- Les composantes $j = 2$ et $j = 3$ correspondent à des mouvements périodiques. En effet, le danseur est en général synchronisé avec la musique et nous avons supposé qu'une partie de son mouvement au moins tâchera d'en suivre le rythme. Par conséquent, nous avons choisi $T_2 = 8b$ et $T_3 = 4b$, où b désigne la pulsation du morceau sur lequel est effectuée la chorégraphie. Cette pulsation peut soit être celle indiquée dans la base de données accompagnant les enregistrements, soit être extraite du signal musical en utilisant un algorithme

spécialisé [117]. Pour permettre une grande variabilité des composantes au sein d'une période, les paramètres de stabilité l_1 et l_2 ont reçu la valeur assez faible de 0.1. Les longueurs caractéristiques λ_1 et λ_2 ont été choisies de telle manière à avoir une dépendance des échantillons sur quelques périodes seulement.
– Le choix des hyperparamètres de la dernière composante ont été pris de telle manière à ce qu'elle devienne équivalente à la fonction *blanche* $\sigma^2\delta_{nn'}$, permettant de modéliser un signal décorrélé temporellement.

La fonction de covariance articulatoire indique la corrélation attendue entre la localisation spatiale des articulations, indépendamment du temps. Certaines considérations physiques pourraient être introduites ici de manière à en choisir une expression analytique comme pour la covariance temporelle, mais nous avons choisi dans [133] une autre stratégie, basée sur une courte procédure itérative.

Dans cette technique, k_j^{art} est d'abord initialisée par la fonction blanche $\delta_{(a,d)(a',d')}$ et les composantes $\tilde{s}$ sont ainsi estimées en se basant uniquement sur leur structure temporelle. Ensuite, munis de ces premières estimées, nous avons choisi d'utiliser pour $k_j^{\text{art}}\left((a,d),(a',d')\right)$ la corrélation empirique de $\tilde{s}_j\left(\cdot,a,d\right)$ et de $\tilde{s}\left(\cdot,a',d'\right)$:

$$k_j^{\text{art}}\left((a,d),(a',d')\right) = \frac{\frac{1}{N}\sum_n \left[\tilde{s}_j\left(n,a,d\right)\tilde{s}_j\left(n,a',d'\right)\right]}{\sqrt{\left[\frac{1}{N}\sum_n \tilde{s}_j\left(n,a,d\right)^2\right]\left[\frac{1}{N}\sum_n \tilde{s}_j\left(n,a',d'\right)^2\right]}}. \tag{8.2.4}$$

Avec cette nouvelle expression de la covariance spatiale de chaque composante, nous avons procédé à une deuxième séparation de l'observation, donnant le résultat final.

8.2.4 Résultats et perspectives

La stratégie décrite plus haut a été appliquée à plusieurs captations Kinect de mouvements de danse. Elle permet de décomposer le mouvement renvoyé par OpenNI comme une somme de 4 mouvements élémentaires qui correspondent bien aux attentes. Une illustration de cette décomposition est donnée en figure 8.4 pour une des coordonnées d'une des articulations d'un danseur. Comme on peut le constater, les 4 composantes extraites présentent bien les caractéristiques désirées : la première, très lisse, donne les lentes variations, tandis que les deux suivantes, périodiques, permettent de constater que c'est surtout à l'échelle de la mesure qu'est synchronisée la danse. La dernière composante peut être interprétée comme donnant les fluctuations aléatoires du danseur autour de ses déplacements réguliers.

Il est possible d'envisager de nombreuses applications à une telle décomposition dans le cadre de l'analyse de mouvements de danse. Dans notre étude préliminaire [133], nous avons considéré un problème de lissage, qui consiste à supprimer des observations les erreurs de suivi de l'algorithme d'OpenNI, ou bien à corriger les hésitations du danseur. Nous avons simplement abordé ce problème par la suppression de la composante aléatoire des observations. Il faut noter qu'un tel traitement n'est pas équivalent à un simple filtrage passe-bas des signaux, puisque des mouvements rapides sont bel et bien conservés dans les composantes périodiques. J'ai illustré un tel traitement en figure 8.5. Il est possible de modifier les paramètres de l'analyse de manière à favoriser l'isolation de certains phénomènes qu'on cherche à mettre en lumière.

Si cette étude est résolument préliminaire, elle démontre néanmoins l'intérêt d'avoir proposé un formalisme aussi général pour la séparation de sources, qui peut être réutilisé à l'identique dans des cas de séparation encore inédits. On peut noter en outre qu'une telle approche permet non seulement la séparation, mais également l'interpolation des données ou la prédiction, puisque rien n'oblige l'inférence à être effectuée sur les mêmes points que ceux de l'observation. Une conséquence directe de cette remarque est qu'il est possible avec le même modèle de déterminer la valeur de certaines articulations étant données d'autres, ou bien de prédire leur position sur des trames non observées. En outre, ces prédictions sont accompagnées d'une estimation de leur incertitude, par

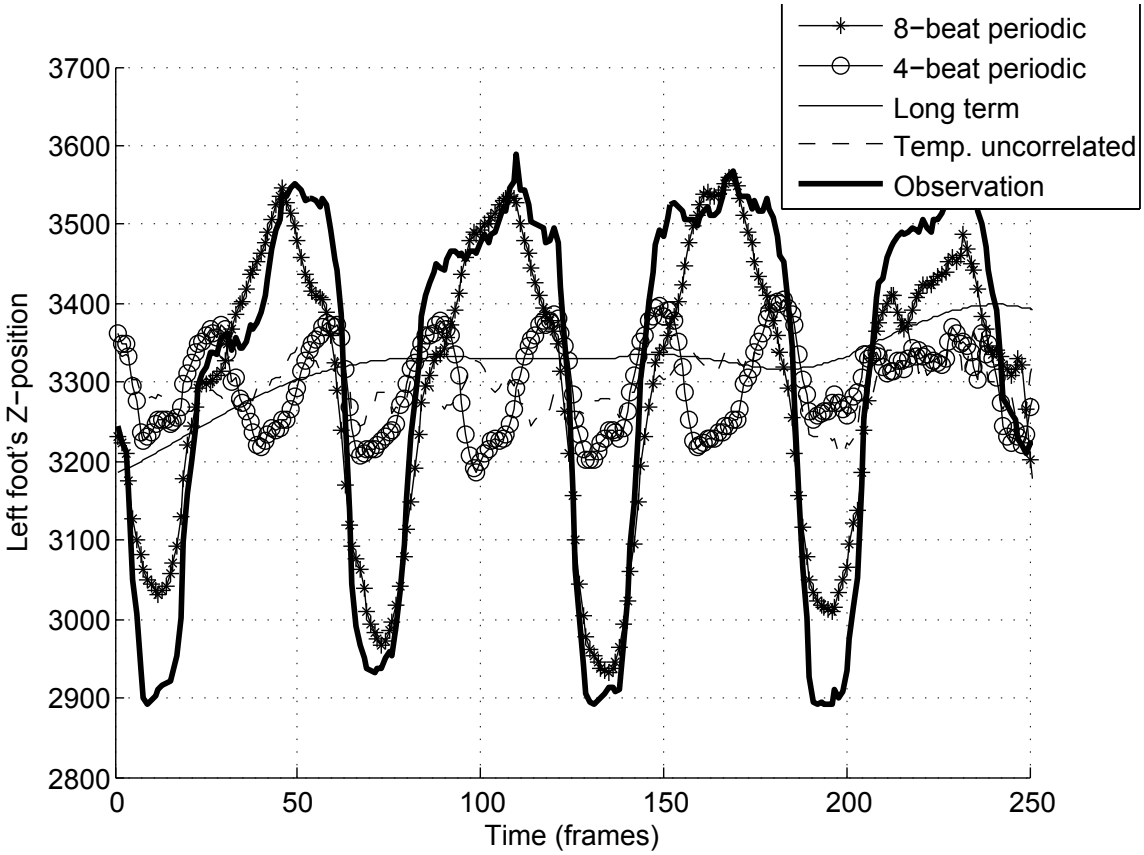

FIGURE 8.4: Exemple de décomposition obtenue pour l'élevation du pied gauche de Thomas, un des sujets de l'expérience (d'après [133]).

le biais de la matrice de covariance *a posteriori*.

Appliquer de tels modèles pour l'étude du mouvement me paraît une direction prometteuse et je pense que leur extension à des mixages complexes tels que ceux décrits au chapitre 6 permettrait d'introduire une notion de déphasage entre les articulations, qui semble importante par exemple pour l'étude de la marche ou de la course.

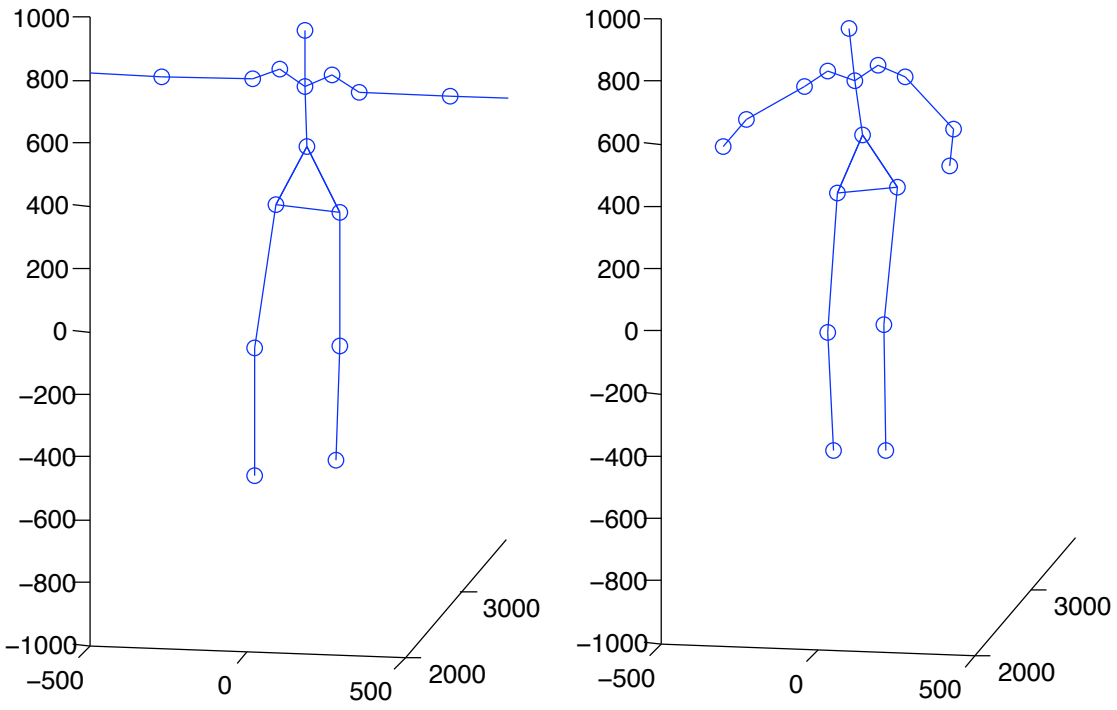

FIGURE 8.5: Illustration d'une correction du suivi d'OpenNI. À gauche, l'observation initiale, sur laquelle on peut constater une position non naturelle des bras. À droite, le signal filtré pour la même trame, où les bras ont repris une position réaliste (d'après [133]).

Conclusion de la deuxième partie

Durant mon travail, j'ai assimilé la séparation de sources à la recherche de leur distribution *a posteriori* étant donnés les mélanges.

Une contribution théorique de mon travail a été de montrer que les processus gaussiens peuvent être utilisés dans toute leur généralité comme un *a priori* intéressant pour décrire les signaux sources en vue de leur séparation. Ainsi, j'ai assimilé dans toute cette partie le problème de la séparation de sources à celui de l'identification de leur *distribution a posteriori* étant donnés les mélanges. En effet, une telle distribution résume notre état de connaissance sur les sources après l'observation des mélanges et si une estimée est nécessaire, des critères de décision classiques peuvent être appliqués sur la distribution *a posteriori*.

Dans un premier temps, j'ai supposé les fonctions de moyenne et de covariance des processus sources connues.

Le premier cas de figure qui m'a intéressé dans le chapitre 5 est celui où on a accès à un seul signal de *mélange* ($I = 1$), qui est simplement la somme des J sources. Dans ces conditions, j'ai considéré le problème assez général où l'objectif est de déterminer la valeur des sources en certains points de l'espace étant données les valeurs de leur mélange en d'autres. Cette formulation généralise ce qu'on appelle habituellement un problème de *régression*, où le mélange est constitué d'un signal utile auquel s'ajoute un bruit. Dans ce contexte, j'ai montré que les processus gaussiens offrent une solution naturelle au problème de la séparation, puisqu'il est facile de calculer la distribution *a posteriori* des sources.

Après ce premier contact avec la séparation de processus gaussiens, j'ai envisagé le cas un peu plus complexe où plusieurs mélanges des sources sont disponibles ($I > 1$). Pour commencer, j'ai supposé que chaque mélange est une combinaison linéaire différente des sources, et que la procédure de mixage est donc donnée par une *matrice de mélange*, que j'ai supposée connue pour commencer. Dans ces conditions, j'ai montré que la procédure de séparation vue pour le cas monocanal se généralise très bien à ce cas multicanal. Une fois encore, j'ai envisagé le problème dans toute sa généralité comme la recherche de la valeur des sources en certains points alors que les mélanges sont observés en d'autres.

Un cas particulier important de ces procédures de séparation est constitué par celui de la séparation de PGLS. J'ai donc repris les équations obtenues dans les sections précédentes pour montrer que si on cherche à séparer les sources sur les mêmes points que ceux observés et qu'elles sont des PGLS régulièrement échantillonnés, alors les calculs se simplifient grandement. J'ai ainsi abouti à l'expression de la distribution *a posteriori* des TFCT des sources, faisant ainsi le pont avec les expressions classiques de la littérature.

S'il est pratique pour les calculs de considérer que les mélanges sont des combinaisons linéaires des sources, une telle hypothèse peut s'avérer assez limitative en pratique. J'ai ainsi argumenté en faveur de modèles de mixage permettant de mieux modéliser le lien entre sources et mélanges. La stratégie dans ce but consiste à modéliser les I mélanges comme la somme de J signaux images. Chaque image est composée de I signaux, qui correspondent à la contribution de la source correspondante dans chacun des mélanges. Le modèle instantané est un cas simple où chaque image est obtenue en pondérant le signal source par différents gains constants dans chaque mélange. En toute généralité, un processus de mixage est un modèle qui établit le lien entre une source et son image.

En prenant l'exemple de l'audio où il est fréquent que les mélanges soient produits par un

filtrage des sources, j'ai motivé l'introduction du modèle de mélange *convolutif* dans le chapitre 6. Dans ce modèle, l'image de chaque source est obtenue comme la convolution d'une réalisation de cette source par I filtres différents, un par canal de mélange. Cette idée se généralise sans problème au cas des signaux définis sur des espaces Euclidiens de dimension quelconque.

C'est ainsi que j'ai abordé le problème de la séparation de sources dans le cas assez général où les différents mélanges observés sont produits par mixage convolutif des sources et où on souhaite estimer ces sources en des points différents de ceux où les mélanges sont observés. J'ai montré que les expressions correspondantes se trouvent sans difficulté particulière, bien qu'elles soient assez complexes à mettre en œuvre sur le plan calculatoire.

Compte tenu de son importance dans les applications, j'ai ensuite considéré le cas particulier des mélanges convolutifs de PGLS. Si le support des filtres de mélange est suffisamment petit devant la taille des trames, que la fenêtre de pondération utilisée est approximativement constante sur le support des filtres et que la transformée fréquentielle utilisée est celle de Fourier, j'ai montré que la séparation d'un mélange convolutif peut être effectuée de la même manière que dans le cas d'un mélange instantané, à la différence que la matrice de mélange dépend de la fréquence considérée. Ce résultat est connu dans la littérature, mais j'ai montré qu'on n'a rien à gagner à se restreindre au cas des séries temporelles, puisqu'il est valable en dimension quelconque.

Une fois le modèle de mixage convolutif établi, je me suis penché sur le modèle *diffus*, introduit récemment dans la communauté de la séparation de sources audio. En général, ce modèle est présenté pour les seuls PGLS. Plutôt que de supposer un lien déterministe entre source et image comme le font les modèles instantanés et convolutifs, le modèle diffus caractérise simplement l'image d'une source par une matrice de *covariance spatiale* dépendante de la fréquence, qui établit un lien probabiliste entre les valeurs prises par ses différents canaux. Le mixage ponctuel en est un cas particulier, pour lequel cette covariance spatiale dégénère en une relation déterministe.

J'ai présenté trois modèles de mixages classiques : l'instantané, le convolutif et le diffus. Pour chacun, j'ai donné les expressions explicites des distributions *a posteriori* des sources étant donnés les mélanges, pour le cas général et pour le cas PGLS.

Plutôt que de me contenter d'une telle définition, j'ai montré que l'image d'une source mixée de manière diffuse peut se comprendre comme la superposition d'une multitude de réalisations indépendantes de cette source, chacune mixée de manière ponctuelle. Il apparait donc que ce formalisme permet de modéliser des sources diffuses, c'est-à-dire des sources qui ne peuvent pas se réduire à un seul point vibrant. Cette discussion permet d'étendre le modèle diffus hors du seul contexte des PGLS.

Une fois ce modèle présenté, j'ai abordé le problème de la séparation de sources mixées de manière diffuse. J'ai ainsi montré que dans le cas des PGLS, il est facile de déduire la distribution *a posteriori* des images des sources pour peu que les paramètres du modèle soient connus. Ce faisant, j'ai généralisé les résultats obtenus auparavant pour le seul cas des séries temporelles. Cela dit, cette extension s'est faite très naturellement.

Pendant toute cette discussion sur la séparation de processus gaussiens, j'ai supposé connus les paramètres du modèle, dont les fonctions de moyenne et de covariance des sources ainsi que les filtres de mélange ou les covariances spatiales, selon le modèle considéré. Je suis alors revenu sur cette hypothèse très forte au chapitre 7, pour montrer qu'il est en fait possible d'estimer ces paramètres à partir de la seule observation des mélanges. Cela est rendu possible dès lors que certaines hypothèses sont faites sur les familles paramétriques auxquelles appartiennent les fonctions de covariance des sources ou bien sur le type de mélange considéré. Dans ce cas *semi-informé*, le problème peut s'exprimer comme la recherche des hyperparamètres qui maximisent la vraisemblance des mélanges observés. J'ai évoqué les différentes approches suggérées dans la littérature pour mener à bien ce travail d'estimation.

On peut appliquer le formalisme gaussien pour séparer des sources dont on n'observe que les mélanges, moyennant certaines hypothèses. J'ai appelé *semi-informée* cette configuration et j'en ai présenté deux exemples.

De manière à mettre en pratique ce formalisme général de la séparation de processus gaussiens, j'ai considéré dans le chapitre 8 deux exemples où l'objectif est d'effectuer une séparation à partir de la seule observation des mélanges. C'est ainsi que j'ai démontré l'efficacité de l'approche pour

la séparation de rythmiques à partir d'enregistrements musicaux, ainsi que pour la décomposition de mouvements de danse en composantes élémentaires. Dans ces deux cas très différents, c'est la même logique qui sous-tend les calculs et il est intéressant de constater que le même formalisme permet d'aborder des problèmes en apparence très différents.

Troisième partie

Séparation informée paramétrique

Introduction

Cette partie, ainsi que la suivante, est consacrée au détail de mes travaux dans le domaine de la séparation informée de signaux audio, que j'ai déjà introduite brièvement en section 1.3 page 7. La principale différence de la séparation informée par rapport à la séparation aveugle ou semi-informée réside dans le fait que les paramètres de séparation peuvent y être appris à partir des sources à séparer elles-mêmes, plutôt qu'à partir du seul mélange. Ce cas de figure se comprend volontiers comme un problème de codage, résumé en figure 1.1 page 8. Dans une première phase d'*encodage*, on dispose à la fois des sources et du mélange et on peut générer une *information annexe*. Dans une deuxième phase de *décodage* où les sources ne sont plus disponibles, on peut utiliser à la fois le mélange et l'information annexe pour les récupérer.

Ma thèse s'est déroulée dans le cadre du projet DReaM (ANR-09-CORD-006-03), portant sur l'écoute active de la musique. La séparation informée en est une composante forte.

Lorsque j'ai débuté ma thèse, la seule technique de séparation informée disponible était *l'inversion locale* et ses variantes, présentée en section 1.3.3 et proposée par Parvaix et Girin [169, 170, 167]. Si les liens entre séparation informée et codage multicanal étaient pressentis [170], ils n'étaient pas encore clairement identifiés. Dans ce contexte, un des objectifs de ma thèse était d'étudier la question et de proposer des techniques innovantes pour la séparation informée dans le cadre du projet DReaM [6]. Ce projet, financé par l'Agence Nationale de la Recherche, a regroupé plusieurs équipes scientifiques provenant de laboratoires différents, dont le GIPSA-lab [7] à Grenoble, l'Institut Langevin à Paris, le LaBRI [8] à Bordeaux et enfin Télécom ParisTech. C'est une riche collaboration scientifique qui nous a permis au fil des ans de comprendre la séparation de sources informée comme une instance particulière d'un problème de codage audio multicanal.

Plusieurs techniques de séparation informée ont émergé au cours de mon doctorat [87, 200] au sein même des équipes du projet DReaM. Je me suis pour ma part assez tôt concentré sur un modèle gaussien, que je vais présenter ici. Il est entendu que cette présentation bénéficie de trois années de maturation et d'échanges avec les autres membres du consortium, que je tiens encore à vivement remercier pour leur aide et le partage de leurs idées. Considérer comme je vais le faire ici que la séparation informée est une technique de codage multicanal paramétrique particulière (*parametric audio coding*) est une idée originale d'Alexey Ozerov, qui a eu la gentillesse de constamment me faire part de ses lumières sur la problématique du codage audio. Par ailleurs, j'ai eu la chance de voir mon travail sur la séparation informée paramétrique déjà repris et étendu par Gorlow *et al.* dans [88] pour le cas des sources ponctuelles.

Cette partie est structurée de la manière suivante. Tout d'abord, je vais montrer au chapitre 9 comment le formalisme gaussien que j'ai présenté dans les parties I et II peut être mis à profit très naturellement pour la séparation informée paramétrique. J'y présenterai une formalisation originale du problème de la séparation informée, qui étend considérablement celles qu'on peut trouver dans la littérature. Le problème de l'estimation et du codage des paramètres sera abordé en deux temps. Au chapitre 10, je me concentrerai sur le cas simple où le mixage diffus présenté en section 6.2 n'intervient pas. Le cas général sera abordé au chapitre 11. Enfin, je proposerai une évaluation du système proposé dans le chapitre 12.

6. Le Disque Repensé pour l'Écoute Active de la Musique, `http://dream.labri.fr`
7. Le GIPSA-lab esr le laboratoire de recherche Grenoble, Images, Parole, Signal et Automatique
8. Le LaBRI est le Laboratoire Bordelais de Recherche en Informatique

Chapitre 9

Processus gaussiens et séparation informée

Dans toute la suite de cet exposé, les signaux considérés sont des enregistrements audio régulièrement échantillonnés et sont donc des fonctions définies sur $\mathbb{T} = \mathbb{Z}$ et à valeurs dans $\mathbb{R}$. On considère comme en partie II que les I mélanges $\tilde{x}(\cdot, i)$ sont la somme des J images des sources $\tilde{y}(\cdot, \cdot, j)$:

$$\forall (t, i)\,,\ \tilde{x}(t, i) = \sum_{j=1}^{J} \tilde{y}(t, i, j)\,,$$

où le lien entre une image $\tilde{y}(\cdot, \cdot, j)$ et le processus source correspondant $\tilde{s}(\cdot, j)$ est donné par le *modèle de mixage* choisi, comme on l'a vu aux chapitres 5 et 6. Si l'image est produite par filtrage linéaire d'une seule réalisation de la source, on parle de mélange *convolutif*. Si elle est produite par filtrage linéaire de plusieurs réalisations indépendantes de cette source, je parlerai plus volontiers d'un mixage *diffus*. Dans toute la suite de mon exposé, je considèrerai que les sources sont des PGLS indépendants.

Dans le cas particulier de la séparation informée, on dispose dans une première phase d'*encodage* de la connaissance des réalisations non seulement des mélanges $\tilde{x}$, mais également des sources. Lors de cette phase, on peut produire une *information annexe* utilisée lors d'une deuxième phase de *décodage*, où l'objectif est de récupérer certains des signaux mixés.

9.1 Formalisation

Un des premiers problèmes qui va m'occuper dans cette partie est la formalisation des différentes configurations pouvant survenir dans un problème de séparation informée. La formalisation que je propose dans cette section est une généralisation de celles qu'on peut trouver dans les différentes études relatives à la séparation informée [169, 170, 167, 87, 130, 137], ou même de celles relatives au codage audio multicanal [104, 61], qui n'en considèrent toutes que des cas particuliers.

Je propose d'identifier trois dimensions principales pour caractériser un problème particulier de séparation informée. Tout d'abord, celle de la nature des sources observées, qui sont ponctuelles ou diffuses. Une deuxième caractérisation du problème concerne la nature du mixage de ces sources dans les mélanges. Enfin, on peut être intéressé par la récupération des signaux sous la forme de sources ponctuelles ou d'images. Je vais préciser maintenant ces différents éléments, qu'on trouvera résumés sur la figure 9.1 page suivante.

9.1.1 Signaux observés : sources ponctuelles et sources diffuses

Tout d'abord, parmi les J processus sources, il y en a $S_p \leq J$ dont on ne dispose que d'une seule réalisation. Je parlerai alors de sources observées *ponctuelles* $\tilde{\boldsymbol{s}}_p(\cdot, j)$. Les $S_d = J - S_p$ autres sont

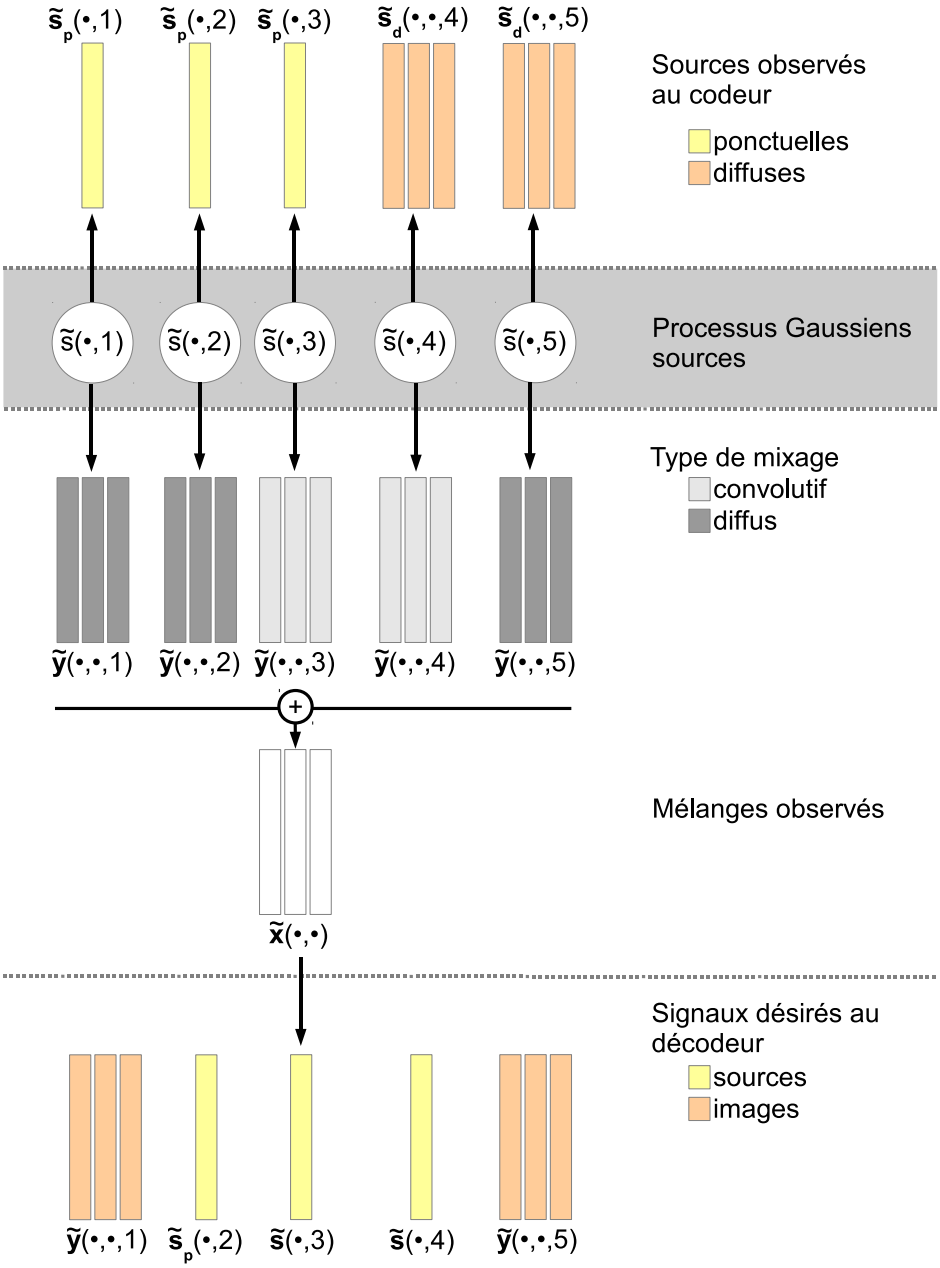

FIGURE 9.1: Exemple de configuration pour la séparation informée. Les processus sources $\tilde{s}(\cdot,\cdot)$ sont des objets mathématiques. Au codeur, les sources $\mathcal{S}_p = \{1, 2, 3\}$ sont observées sous forme ponctuelle, tandis que les sources $\mathcal{S}_d = \{4, 5\}$ sont observées sous forme diffuse. Lors de la production du mélange et indépendamment de la manière dont elles sont observées au codeur, les sources $\mathcal{M}_c = \{3, 4\}$ sont mixées de manière convolutive, tandis que les sources $\mathcal{M}_d = \{1, 2, 5\}$ sont mixées de manière diffuse. Enfin, l'objectif du traitement est de récupérer les sources $\mathcal{Z}_s = \{2, 3, 4\}$ sous forme ponctuelle et les sources $\mathcal{Z}_y = \{1, 5\}$ sous forme d'images. De plus, aucune source n'est à ignorer pour la séparation ($\mathcal{Z}_\varnothing = \varnothing$). On peut noter que les sources à récupérer sous forme ponctuelle sont bien soit mixées de manière convolutive, soit observées sous forme ponctuelle au codeur : $\mathcal{Z}_s \subset (\mathcal{S}_p \cup \mathcal{M}_c)$. Dans le cas de la source $2 \in \mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d$, c'est la source ponctuelle observée au codeur qu'on cherche à estimer, puisque son mixage diffus fait qu'il n'y a pas d'unique réalisation menant à son image dans le mélange, mais une multitude, comme on l'a vu en section 6.2.2..

uniquement disponibles sous la forme d'un mixage diffus des processus sources correspondants. Je parlerai alors de sources observées *diffuses* $\tilde{\boldsymbol{s}}_d(\cdot,\cdot,j)$. Chacune de ces sources diffuses est un ensemble de I signaux.
Par analogie avec le mixage diffus présenté en section 6.2, une source diffuse $\tilde{s}_d(\cdot,\cdot,j)$ sera ainsi définie comme un mixage diffus d'un des processus source $\tilde{s}(\cdot,j)$. Ainsi, la covariance d'une source diffuse $\boldsymbol{s}_d(f,n,\cdot,j)$ pour un point (f,n) donné s'écrira :

$$\mathbb{E}\left[\boldsymbol{s}_d(f,n,\cdot,j)\,\boldsymbol{s}_d(f,n,\cdot,j)^H\right] = P(f,n,j)\,R_j^{\text{obs}}(f)\,, \tag{9.1.1}$$

où $P(f,n,j)$ est la DSP du processus source $\tilde{s}(\cdot,j)$ correspondant au point (f,n), tandis que $R_j^{\text{obs}}(f)$ désigne la matrice de covariance spatiale, de dimension $I\times I$, *de l'observation* de cette source à l'encodeur [1], à la fréquence f. Dans la suite, je désignerai par $\mathcal{S}_p$ et $\mathcal{S}_d$ l'ensemble des indices des sources observées au codeur de manière ponctuelle et diffuse, respectivement.

Une source est un *processus*, il s'agit d'un objet mathématique. On observe à l'encodeur des *réalisations* des sources. Pour une unique réalisation on parle de source ponctuelle tandis qu'on parle de source diffuse pour l'observation de plusieurs réalisations.

Avant d'aller plus loin, une courte discussion s'impose ici sur l'avantage de considérer des sources diffuses. Une telle source est caractérisée par deux propriétés importantes. La première est qu'elle présente les mêmes variations de DSP sur ses différents canaux et la deuxième est que ces canaux présentent une corrélation donnée par la matrice $R_j^{\text{obs}}(f)$, qui dépend de la fréquence et qui rend compte d'un processus de mixage complexe d'un unique processus source $\tilde{s}(\cdot,j)$. Si une de ces deux propriétés n'est pas respectée par un signal source multicanal observé, il est inutile d'adopter un modèle de source diffuse pour le modéliser :

- Si les différents canaux du signal en question ne présentent manifestement pas la même DSP, à une spatialisation près, alors il sera mal modélisé comme une source diffuse.
- Si les différents canaux du signal sont manifestement produits par des processus physiques distincts, même si leur DSP est la même, il n'est pas très avantageux de le modéliser comme une source diffuse. En effet, la principale force du modèle diffus est de tenir compte des dépendances *a priori* entre les canaux d'un signal. Cette force disparaît si ces canaux sont indépendants.

Certains signaux musicaux multicanaux, comme la section rythmique, sont souvent mal modélisés comme une seule source de covariance spatiale constante. On se ramène alors au cas de I sources ponctuelles indépendantes *a priori*. Certains modèles de sources comme NTF peuvent de toute façon rendre compte des redondances dans les DSP des différentes sources.

Une stratégie que j'ai souvent utilisée dans les applications consiste à tout simplement ignorer le fait que certains groupes de signaux sont relatifs à la même source et à les considérer comme I sources ponctuelles indépendantes, en augmentant simplement S_p. Cette stratégie a l'avantage de la simplicité puisqu'elle permet de simplifier l'apprentissage du modèle, mais elle a l'inconvénient de nécessiter un modèle indépendant pour la DSP des différents canaux de ces sources, menant à une information annexe plus conséquente. Cependant, si on utilise un modèle de sources NTF, ce coût supplémentaire est négligeable, puisqu'il suffit alors de rajouter $(I-1)\,K$ paramètres à la matrice de gains Q du modèle pour chaque source diffuse ainsi transformée en I sources ponctuelles. Si on utilise le modèle par compression d'images, le coût en débit est plus important.

1. Il faut bien souligner ici que cette matrice de covariance spatiale de l'observation est différente de celle $R_j(f)$ de l'image de la source dans le mélange, qu'on verra plus loin. Elles ne seront identiques que si la source diffuse observée est sommée telle quelle dans le mélange.

9.1.2 Mixages convolutifs ou diffus

Après une première caractérisation des problèmes de séparation informée portant sur la nature des signaux sources observés au codeur, une deuxième porte sur la nature du mixage, ou plutôt sur la nature du modèle rendant compte du mixage. *Indépendamment de la manière dont ils sont observés au codeur*, chacun des j processus sources $\tilde{s}$ est mixé de manière à produire son image $\tilde{y}(\cdot,\cdot,j)$ dans le mélange. Je considèrerai que ce mixage est soit convolutif, soit diffus. Je désignerai par M_c et M_d le nombre de sources mixées de manière convolutive et diffuse, respectivement. Les indices correspondants seront rassemblés dans les ensembles $\mathcal{M}_c$ et $\mathcal{M}_d$.

Comme on le voit, une source dont on dispose d'une observation ponctuelle au codeur peut être mixée de manière diffuse, et une source observée sous forme diffuse au codeur peut être mixée de manière ponctuelle, rendant possible son estimation ponctuelle au décodeur. On a alors :

$$\forall j \in \mathcal{M}_c,\ \mathbb{E}\left[\boldsymbol{y}(f,n,\cdot,j)\,\boldsymbol{y}(f,n,\cdot,j)^H\right] = P(f,n,j)\,\underbrace{A_j(f)\,A_j(f)^H}_{R_j(f)}, \tag{9.1.2}$$

$$\forall j \in \mathcal{M}_d,\ \mathbb{E}\left[\boldsymbol{y}(f,n,\cdot,j)\,\boldsymbol{y}(f,n,\cdot,j)^H\right] = P(f,n,j)\,R_j(f), \tag{9.1.3}$$

où les notations sont les mêmes que celles de la section 6.2 page 89. Quatre situations possibles émergent, en fonction du type d'observation et du type de mixage, qui sont résumées en table 9.1.

	mixage convolutif	mixage diffus
source ponctuelle	La seule réalisation $\boldsymbol{s}_p(f,n,j)$ observée au codeur est aussi notée $\boldsymbol{s}(f,n,j)$. Elle est mixée de manière convolutive, donnant lieu à l'image $\boldsymbol{y}(f,n,\cdot,j)$, dont la covariance est donnée par 9.1.2. *(On peut estimer $\boldsymbol{s}$ ou $\boldsymbol{y}$ au décodeur.)*	Au codeur, on observe une réalisation $\boldsymbol{s}_p(f,n,j)$ de la source. L'image de cette source dans le mélange est notée $\boldsymbol{y}(f,n,\cdot,j)$, et est caractérisée par la covariance 9.1.3, où $R_j(f)$ donne la covariance spatiale dans le mélange. *(On peut estimer $\boldsymbol{s}_p$ ou $\boldsymbol{y}$ au décodeur.)*
source diffuse	Au codeur, on observe $\boldsymbol{s}_d(f,n,\cdot,j)$, formée de I canaux et dont la covariance est donnée par 9.1.1, avec une matrice de covariance spatiale $R_j^{\text{obs}}(f)$. On suppose qu'une autre réalisation $\boldsymbol{s}(f,n,j)$ existe de la source, non observée, qui est mixée pour donner lieu à l'image $\boldsymbol{y}(f,n,\cdot,j)$, dont la covariance est donnée par 9.1.2. *(On peut estimer $\boldsymbol{s}$ ou $\boldsymbol{y}$ au décodeur.)*	On observe $\boldsymbol{s}_d(f,n,\cdot,j)$, formée de I canaux et dont la covariance est donnée par 9.1.1, avec une matrice de covariance spatiale $R_j^{\text{obs}}(f)$. L'image de cette source dans le mélange est notée $\boldsymbol{y}(f,n,\cdot,j)$, et caractérisée par la covariance 9.1.3, où $R_j(f)$ donne la covariance spatiale dans le mélange qui est *distincte* de $R_j^{\text{obs}}(f)$ dans le cas général. *(On peut estimer $\boldsymbol{y}$ au décodeur.)*

TABLE 9.1: Différentes configurations de séparation informée en fonction du type des sources observées et du type de mixage.

Il est important de noter que dans le formalisme que je propose, les réalisations des sources qui conduisent aux mélanges observés après mixage ne sont pas nécessairement les mêmes que celles qui conduisent aux signaux observés à l'encodeur. Je ne supposerai que ce sera le cas que pour les sources observées de manière ponctuelle et mixées de manière convolutive, auquel cas on aura :

$$\forall j \in \mathcal{S}_p \cap \mathcal{M}_c,\ \boldsymbol{s}_p(f,n,j) = \boldsymbol{s}(f,n,j).$$

Pour les autres, les sources observées ne correspondent pas aux mêmes réalisations que les sources mixées. Cela permet de rendre compte du fait qu'on puisse vouloir récupérer sous forme ponctuelle une source observée au codeur sous forme diffuse, pour peu qu'on suppose qu'elle soit mixée dans les mélanges de manière ponctuelle. Bien entendu, il faut voir là un artifice mathématique par lequel on peut rendre compte de beaucoup de cas de figure complexes. En général, ce sont bien les signaux observés au codeur qui font l'objet d'un mixage et qui donnent lieu aux mélanges.

Quoiqu'il en soit, pour toutes les sources mixées de manière convolutive ($j \in \mathcal{M}_c$), je noterai $\boldsymbol{s}(f,n,j)$ leur réalisation qui donne lieu après mixage à l'image $\boldsymbol{y}(f,n,\cdot,j)$ dans le mélange, dont la matrice de covariance est donnée par 9.1.2. Pour celles qui sont en plus observées de manière ponctuelle au codeur, je supposerai que cette réalisation est celle observée au codeur : on aura $\boldsymbol{s}_p(f,n,j) = \boldsymbol{s}(f,n,j)$.
Par ailleurs et comme on le verra aux chapitres 10 et 11, je propose d'aborder non seulement le cas où les paramètres de mixage sont connus mais aussi celui où il faut les estimer.

9.1.3 Signaux à récupérer

Enfin, une troisième caractérisation des problèmes de séparation informée porte sur les signaux qu'on souhaite récupérer au décodeur. Certains des signaux mixés sont à estimer sous la forme de sources ponctuelles et d'autres sous la forme d'images. Je considèrerai en outre qu'on puisse ne pas être intéressé par la séparation de tous les signaux. Soit $\mathcal{Z}_s$ l'ensemble des Z_s indices des sources ponctuelles à estimer et $\mathcal{Z}_y$ celui des Z_y signaux à récupérer sous forme d'image, avec $\mathcal{Z}_s \cap \mathcal{Z}_y = \varnothing$. Les autres $Z_\varnothing = J - Z_s - Z_y$ sources, dont les indices sont rassemblés dans $\mathcal{Z}_\varnothing = \mathbb{N}_J \setminus (\mathcal{Z}_s \cup \mathcal{Z}_y)$ ne sont pas à séparer.

Dans l'approche que je propose, il y a certaines contraintes qui font qu'on ne peut pas récupérer sous forme ponctuelle n'importe quelle source. En effet, si on n'a pas observé un processus source sous forme ponctuelle au codeur et si son mixage dans le mélange est diffus, alors je considèrerai que cette source ne peut être récupérée que sous la forme d'image. Je n'ai d'ailleurs pas connaissance d'une quelconque étude qui permettrait une telle procédure. Il est cependant possible d'aborder ce problème en séparant d'abord l'image de cette source, puis en tâchant de manière aveugle d'en reconstruire une version ponctuelle. Ce problème non trivial [2] pourrait être abordé sous l'angle de la synthèse de processus gaussien, présentée en annexe A. Dans les autres cas, on peut récupérer les sources soit sous une forme ponctuelle, soit sous la forme d'image.

Il est intéressant de constater que le formalisme proposé permet de récupérer des sources ponctuelles dans deux situations jusqu'alors inédites. Tout d'abord, même si elles sont observées de manière diffuse au codeur, on peut en estimer une réalisation ponctuelle si leur mixage est ponctuel. Dans ce cas, c'est la réalisation unique $\boldsymbol{s}(\cdot,\cdot,j)$ menant à l'image qu'on cherche à estimer. Ensuite, si elles sont observées sous forme ponctuelle $\boldsymbol{s}_p(\cdot,\cdot,j)$ à l'encodeur, alors on peut toujours chercher à estimer $\boldsymbol{s}_p(\cdot,\cdot,j)$ à partir du mélange, même si c'est par un modèle diffus qu'on modélise la production de l'image $\boldsymbol{y}(\cdot,\cdot,\cdot,j)$ dans le mélange.

L'ensemble des cas de figure où il est possible d'estimer une réalisation ponctuelle des sources peut se résumer en disant qu'il faut avoir :

$$\mathcal{Z}_s \subset (\mathcal{S}_p \cup \mathcal{M}_c)\,. \tag{9.1.4}$$

Formellement, l'objectif de la procédure est alors de permettre au décodeur d'estimer pour chaque point (f,n) le vecteur $\boldsymbol{z}(f,n,\cdot)$, de dimension $(Z_s + IZ_y) \times 1$, défini par :

$$\boldsymbol{z}(f,n,\cdot) = \begin{bmatrix} \boldsymbol{s}(f,n,\mathcal{Z}_s \cap \mathcal{M}_c) \\ \boldsymbol{s}_p(f,n,\mathcal{Z}_s \cap \mathcal{M}_d \cap \mathcal{S}_p) \\ \boldsymbol{y}(f,n,\cdot,\mathcal{Z}_y) \end{bmatrix}, \tag{9.1.5}$$

où $\boldsymbol{s}(f,n,\mathcal{Z}_s \cap \mathcal{M}_c)$ est un vecteur formé des éléments de $\boldsymbol{s}(f,n,\cdot)$ dont l'indice est dans $\mathcal{Z}_s \cap \mathcal{M}_c$. De la même manière, $\boldsymbol{s}_p(f,n,\mathcal{Z}_s \cap \mathcal{M}_d \cap \mathcal{S}_p)$ désigne le vecteur des sources ponctuelles observées au

2. On peut toujours, de manière brutale, effectuer la moyenne des canaux d'une image pour récupérer un signal monophonique. Cette procédure simple est exposée au risque d'interférences destructives.

codeur dont l'indice est dans $\mathcal{Z}_s \cap \mathcal{M}_d \cap \mathcal{S}_p$. Si un de ces ensembles est vide, le vecteur correspondant est vide également. $\boldsymbol{y}(f,n,\cdot,\mathcal{Z}_y)$ se définit pareillement comme le vecteur de dimension $IZ_y \times 1$, composé de la concaténation des $\{\boldsymbol{y}(f,n,\cdot,j)\}_{j\in\mathcal{Z}_y}$:

$$\boldsymbol{y}(f,n,\cdot,\mathcal{Z}_y) = \begin{bmatrix} \boldsymbol{y}\left(f,n,\cdot,\mathcal{Z}_y^{(1)}\right) \\ \vdots \\ \boldsymbol{y}\left(f,n,\cdot,\mathcal{Z}_y^{(Z_y)}\right) \end{bmatrix}, \tag{9.1.6}$$

où $\mathcal{Z}_y^{(j)}$ est le $j^{\text{ème}}$ élément de $\mathcal{Z}_y$. Une fois les TFCT des signaux recherchés estimés, les signaux temporels correspondants peuvent être récupérés par transformée de Fourier inverse et addition-recouvrement.

9.1.4 Approche gaussienne pour la séparation informée

L'approche adoptée dans cette partie consiste à modéliser chaque source comme un processus gaussien localement stationnaire et à assimiler l'information annexe Θ aux paramètres requis par la séparation de processus gaussiens, telle que présentée en partie II.

– **L'encodeur** utilise la connaissance des mélanges et des sources pour produire l'information annexe :

$$\Theta = \left\{ \{P(f,n,j)\}_{f,n,j} \, , \, \{R_j(f)\}_{f,j} \, , \, \{U_j(f)\}_{j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d} \right\}, \tag{9.1.7}$$

où P désigne l'ensemble des DSP des sources, $\{R_j(f)\}_{f,j}$ sont les matrices de covariance spatiale des images, données soit sous la forme générale 9.1.3 si leur mixage est diffus, soit par le biais des filtres de mélange s'il est convolutif. Dans ce dernier cas, on peut également transmettre les réponses impulsionnelles des filtres.
Les filtres $\{U_j(f)\}_{j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d}$ dits de *formation de voi*e, que je n'ai pas encore abordés, permettent de récupérer sous forme ponctuelle des sources observées au codeur sous forme ponctuelle, mais mixées de manière diffuse. Je reviendrai sur ce point en section 9.2.
La manière de construire Θ au codeur à partir des observations est précisée aux chapitres 10 et 11

– **Le décodeur paramétrique**, muni de l'information annexe Θ et des mélanges, peut déterminer les meilleures estimées $\widehat{\boldsymbol{z}}(f,n,\cdot)$ des signaux recherchés selon le critère de l'erreur quadratique moyenne :

$$\widehat{\boldsymbol{z}}(f,n,\cdot) = K(\boldsymbol{z}(f,n,\cdot),\boldsymbol{x}(f,n,\cdot))\, K(\boldsymbol{x}(f,n,\cdot),\boldsymbol{x}(f,n,\cdot))^{-1}\, \boldsymbol{x}(f,n,\cdot), \tag{9.1.8}$$

où l'expression des différentes matrices de covariance dépend des paramètres Θ du modèle. Je reviendrai sur ce point en section 9.2, mais on peut d'ores et déjà remarquer que cette procédure de séparation est d'une faible complexité : $\mathcal{O}\left(NFI^3\right)$ et peut donc aisément être implémentée pour fonctionner en temps réel sur des appareils disposant d'une puissance de calcul limitée.

Comme on le voit, le système que je propose pour la séparation informée revient simplement à transmettre à l'algorithme de séparation les grandeurs dont il a besoin pour récupérer de manière optimale les signaux recherchés selon le critère de l'erreur quadratique moyenne. En termes simples, ces grandeurs vont s'avérer être les spectrogrammes des sources ainsi que les paramètres de mélange. Au décodeur, un filtrage de Wiener généralisé permet d'effectuer la séparation.
Cette formulation soulève plusieurs problématiques importantes. Tout d'abord, il faut définir la procédure de séparation à mettre en œuvre au décodeur pour récupérer les signaux recherchés à partir des mélanges et de l'information annexe. Comme on peut le voir dans l'expression 9.1.8, cela se fera par le biais d'une séparation similaire à celles présentées en partie II. Cette procédure fait cependant intervenir quelques subtilités, liées au nombre conséquent des cas de figures envisagés. J'évoquerai ce point en section 9.2.

Ensuite, l'information annexe 9.1.7 contient bien trop de paramètres pour être d'une réelle utilité pratique. En effet, un encodage naïf de Θ impliquerait des débits très importants pour sa transmission du codeur au décodeur, compte tenu des JFN coefficients des DSP $P(f,n,j)$ des sources. En somme, il serait plus avantageux de directement inclure les signaux séparés dans Θ, ce qui serait un constat d'échec pour la séparation informée. Fort heureusement, il existe des stratégies que je vais bientôt présenter en section 9.3 pour rendre l'approche très efficace en termes de coûts de transmission.
Par conséquent, le choix 9.1.7 pour l'information annexe n'a pas d'intérêt pratique. En revanche, il a un intérêt théorique important, puisqu'il est celui qui mène aux meilleures performances qu'on puisse atteindre en utilisant ce modèle 9.1.8 de séparation. Dans la littérature, ce genre de configuration idéale pour un modèle particulier est traditionnellement appelée *une configuration oracle* [213]. Ainsi, le lot 9.1.7 de paramètres pour l'information annexe portera le nom de configuration oracle dans cette étude.

9.2 Séparation au décodeur

Si on suppose connue l'information annexe 9.1.7, la séparation au décodeur peut se faire naturellement en utilisant le formalisme gaussien qui m'a occupé dans la partie II. Pour obtenir la distribution *a posteriori* de $\boldsymbol{z}(f,n,\cdot)$ étant donné le mélange $\boldsymbol{x}(f,n,\cdot)$, il est nécessaire de définir les matrices de covariance suivantes [3] :

– Covariance de $\boldsymbol{z}$:

$$\begin{aligned} K(\boldsymbol{z}(f,n,\cdot),\boldsymbol{z}(f,n,\cdot)) &= \mathbb{E}\left[\boldsymbol{z}(f,n,\cdot)\,\boldsymbol{z}(f,n,\cdot)^H\right] \\ &= \begin{bmatrix} \operatorname{diag}(P(f,n,\mathcal{Z}_s)) & 0 \\ 0 & \operatorname{blockdiag}\left(\{P(f,n,j)\,R_j(f)\}_{j\in\mathcal{Z}_y}\right) \end{bmatrix}, \end{aligned}$$

où $\operatorname{blockdiag}\left(\{M_q\}_{q=1,\dots,Q}\right)$ est la matrice bloc-diagonale dont les blocs diagonaux sont les matrices M_q :

$$\operatorname{blockdiag}\left(\{M_q\}_{q=1,\dots,Q}\right) = \begin{bmatrix} M_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & M_Q \end{bmatrix}. \tag{9.2.1}$$

– Covariance du mélange :

$$\begin{aligned} K(\boldsymbol{x}(f,n,\cdot),\boldsymbol{x}(f,n,\cdot)) &= \mathbb{E}\left[\boldsymbol{x}(f,n,\cdot)\,\boldsymbol{x}(f,n,\cdot)^H\right] \\ &= \sum_{j=1}^{J} P(f,n,j)\,R_j(f), \end{aligned} \tag{9.2.2}$$

– Enfin, la covariance entre $\boldsymbol{z}$ et $\boldsymbol{x}$, de dimension $(Z_s+IZ_y)\times I$, est :

$$K(\boldsymbol{z}(f,n,\cdot),\boldsymbol{x}(f,n,\cdot)) = \mathbb{E}\left[\boldsymbol{z}(f,n,\cdot)\,\boldsymbol{x}(f,n,\cdot)^H\right]. \tag{9.2.3}$$

Cette matrice est formée par la concaténation verticale de plusieurs groupes de matrices de covariance de différents signaux avec les mélanges.

1. Tout d'abord, les matrices de covariances $\{K(\boldsymbol{s}(f,n,j),\boldsymbol{x}(f,n,\cdot))\}_{j\in\mathcal{Z}_s\cap\mathcal{M}_c}$ des sources ponctuelles à estimer qui sont mixées de manière convolutive et des mélanges, chacune de dimension $1\times I$:

$$\forall j\in\mathcal{Z}_s\cap\mathcal{M}_c,\ K(\boldsymbol{s}(f,n,j),\boldsymbol{x}(f,n,\cdot)) = P(f,n,j)\,A_j(f)^H.$$

3. Lorsque $j\in\mathcal{M}_c$, $R_j(f)$ est construit à partir de $A_j(f)$ par $R_j(f) = A_j(f)\,A_j(f)^H$.

2. Ensuite, les matrices de covariances $\{K(\boldsymbol{s}_p(f,n,j),\boldsymbol{x}(f,n,\cdot))\}_{j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d}$ des sources ponctuelles à estimer qui sont observées sous forme ponctuelle à l'encodeur mais mixées de manière diffuse, chacune de dimension $1\times I$:

$$\forall j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d,\ K(\boldsymbol{s}_p(f,n,j),\boldsymbol{x}(f,n,\cdot))=\mathbb{E}\left[\boldsymbol{s}_p(f,n,j)\,\boldsymbol{x}(f,n,\cdot)^H\right].$$

Cette covariance n'est pas donnée comme dans le cas précédent par $P(f,n,j)\,A_j(f)^H$, puisqu'il n'existe pas de tel filtre de mélange $A_j(f)$ pour une source mixée de manière diffuse. Je propose de modéliser cette covariance par :

$$\mathbb{E}\left[\boldsymbol{s}_p(f,n,j)\,\boldsymbol{x}(f,n,\cdot)^H\right]=U_j(f)^\top P(f,n,j)\,R_j(f),\tag{9.2.4}$$

où $U_j(f)$, vecteur de dimension $I\times 1$, sera appelé un *filtre de formation de voie*, inclus dans l'information annexe.

3. Enfin, les Z_y matrices de covariance $\{K(\boldsymbol{y}(\boldsymbol{f},\boldsymbol{n},\cdot,j),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))\}_{j\in\mathcal{Z}_y}$ des images à estimer et des mélanges.

$$\forall j\in\mathcal{Z}_y,\ K(\boldsymbol{y}(f,n,\cdot,j),\boldsymbol{x}(f,n,\cdot))=P(f,n,j)\,R_j(f).$$

Muni de l'expression de ces trois matrices de covariance, la distribution *a posteriori* de $\boldsymbol{z}$ étant donné $\boldsymbol{x}$ est donnée par :

$$\boldsymbol{z}(\boldsymbol{f},\boldsymbol{n},\cdot)\mid\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot)\sim\mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}},K_{\text{post}}\right),\tag{9.2.5}$$

avec

$$\boldsymbol{\mu}_{\text{post}}=K(\boldsymbol{z}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))\,K(\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))^{-1}\,\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot)\tag{9.2.6}$$

et

$$\begin{aligned}K_{\text{post}}&=K(\boldsymbol{z}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{z}(\boldsymbol{f},\boldsymbol{n},\cdot))\\&\quad-K(\boldsymbol{z}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))\,K(\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))^{-1}\,K(\boldsymbol{z}(\boldsymbol{f},\boldsymbol{n},\cdot),\boldsymbol{x}(\boldsymbol{f},\boldsymbol{n},\cdot))^H.\end{aligned}\tag{9.2.7}$$

La tâche du décodeur est donc de construire les différentes matrices de covariance requises pour la séparation, et d'inverser pour chaque point (f,n) une matrice de dimension $I\times I$, en général petite (un cas classique est celui des mélanges stéréophoniques pour lesquels $I=2$). La complexité totale de cette phase de séparation au décodeur est donc de $\mathcal{O}\left(FNI^3\right)$. Pour référence ultérieure, j'ai consigné les opérations correspondantes dans l'algorithme 9.1.

9.3 Modèles de sources et de mixage

Sans considérer pour l'instant le problème de l'estimation de leurs valeurs que j'aborderai au chapitre 10, considérons pour commencer le nombre des différents constituants de l'information annexe 9.1.7 afin de déterminer ceux dont l'encodage entraîne un coût important en termes de débit.

Il apparaît tout d'abord que le nombre $FI(M_c+IM_d)$ des paramètres de mixage $\{R_j(f)\}_{j,f}$ ne dépend pas de la longueur des signaux. On peut donc considérer que leur transmission du codeur vers le décodeur ne pose pas vraiment de problème en comparaison du débit nécessaire à l'encodage d'un morceau de quelques minutes. La charge en débit correspondante est en effet la plupart du temps faible. Cependant, si on dispose des réponses impulsionnelles $a_{ij}(\tau)$ des filtres de mélange, de longueur $H<F$, il est avantageux de les transmettre de préférence aux réponses fréquentielles puisque dans ce cas le nombre des paramètres de mixage devient $HIM_c+FI^2M_d$ et qu'il est moins coûteux d'encoder des valeurs réelles que des valeurs complexes. Par ailleurs, les mêmes remarques tiennent aussi pour le nombre de paramètres correspondants aux filtres de formation de

Algorithme 9.1 Décodeur gaussien paramétrique pour la séparation informée.

Entrées :
- I signaux régulièrement échantillonnés de mélange $\tilde{\boldsymbol{x}}$
- Paramètres ρ et L_0 de tramage
- Famille paramétrique $\mathcal{P}$ utilisée pour les DSP des sources
- Flux binaire de l'information annexe

Initialisation
- Construire la TFCT $\boldsymbol{x}$ des mélanges
- Construire l'information annexe $\Theta = \left\{\theta\,,\, \{R_j\,(f)\}_{f,j}\,,\, \{U_j\,(f)\}_{j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d}\right\}$ à partir de son flux binaire

Séparation
- Construire les matrices $K\left(\boldsymbol{z}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right),\boldsymbol{x}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right)\right)$ et $K\left(\boldsymbol{x}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right),\boldsymbol{x}\left(\boldsymbol{f},\boldsymbol{n},\cdot\right)\right)$ selon 9.2.3 et 9.2.2 page 123
- Estimer les signaux $\boldsymbol{z}$ en appliquant 9.2.6 page ci-contre pour chaque point (f,n)
- Récupérer les formes d'ondes correspondantes $\tilde{\boldsymbol{z}}$ par TFCT inverse

Sortie
- Retourner $\tilde{\boldsymbol{z}}$

voie $\{U_j\,(f)\}_{f,j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d}$, qu'on peut également transmettre sous forme de réponse impulsionnelle. Dans tous les cas, je n'envisagerai pas d'autre stratégie particulière dans la présente étude pour réduire le nombre de ces paramètres de mixage, considérant que les débits correspondants sont satisfaisants pour les applications.

Pour un morceau stéréo de 3mn composé de 2 sources diffuses et de 3 sources ponctuelles dont les filtres de mélange sont d'un ordre $H = 200$, une quantification en 32 bits de chaque scalaire réel conduit à un débit pour les paramètres de mixage de 0.3kbps/source, sans tenir compte du gain apporté par un codage entropique ultérieur. Ceci est très faible devant le débit requis par les autres paramètres du modèle, de l'ordre de 5kbps.

On remarque par contre que le nombre de paramètres contenus par Θ dans 9.1.7 est très largement dominé par la valeur des JFN coefficients des DSP des sources, qu'on ne peut pas encoder de manière naïve par quantification scalaire sans aboutir à des débits prohibitifs. Cependant, j'ai montré au chapitre 4 qu'il existe des modèles permettant de réduire énormément le nombre de paramètres requis pour la modélisation de la DSP d'un groupe de signaux. La stratégie adoptée dans ma thèse a consisté à utiliser de tels modèles de sources en lieu et place des véritables DSP et ainsi à simplement remplacer P dans 9.1.7 par un *modèle de sources* $\mathcal{P}\,(\cdot\mid\theta)$:

$$P\,(f,n,j) = \mathcal{P}\,(f,n,j\mid\theta)\,. \tag{9.3.1}$$

Pour le même exemple, si on choisit un modèle NTF à $K = 20$ composantes, le nombre de paramètres de sources passe de $JFN \approx 1.3\times10^7$ à $K\,(J+F+N) \approx 1.1\times10^5$, soit une diminution de deux ordres de grandeur.

Il est remarquable que n'importe quel modèle de sources puisse être utilisé à ce stade, paramétré par un ensemble θ de paramètres et qui permet d'obtenir une estimation de la DSP $P\,(f,n,j)$ de chaque source à chaque point TF par application de 9.3.1.

Lors de mon travail, je me suis contenté des deux modèles NTF et CI que j'ai présentés au chapitre 4 page 57, mais il est entendu qu'une réduction encore plus importante du nombre de paramètres de l'information annexe peut être obtenue en utilisant des modèles plus sophistiqués tels que ceux présentés en [53, 163].

Une fois le modèle $\mathcal{P}\,(\cdot\mid\theta)$ de sources choisi, il s'agit d'en estimer les paramètres θ et de procéder à leur quantification de manière à pouvoir transmettre l'information annexe du codeur vers

le décodeur. J'envisagerai cette problématique plus avant au chapitre 10.

Dans tous les cas, l'information annexe oracle 9.1.7 est remplacée en pratique par :

$$\Theta = \left\{ \theta \,,\, \{R_j\,(f)\}_{f,j} \,,\, \{U_j\,(f)\}_{j \in \mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d} \right\}, \tag{9.3.2}$$

où θ désigne le lot de paramètres de sources, à partir desquelles il est possible de reconstruire une approximation $\mathcal{P}\,(f, n, j \mid \theta)$ de la DSP des signaux par le biais d'un modèle de sources $\mathcal{P}\,(\cdot \mid \theta)$ donné.

Au niveau du décodeur, l'utilisation d'un modèle de sources différent de celui de la configuration oracle ne pose pas de problème particulier. En effet, muni de cette nouvelle information annexe 9.3.2, il est simple de procéder à la séparation en remplaçant l'expression exacte $P\,(f, n, j)$ par son approximation $\mathcal{P}\,(f, n, j \mid \theta)$ dans les expressions menant aux estimées. La complexité de ce traitement au décodeur est la même que dans la configuration oracle.

9.4 Un codage paramétrique

Il est intéressant d'établir ici un parallèle [4] entre l'expression 9.1.8 page 122 des sources estimées au décodeur et l'utilisation de modèles paramétriques dans le codage audio à bas débit [33].

Dans un codeur paramétrique, on suppose que le signal à encoder est un représentant d'une famille paramétrique connue. Le rôle de l'encodeur est d'identifier ce représentant à partir de l'observation du signal, et d'encoder les paramètres correspondants. Au décodeur, le signal est reconstruit en utilisant les paramètres produits par l'encodeur. Par exemple, il est courant de modéliser un signal de voix comme produit par filtrage autorégressif d'une combinaison de signaux périodiques d'excitation, qui correspondent aux cycles vibratoires de la glotte. Cette idée a conduit à l'élaboration de plusieurs formats d'encodage très populaires des signaux vocaux, dont CELP, MELP et leurs nombreuses variantes [33].

L'intérêt d'un codeur paramétrique est qu'il est souvent moins coûteux, en termes de débit, de transmettre les paramètres d'un modèle génératif que de trouver un moyen de reconstruire fidèlement n'*importe quelle* forme d'onde. En effet, le nombre de paramètres des modèles utilisés pour le codage est généralement assez faible, ce qui permet de les transmettre très efficacement. De plus, si on sait que les signaux à compresser seront toujours d'un type particulier, il est intéressant de tirer parti de cette connaissance pour réduire le débit nécessaire à leur encodage.

Si on fait l'hypothèse que le signal est une droite, il est bien plus efficace de le décrire avec sa pente et son ordonnée à l'origine plutôt qu'en cherchant à décrire chacun de ses échantillons. C'est le principe d'un codeur paramétrique.

L'inconvénient de ces codeurs est que si le signal à encoder n'obéit pas au modèle paramétrique choisi, la distorsion introduite par l'encodage sera très forte. En effet, le signal produit au décodeur est par construction un représentant de la famille paramétrique considérée. Si le signal observé est très mal modélisé par cette famille, il est inévitable d'avoir une très mauvaise qualité de reconstruction au décodeur. Dans les codeurs de la parole, il est par exemple classique d'utiliser deux modèles différents pour les portions voisées et les portions non voisées, qui ne sont pas le fruit du même processus physique de synthèse. Il est alors crucial de fidèlement identifier quelles portions du signal sont voisées et quelles portions ne le sont pas, de manière à ne jamais utiliser le mauvais modèle lors de l'encodage. En cas d'erreur, le signal reconstruit perd beaucoup en intelligibilité. Un autre exemple classique d'inadéquation d'un signal aux modèles est celui de la musique, qui est très mal compressée par un codeur paramétrique de voix. Cela se comprend aisément par le constat que la musique présente souvent une multitude de fréquences fondamentales simultanées, ce qui n'est pas pris en compte dans les codeurs spécifiques à la voix.

4. Je remercie Alexey Ozerov de m'avoir indiqué ce parallèle.

Le système proposé repose sur un modèle génératif où les sources sont des PGLS, mixés dans les signaux de mélanges. Leurs estimées sont données par une forme paramétrique. En ce sens, il se rapproche d'un codeur paramétrique.

Or, le système que je propose dans cette partie pour la séparation informée est bien un modèle paramétrique des signaux sources. En effet, l'expression 9.1.8 des estimées montre que les signaux encodés ne peuvent pas être quelconques, mais doivent correctement obéir au modèle de sources et doivent donc être des PGLS dont la DSP est correctement modélisée par $\mathcal{P}(\cdot \mid \theta)$. De plus, les signaux mélangés à partir desquels la séparation est effectuée doivent être formés à partir des signaux sources selon le modèle de mixage choisi et en utilisant les paramètres de mixage utilisés pour la séparation. Cet encodage paramétrique des sources est cependant d'un genre nouveau, puisqu'il est *conditionné* à l'observation d'un mélange $\tilde{\boldsymbol{x}}$, commun à la fois à l'encodeur et au décodeur, par opposition aux codeurs traditionnels, qui ne supposent pas une telle observation commune.

Ce système bénéficie ainsi des mêmes avantages et souffre des mêmes inconvénients que les codeurs paramétriques classiques. Comme on le verra au chapitre 12, il permet des débits pour la transmission des sources de l'ordre de $1-5$kbps/source, ce qui est très faible pour la transmission de signaux musicaux et très intéressant pour les applications d'écoute active envisagées.

Son principal inconvénient est que les signaux estimés ne peuvent pas être identiques aux signaux originaux, quel que soit le débit disponible pour la transmission de l'information annexe [5]. En d'autres termes, les performances de cette approche paramétrique sont *bornées*. Deux raisons principales peuvent être évoquées pour expliquer ce fait.

En premier lieu, quand bien même le modèle choisi serait parfaitement respecté par les signaux, l'expression 9.1.8 ne garantit que la minimisation *en moyenne* de l'erreur quadratique, et non pas la récupération exacte des signaux mélangés. La covariance *a posteriori* correspondante 9.2.7 n'est en effet pas nulle, ce qui indique clairement que le modèle ne prétend pas à une quelconque *certitude* sur les estimées.

En outre, le système repose sur plusieurs approximations qui peuvent ne pas être respectées exactement en pratique :

- Les signaux sources sont supposés être des processus gaussiens localement stationnaires. Bien que cette hypothèse se justifie souvent, elle demeure une approximation. On a vu en effet en section 3.3 que le recouvrement entre les trames interdit en toute rigueur de supposer les différents points TF comme indépendants, ce qu'on fait pourtant dans les traitements.
- Lors de l'utilisation du modèle convolutif, on suppose l'exactitude des modèles de mixage 6.1.15 et 6.2.1. Cependant, ces modèles reposent sur certaines approximations qui peuvent ne pas être respectées. En particulier, il est courant d'avoir des filtres de mélange dont les réponses impulsionnelles ne sont pas finies. Dans ce cas, les modèles de mixage utilisés ne sont qu'une approximation.
- Quand on remplace les DSP des signaux sources par leur approximation paramétrique en 9.3.1, on commet inévitablement une erreur qui produit un écart entre les signaux originaux et leur approximation 9.1.8.

Malgré toutes ces causes possibles de défaillance, le modèle de séparation paramétrique que je propose dans cette partie est à la fois simple à implémenter, offre des bonnes performances et conduit à des débits extrêmement compétitifs pour la séparation informée. Le rapprochement effectué ici entre ce système et les codeurs paramétriques permet de bien cerner à la fois les forces et les limites de l'approche.

9.5 Structure de la chaine de traitement

Pour conclure ce chapitre introductif sur l'approche paramétrique que je propose pour la séparation informée, je vais préciser l'architecture du système correspondant, que j'ai représentée en figure 9.2.

5. Sauf cas exceptionnels assez irréalistes en pratique.

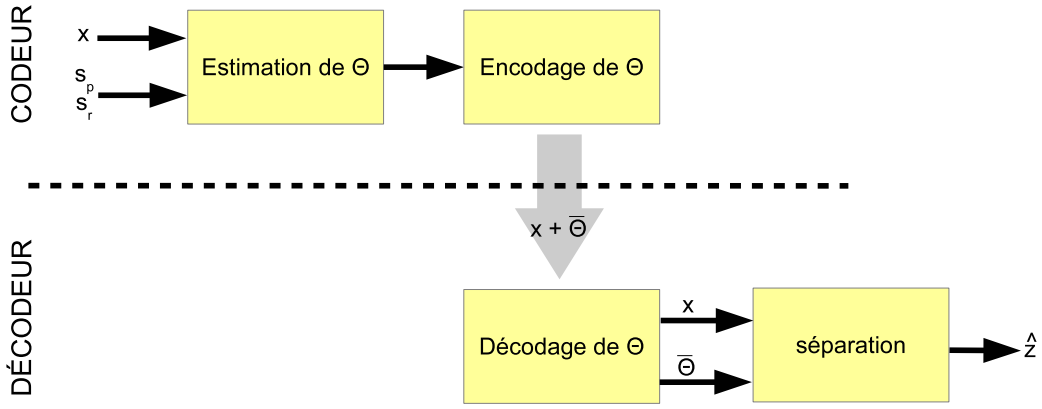

FIGURE 9.2: Structure du système de séparation informée paramétrique. L'information annexe Θ est produite après analyse conjointe à l'encodeur des sources et des mélanges, puis utilisée au décodeur pour estimer les sources à partir des mélanges.

- **Le codeur** dispose de la connaissance des mélanges, des sources ponctuelles et des sources diffuses. Muni de ces informations, il produit dans un premier temps l'information annexe Θ. Dans un deuxième temps, cette information annexe est encodée, de manière à générer un *flux annexe*, entendu comme une suite de bits transmise au décodeur en plus des mélanges. La version quantifiée de l'information annexe est notée $\bar{\Theta}$.
- **Le décodeur** récupère à la fois les mélanges et le flux annexe et commence par décoder le flux pour reconstruire l'information annexe quantifiée $\bar{\Theta}$. Lorsqu'il dispose des mélanges $\boldsymbol{x}$ et de $\bar{\Theta}$, il procède à la séparation en estimant les signaux $\boldsymbol{z}$ recherchés par 9.2.6, où les matrices de covariance utilisées sont définies en section 9.2 en utilisant $\bar{\Theta}$ en lieu et place des paramètres de sources et de mixage.

Cette architecture étant posée, il me reste à préciser en détail les deux traitements principaux effectués par le codeur : l'estimation et l'encodage de l'information annexe. Ce point fait l'objet des deux chapitres suivants.

Chapitre 10

Codeur informé paramétrique ($S_d = M_d = 0$)

Dans ce chapitre, je vais m'intéresser aux traitements effectués par l'encodeur dans le système paramétrique. De manière à simplifier mon exposé, j'ai décidé d'aborder le problème de l'estimation de l'information annexe en deux temps. Pour commencer, je vais considérer dans ce chapitre le cas où aucune source n'est observée au codeur sous forme diffuse, ni mixée de manière diffuse dans le mélange, conduisant à $S_d = M_d = 0$. Ce cas constitue l'état de l'art dans le domaine de la séparation informée et généralise déjà tous les systèmes dont j'ai connaissance. En effet, il permet en premier lieu les mélanges convolutifs, alors que de nombreux systèmes existants sont restreints aux mélanges linéaires instantanés [1] [170, 130, 167, 87]. De plus, il inclut à la fois la récupération des images et des sources ponctuelles, une propriété que je n'ai pas encore rencontrée chez un autre système [2]. Dans cette configuration où $S_d = 0$ et $M_d = 0$, l'information annexe 9.3.2 à estimer devient :

$$\Theta = \left\{ \theta \, , \, \{A_j \, (f)\}_{j,f} \right\}, \tag{10.0.1}$$

et se réduit donc aux paramètres de sources θ et aux filtres de mélange A.

Comme on l'a vu en section 9.1, il est toujours possible de se ramener au cas $S_d = 0$ en assimilant chaque source diffuse à I sources ponctuelles.

Comme souligné en figure 9.2, cet encodeur a deux tâches principales à effectuer : l'estimation de l'information annexe et son encodage. Dans un premier temps, j'aborderai en section 10.1 la question de la stratégie générale adoptée pour l'apprentissage de l'information annexe. On y verra qu'au lieu de s'atteler à un apprentissage *discriminant*, optimal mais complexe compte tenu de la paramétrisation du problème, la stratégie adoptée sera *générative*, plus simple et permettant de découpler l'apprentissage des paramètres de sources et de mixage. Une fois cette stratégie établie, je montrerai comment on peut apprendre les paramètres en section 10.2. C'est alors seulement que j'aborderai le problème de leur encodage en section 10.3. Je terminerai en section 10.4 par un résumé des opérations effectuées par l'encodeur paramétrique dans le cas $S_d = M_d = 0$.

10.1 Approche discriminante ou générative

L'objectif de l'apprentissage est de produire l'information annexe $\Theta^\star$ qui sera la plus susceptible de conduire à des bonnes estimées des sources au moment du décodage. Dans un cadre probabiliste,

1. Cette affirmation doit être cependant tempérée par le fait que la plupart peuvent être étendus au cas convolutif en considérant simplement qu'il est équivalent à un mixage instantané dont la matrice de mélange varie en fonction de la fréquence. Cela a été démontré dans l'évaluation commune [134] que nous avons menée et dont on trouvera les résultats au chapitre 12.

2. Dans le cas ponctuel, il faut cependant remarquer qu'on peut très bien remixer les sources avec leurs filtres de mélange pour obtenir les images. Il serait donc équivalent de récupérer toutes les sources sous forme ponctuelle pour remixer ensuite celles dont on souhaite les images. L'intérêt de l'approche n'apparaîtra clairement que lorsqu'on considérera le cas des mélanges diffus au chapitre suivant.

cet apprentissage peut se formuler comme la recherche de $\Theta^\star$ qui maximise la vraisemblance des signaux recherchés dans leur distribution *a posteriori* $p(\boldsymbol{z} \mid \boldsymbol{x}, \Theta)$ étant donnés les mélanges et Θ :

$$\begin{aligned} \Theta^\star &= \underset{\Theta}{\operatorname{argmax}} \log\left(p(\boldsymbol{z} \mid \boldsymbol{x}, \Theta)\right), \\ &= \underset{\Theta}{\operatorname{argmax}} \sum_{f,n} \log\left(p(\boldsymbol{z}(f,n,\cdot) \mid \boldsymbol{x}(f,n,\cdot), \Theta)\right) \end{aligned} \tag{10.1.1}$$

où on a vu en section 9.2 que pour chaque point TF, la distribution *a posteriori* $p(\boldsymbol{z}(f,n,\cdot) \mid \boldsymbol{x}(f,n,\cdot), \Theta)$ des sources est gaussienne, de moyenne $\boldsymbol{\mu}_{\text{post}}(f,n)$ défini en 9.2.6 et de covariance $K_{\text{post}}(f,n)$ défini en 9.2.7 page 124. En utilisant 2.4.3 page 34, la maximisation de 10.1.1 est donnée par :

$$\Theta^\star = \underset{\Theta}{\operatorname{argmin}} \sum_{f,n} \left(\left(\boldsymbol{z}(f,n,\cdot) - \boldsymbol{\mu}_{\text{post}}(f,n) \right)^H K_{\text{post}}(f,n) \left(\boldsymbol{z}(f,n,\cdot) - \boldsymbol{\mu}_{\text{post}}(f,n) \right) \right) + \sum_{f,n} \ln \left| K_{\text{post}}(f,n) \right|. \tag{10.1.2}$$

Comme on le voit, la recherche de l'information annexe par l'encodeur peut ainsi être présentée comme un problème d'optimisation. La minimisation de l'expression 10.1.2 conduit au choix de Θ qui produit les meilleures sources estimées par séparation du mélange. En effet, c'est lui qui conduit à la distribution *a posteriori* dans laquelle les vraies sources sont les plus probables. Cette méthode d'apprentissage ne se préoccupe pas tant de rendre compte du processus *génératif* par lequel les sources sont produites puis mixées pour former les mélanges, mais plutôt de la meilleure manière de *récupérer* les signaux recherchés compte tenu des mélanges. En ce sens, on peut s'inspirer du vocabulaire couramment utilisé dans le domaine de la classification automatique [184, 178] et parler ici d'un apprentissage *discriminant* de l'information annexe, par opposition à un apprentissage *génératif*, qui rechercherait plutôt l'information annexe qui permet au mieux d'expliquer la *production* des données.

En classification, un apprentissage *discriminant* cherche à modéliser la meilleure manière de directement classer les données, tandis qu'un apprentissage *génératif* cherche à expliquer au mieux le processus de synthèse des données.

Malheureusement, le problème posé par 10.1.2 se heurte à plusieurs difficultés importantes. Tout d'abord, les signaux $\boldsymbol{z}$ recherchés ne sont pas nécessairement observés au codeur. Ce n'est le cas que si on souhaite récupérer uniquement des sources ponctuelles ($Z_y = 0$), auquel cas $\boldsymbol{z} = \boldsymbol{s}$. Même dans ce cas, le problème 10.1.2 présente une complexité importante et je ne l'ai pas abordé dans le cadre de mon travail. De mon point de vue, la raison pour laquelle cette optimisation est si complexe réside dans le fait que toute sa paramétrisation repose sur une compréhension résolument *générative* de la nature des signaux, où les paramètres de sources θ régissent la synthèse de $\boldsymbol{s}$, tandis que les paramètres de mixage rendent compte de la production des mélanges $\boldsymbol{x}$ à partir de $\boldsymbol{s}$. Pour qu'une approche discriminante telle que 10.1.1 conduise à des calculs plus simples pour le codeur, je pense qu'il m'aurait fallu directement modéliser les sources comme produites par un filtrage des mélanges, et chercher les filtres qui permettent au mieux de les expliquer [3]. Je laisse cependant une telle étude à des prolongements de mon travail.

Dans ces conditions, j'ai fait le choix de plutôt m'orienter vers un apprentissage *génératif* de l'information annexe, beaucoup plus simple. Un tel apprentissage ne cherchera pas à maximiser la vraisemblance des sources dans leur distribution *a posteriori*, mais plutôt à rendre compte correctement de la procédure de synthèse à la fois des sources et des mélanges compte tenu du modèle. En termes probabilistes, cela peut s'écrire comme la recherche de l'information annexe $\Theta^\star$ qui maximise la vraisemblance $p(\boldsymbol{s}, \boldsymbol{x} \mid \Theta^\star)$ des signaux observés à l'encodeur, conduisant à un nouveau problème d'optimisation :

$$\Theta^\star = \underset{\Theta}{\operatorname{argmax}}\, p(\boldsymbol{s}, \boldsymbol{x} \mid \Theta),$$

3. Cette approche discriminante est celle que m'a indiquée Roland Badeau comme piste initiale au cours de la première semaine de mon doctorat. Ce sont des résultats prometteurs mettant en œuvre l'approche générative qui ont détourné mon attention de cette excellente idée jusqu'à ce jour.

qui peut se simplifier compte tenu de l'indépendance conditionnelle entre les mélanges $\boldsymbol{x}$ et les paramètres θ si les sources $\boldsymbol{s}$ sont connues :

$$\Theta^\star = \underset{\Theta}{\operatorname{argmax}} \underbrace{p\left(\boldsymbol{s} \mid \theta\right)}_{\text{sources}} \underbrace{p\left(\boldsymbol{x} \mid \boldsymbol{s}, A\right)}_{\text{mixage}}. \tag{10.1.3}$$

Comme on le voit, les problème à résoudre devient très similaire à celui déjà présenté en section 7.3, avec l'importante différence qu'à présent, les sources sont connues, en plus des mélanges. En figure 10.1, j'ai représenté cette différence de perspective entre une approche discriminante et une approche générative pour l'apprentissage de l'information annexe.

Comme on le voit, dans un apprentissage génératif, la détermination de l'information annexe Θ peut se faire en deux temps. Tout d'abord, on détermine les paramètres θ du modèle paramétrique des DSP qui maximisent la vraisemblance $p\left(\boldsymbol{s} \mid \theta\right)$ des signaux sources. Ensuite, on estime les paramètres A qui permettent au mieux d'expliquer les mélanges $\boldsymbol{x}$ comme un mixage convolutif des sources.

10.2 Apprentissage de l'information annexe

10.2.1 DSP des sources

Le premier problème auquel fait face l'encodeur paramétrique est celui de l'estimation des paramètres du modèle choisi pour les DSP des sources à partir de l'observation $\boldsymbol{s}$ d'une de leurs réalisations :

$$\theta^\star = \underset{\theta}{\operatorname{argmax}}\, p\left(\boldsymbol{s} \mid \theta\right).$$

J'ai déjà abordé cette problématique en section 4.1 page 57 dans le cas très général où il s'agit d'apprendre les paramètres d'un modèle paramétrique $\mathcal{P}\left(\cdot \mid \theta\right)$ quelconque et j'ai donné sa solution explicite pour les deux modèles de sources NTF et CI que je considèrerai dans les applications. L'apprentissage des paramètres d'un modèle NTF se fait en utilisant l'algorithme 4.1 page 67, tandis que le modèle CI est optimisé par une simple transformation des (log-)spectrogrammes des sources selon 4.2.3 page 61.

10.2.2 Paramètres de mixage

Le deuxième problème d'apprentissage à considérer est celui de l'estimation des filtres de mélange [4]. Comme on l'a vu en section 6.1.3 page 87, le mélange convolutif de PGLS peut être approximé dans le domaine de la TFCT par un mixage linéaire instantané, dont la matrice de mélange $A\left(f\right)$, de dimension $I \times J$, dépend de l'indice de fréquence f considéré :

$$\forall\left(f,n\right),\, \boldsymbol{x}\left(f,n,\cdot\right) = A\left(f\right)\boldsymbol{s}\left(f,n,\cdot\right).$$

Dans le cas où à la fois les sources et les mélanges sont connus, il est possible d'estimer $\widehat{A\left(f\right)}$ qui minimise l'erreur quadratique moyenne entre $\widehat{A\left(f\right)}\boldsymbol{s}\left(f,n,\cdot\right)$ et $\boldsymbol{x}\left(f,n,\cdot\right)$ en utilisant des techniques classiques de régression linéaire [116] :

$$\widehat{A\left(f\right)} = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}\left(f,n,\cdot\right)\boldsymbol{s}\left(f,n,\cdot\right)^H \left(\boldsymbol{s}\left(f,n,\cdot\right)\boldsymbol{s}\left(f,n,\cdot\right)^H\right)^{-1}. \tag{10.2.1}$$

Les estimées 10.2.1 permettent d'obtenir les matrices de mélange $\widehat{A\left(f\right)}$ qui expliquent au mieux les mélanges comme des combinaisons linéaires des sources selon le critère de l'erreur quadratique

4. De nombreuses techniques de la littérature supposent ces filtres de mélange connus. Une telle configuration est bien entendu possible dans le formalisme que je propose.

moyenne. Si on sait que le mixage n'est pas convolutif mais plutôt linéaire instantané comme ceux considérés au chapitre 5, alors l'unique matrice de mélange A peut être estimée par :

$$\widehat{A} = \frac{1}{FN} \sum_{f,n} \boldsymbol{x}\left(f,n,\cdot\right) \boldsymbol{s}\left(f,n,\cdot\right)^H \left(\boldsymbol{s}\left(f,n,\cdot\right) \boldsymbol{s}\left(f,n,\cdot\right)^H \right)^{-1}. \tag{10.2.2}$$

Un inconvénient de cette approche est qu'elle ne permet pas la prise en compte d'une éventuelle connaissance sur l'ordre H de la réponse impulsionnelle $a_{ij}\left(\tau\right)$ des filtres de mélange, hormis dans le cas $H = 0$ pour un mélange instantané. L'avantage d'estimer des réponses impulsionnelles plutôt que des matrices de mélange fréquentielles réside principalement dans la diminution correspondante du nombre de paramètres de mixage, qui passe de FIJ à HIJ.

Dans le cas où ce sont les réponses impulsionnelles des filtres qui doivent être estimées, il est possible soit de considérer une transformée de Fourier inverse des réponses fréquentielles obtenues par 10.2.1, soit d'utiliser une technique classique qui consiste à exprimer l'équation temporelle de mélange convolutif

$$\tilde{\boldsymbol{x}}\left(t,i\right) = \sum_{j=1}^{J} \sum_{\tau=0}^{H-1} a_{ij}\left(\tau\right) \tilde{\boldsymbol{s}}\left(t-\tau,j\right) \tag{10.2.3}$$

sous la forme d'une multiplication matricielle

$$\tilde{\boldsymbol{x}}\left(\cdot,i\right) = \tilde{\boldsymbol{S}}\tilde{A}, \tag{10.2.4}$$

où $\tilde{\boldsymbol{S}}$ est une matrice de dimension $(L+H)\times JH$ contenant les H versions retardées $\{\tilde{\boldsymbol{s}}\left(t-\tau,j\right)\}_{\tau=0,\ldots H-1}$ des sources [5] et $\tilde{A}$ est une matrice de dimension $JH \times I$ contenant les réponses impulsionnelles des filtres de mélange. Ces matrices sont construites très simplement à partir des signaux sources $\tilde{\boldsymbol{s}}$ et des réponses impulsionnelles $a_{ij}\left(\cdot\right)$ des filtres de mélange, comme indiqué sur la figure 10.2.

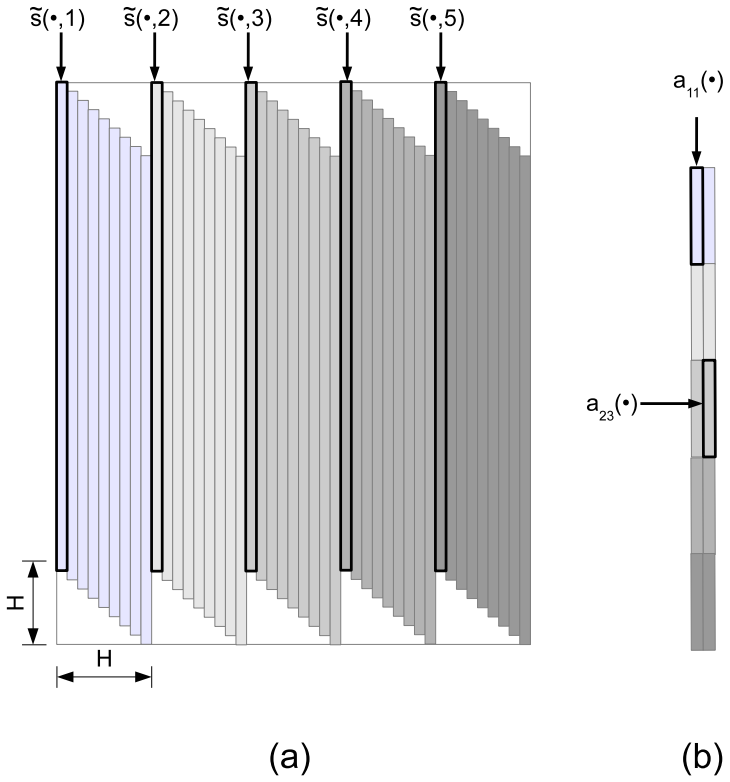

FIGURE 10.2: Structure des matrices (a) $\tilde{\boldsymbol{S}}$ et (b) $\tilde{A}$ par lesquelles un mélange convolutif en temporel 10.2.3 peut s'exprimer comme un produit matriciel 10.2.4.

5. L désigne toujours la longueur des signaux.

Dans ces conditions, on peut appliquer la même méthode que précédemment pour l'estimation de la matrice $\bar{A}$, de manière à obtenir :

$$\widehat{\bar{A}} = \left(\tilde{\boldsymbol{S}}^H \tilde{\boldsymbol{S}}\right)^{-1} \tilde{\boldsymbol{S}}^H \tilde{\boldsymbol{x}}, \tag{10.2.5}$$

où $\tilde{\boldsymbol{x}}$ désigne la matrice de dimension $(L+H) \times I$ contenant les observations temporelles des mélanges, auxquelles on a adjoint H zéros finaux [6]. L'expression des réponses impulsionnelles $a_{ij}(\tau)$ se déduit facilement de l'estimée 10.2.5.

10.3 Quantification et encodage

Une fois les paramètres de sources θ et de mixage A estimés, on dispose de tous les éléments nécessaires pour construire l'information annexe 10.0.1. De manière à pouvoir transformer cette information annexe en un *flux binaire* de données susceptible d'être envoyé au décodeur, il est nécessaire de tout d'abord quantifier les paramètres ainsi trouvés, puis de les encoder. Je désignerai par $\bar{\Theta} = \left\{\bar{\theta}, \bar{A}\right\}$ l'information annexe quantifiée.

Comme je l'ai déjà signalé en section 9.3 page 124, le coût en débit dû à une quantification scalaire naïve des paramètres de mixage A est assez faible, compte tenu du fait que le nombre de ces paramètres ne dépend pas de la longueur du signal traité. Par conséquent, la stratégie que j'ai adoptée dans mes travaux pour les transmettre au décodeur a simplement consisté à leur appliquer une quantification uniforme sur 32 bits, puis à encoder la série de codes résultants en utilisant un codage de Huffman [108] (voir section 13.1.3 page 169).

Le problème de la quantification et de l'encodage des paramètres de sources θ est plus délicat. Compte tenu du fait que leur nombre dépend de la longueur du morceau de musique considéré, il est important de parvenir à une stratégie efficace pour leur encodage.

La stratégie que j'ai adoptée pour l'encodage des paramètres θ a reposé sur une heuristique, jusqu'à ma collaboration avec Alexey Ozerov, qui lui a donné une base théorique solide à l'occasion de notre article commun [161]. Dans cette partie, je donne une autre justification théorique, peut être plus simple, de cette heuristique.

Au cours de mon travail sur la séparation informée, je ne me suis pas préoccupé tout de suite de la théorie sous-jacente au problème de l'encodage des paramètres de sources. La stratégie que j'ai adoptée dans ce but ainsi que pour la création du flux binaire correspondant a longtemps reposé sur des heuristiques [130, 137]. Pour le cas du modèle NTF, j'ai ainsi simplement procédé à une quantification uniforme sur 32 bits des paramètres $\theta = \{W, H, Q\}$ du modèle, suivie d'un encodage de Huffman. Dans le cas du modèle CI [130, 136], la stratégie adoptée a consisté à quantifier puis encoder les paramètres θ_{CI} 4.2.3 en utilisant simplement les compresseurs d'image standards, de type JPEG [216], appliqués sur les log-spectrogrammes des sources. Ces heuristiques se sont avérées déjà très efficaces, parvenant à des débits de l'ordre de $1-5$kbps/source [7].

Ce n'est que lors de mon travail en commun avec Alexey Ozerov que nous nous sommes posés la question des bases théoriques de cette étape de quantification puis d'encodage des paramètres de sources, dans le cadre général du codage informé que je présenterai en partie IV. D'une manière intéressante, ces développements ont principalement permis de montrer que les heuristiques suivies jusqu'alors étaient très proches de la manière optimale de procéder. En section 10.3.1, je vais présenter une justification simple de la stratégie générale choisie pour la quantification du modèle, avant de l'appliquer au modèle CI en section 10.3.2, puis au modèle NTF en section 10.3.3, pour lequel je préciserai aussi les différentes méthodes d'encodage que j'ai considérées.

10.3.1 Critère de quantification

Dans cette section, je suppose identifié le modèle de source θ qui permet au mieux de rendre compte des signaux sources $\boldsymbol{s}$ observés. La principale hypothèse que je ferai ici sera d'assimiler

6. On peut aussi envisager de tronquer $\tilde{\boldsymbol{s}}$.

7. Pour le modèle NTF, le débit dépend principalement du nombre K de composantes. Pour le modèle CI, il dépend du coefficient de qualité utilisé lors de la compression des log-spectrogrammes des sources.

simplement le spectrogramme des sources $v(f,n,j)$ avec la valeur paramétrique de leurs DSP, donnée par le modèle $\mathcal{P}(f,n,j \mid \theta)$ 4.1.1 :

$$\forall (f,n,j),\ v(f,n,j) \approx \mathcal{P}(f,n,j \mid \theta). \tag{10.3.1}$$

Cette hypothèse se justifie simplement en remarquant que c'est précisément l'objectif de l'apprentissage de θ que de minimiser l'écart entre le spectrogramme et le modèle, comme on l'a vu en section 4.1 page 57. On peut remarquer que dans le cas du modèle CI, qui n'est rien d'autre qu'une transformée du spectrogramme, l'approximation 10.3.1 est une égalité stricte.

Dans ces conditions, l'objectif de la quantification peut se comprendre comme la détermination des valeurs $\bar{\theta}$ des paramètres quantifiés qui maximisent à nouveau la vraisemblance des sources. La fonction de coût $\Psi\left(\bar{\theta}, v\right)$ à minimiser reste dans ce cas la même que 4.1.3 page 59 :

$$\Psi\left(\bar{\theta}, v\right) = \sum_{j,f,n} d_0\left(v(f,n,j) \mid \mathcal{P}\left(f,n,j \mid \bar{\theta}\right)\right),$$

où $d_0(a \mid b)$ est la distance d'Itakura-Saito 3.3.4 page 54 entre a et b. La différence est qu'au lieu d'utiliser le spectrogramme v des observations, je suggère d'utiliser à la place le modèle $\mathcal{P}(\cdot \mid \theta)$, comme l'indique 10.3.1. Cette fonction de coût devient alors :

$$\Psi\left(\bar{\theta}, \theta\right) = \sum_{j,f,n} d_0\left(\mathcal{P}(f,n,j \mid \theta) \mid \mathcal{P}\left(f,n,j \mid \bar{\theta}\right)\right). \tag{10.3.2}$$

Trouver une manière de quantifier θ qui minimise une telle expression est difficile. Une astuce dans ce but est de voir que si ses deux opérandes sont proches, la distance d'Itakura-Saito peut être approximée par une erreur quadratique entre leurs logarithmes. En effet, lorsque $a \approx b$, on a :

$$\begin{aligned} d_0(a \mid b) &= \frac{a}{b} - \log\frac{a}{b} - 1 \\ &\approx \left(1 + \log\frac{a}{b} + \frac{1}{2}\left(\log\frac{a}{b}\right)^2\right) - \log\frac{a}{b} - 1 \\ &= \frac{(\log a - \log b)^2}{2}, \end{aligned} \tag{10.3.3}$$

où cette approximation provient d'un développement limité du deuxième ordre $u \approx 1 + \log u + \frac{1}{2}(\log u)^2$ au voisinage de 1 pour $u = \frac{a}{b}$. En utilisant 10.3.3, le critère 10.3.2 devient :

$$\Psi\left(\bar{\theta}, \theta\right) \propto \sum_{j,f,n} \left(\log \mathcal{P}(f,n,j \mid \theta) - \log \mathcal{P}\left(f,n,j \mid \bar{\theta}\right)\right)^2, \tag{10.3.4}$$

qui est beaucoup plus facile à mettre en œuvre. En effet, si on décide de mettre en œuvre une procédure de quantification scalaire, c'est une quantification uniforme de $\log \mathcal{P}(f,n,j \mid \theta)$, de pas constant Δ_θ qui minimise l'erreur quadratique.

Ainsi, je viens de montrer moyennant certaines approximations qu'une manière adéquate de quantifier θ conduit à une quantification uniforme de $\log \mathcal{P}(\cdot \mid \theta)$. On va voir à présent que dans les deux cas d'un modèle CI ou NTF, cela peut être accompli simplement.

10.3.2 Modèle par compression d'image

Tel que je l'ai présenté en section 4.2, le modèle CI fait intervenir un tramage des spectrogrammes suivi d'une transformation spectrale comme la transformée de Fourier discrète ou une transformée en ondelettes. Une particularité de ce genre de transformées et de leurs inverses est qu'elles sont basées sur la multiplication du signal par une matrice orthonormée [8], qui a la propriété très intéressante dans notre contexte de préserver la norme Euclidienne [9].

8. Une matrice W est orthonormée si WW^H est l'identité.
9. La norme euclidienne $\|\boldsymbol{a}\|$ d'un vecteur $\boldsymbol{a}$ est la racine carrée de la somme des carrés de ses entrées.

Soient en effet $\boldsymbol{a}$ et $\boldsymbol{b}$ deux vecteurs de dimension $L \times 1$ et W une matrice orthonormée de dimension $L \times L$. On a :

$$\|W\boldsymbol{a} - W\boldsymbol{b}\| = \|W(\boldsymbol{a} - \boldsymbol{b})\| = \|\boldsymbol{a} - \boldsymbol{b}\|.$$

Or, compte tenu de l'expression 4.2.4 page 61 de $\mathcal{P}(\cdot \mid \theta)$, le critère à minimiser pour $\bar{\theta}$ devient [10] :

$$\Psi\left(\bar{\theta}, \theta\right) = \left\|\mathcal{C}^{-1}\{\theta\} - \mathcal{C}^{-1}\left\{\bar{\theta}\right\}\right\|,$$

qui est donc équivalent à

$$\Psi\left(\bar{\theta}, \theta\right) = \left\|\theta - \bar{\theta}\right\|.$$

Par conséquent, pour peu que le modèle CI considéré fasse intervenir des transformées *orthonormées* appliquées aux *log-spectrogrammes* des sources et qu'on procède à une quantification scalaire des paramètres, c'est une quantification uniforme de θ qui est adéquate pour produire $\bar{\theta}$.

La procédure à suivre pour utiliser le modèle CI de manière optimale serait d'encoder tous les log-spectrogrammes des sources avec un algorithme de type JPEG, modifié de telle manière à ce que la quantification des coefficients spectraux soit uniforme.

Ce n'est en effet pas le cas de JPEG, qui fait intervenir un pas de quantification plus grand pour les grandes fréquences spatiales, partant du constat que la vision humaine est moins sensible à l'introduction de distorsion aux emplacements des transitions sur une image. En utilisant les compresseurs JPEG standards dans [137], j'ai ainsi utilisé une méthode de quantification sous-optimale, mais néanmoins assez efficace et extrêmement facile à appliquer.

Une fois la quantification effectuée et compte tenu du fait que les différents coefficients de $\bar{\theta}$ peuvent être considérés comme indépendants, on peut appliquer un algorithme de codage entropique pour générer le flux binaire correspondant. Dans mon travail, j'ai simplement utilisé celui intégré dans les codeurs d'images que j'ai selectionnés.

10.3.3 Modèle NTF [11]

Si on réécrit le critère 10.3.4 en utilisant l'expression 4.3.2 page 62 du modèle correspondant, on obtient :

$$\Psi\left(\bar{\theta}, \theta\right) = \sum_{j,f,n} \left(\log \sum_{k=1}^{K} W(f,k)\, H(n,k)\, Q(j,k) - \log \sum_{k=1}^{K} \bar{W}(f,k)\, \bar{H}(n,k)\, \bar{Q}(j,k) \right)^2, \quad (10.3.5)$$

qui est difficile à exploiter pour la quantification de θ. Il est plus facile de considérer plutôt le critère $\Phi\left(\bar{\theta}, \theta\right)$ suivant :

$$\Phi\left(\bar{\theta}, \theta\right) = \sum_{j,f,n,k} \left(\log W(f,k)\, H(n,k)\, Q(j,k) - \log \bar{W}(f,k)\, \bar{H}(n,k)\, \bar{Q}(j,k)\right)^2, \quad (10.3.6)$$

dont on peut démontrer [161] qu'il est une majoration de 10.3.5 :

$$\Psi\left(\bar{\theta}, \theta\right) \leq \Phi\left(\bar{\theta}, \theta\right).$$

Maintenant, si on se souvient que le bruit de quantification est de moyenne nulle, que $\log \boldsymbol{ab} = \log \boldsymbol{a} + \log \boldsymbol{b}$ et qu'on suppose que les coefficients de W, H et Q sont quantifiés indépendamment,

10. Je rappelle que $\mathcal{C}^{-1}$ désigne l'ensemble des opérations par lesquelles l'approximation est reconstruite à partir des paramètres dans le modèle CI. Ces opérations font principalement intervenir des transformées fréquentielles inverses.

11. La discussion qui suit sur la quantification et l'encodage des coefficients du modèle NTF est extraite de notre article [161].

les termes croisés dans 10.3.6 s'annuleront en moyenne, pour peu que $K \times \min(J, F, N)$ soit suffisamment grand. En effet, le bruit de quantification de Q est par exemple indépendant de celui de H et de H lui-même. Par conséquent, on peut réécrire 10.3.6 de la manière suivante :

$$\Phi\left(\bar{\theta}, \theta\right) = \sum_{j,f,n,k} \big[\left(\log W(f,k) - \log \bar{W}(f,k)\right)^2 + \left(\log H(n,k) - \log \bar{H}(n,k)\right)^2 + \left(\log Q(j,k) - \log \bar{Q}(j,k)\right)^2 \big],$$

soit :

$$\Phi\left(\bar{\theta}, \theta\right) = JFN \sum_{k} \big[\frac{1}{F} \left(\log W(f,k) - \log \bar{W}(f,k)\right)^2 + \frac{1}{N} \left(\log H(n,k) - \log \bar{H}(n,k)\right)^2 + \frac{1}{J} \left(\log Q(j,k) - \log \bar{Q}(j,k)\right)^2 \big] \quad (10.3.7)$$

A l'examen de 10.3.7, on voit que si on fait le choix de quantifier indépendamment les coefficients du modèle NTF, il faut utiliser une compression logarithmique. De plus, on peut constater que les termes de W, H et Q apparaissent dans 10.3.7 pondérés de manière différente. Par conséquent, de manière à minimiser 10.3.7, W, H et Q doivent être divisés par la racine carrée de ces poids avant quantification uniforme par le même pas de quantification Δ_θ. Alternativement, ils peuvent être quantifiés de manière uniforme en utilisant des pas de quantification différents.

Pour quantifier indépendamment les coefficients $\theta = \{W, H, Q\}$ du modèle NTF, il faut appliquer un compresseur logarithmique suivi d'une quantification scalaire en utilisant les pas de quantification suivants pour W, H et Q :

$$\begin{aligned} \Delta_W &= \sqrt{F/(JFN)}\Delta_\theta \\ \Delta_H &= \sqrt{N/(JFN)}\Delta_\theta \\ \Delta_Q &= \sqrt{J/(JFN)}\Delta_\theta, \end{aligned} \quad (10.3.8)$$

où Δ_θ est un pas de quantification commun pour le modèle.

Ces résultats contrastent avec l'heuristique que j'avais choisie pour [130, 137], où ces coefficients étaient quantifiés sans compression logarithmique préalable. On peut aussi mettre en perspective ces développements par rapport à ceux de Nikunen *et al.* dans [151, 152], qui proposent d'utiliser un compresseur à loi μ, qui se comporte de manière logarithmique pour μ élevé. De plus, nous avons montré que les pas de quantification des paramètres W, H et Q se déduisent simplement d'un paramètre Δ_θ commun, alors que [151, 152] procèdent à une optimisation expérimentale de chaque pas de quantification.

Dans [130, 137], j'ai procédé à la quantification uniforme de W, H et Q directement, plutôt que de leur logarithme. Une telle procédure est donc sous-optimale à la lumière de ces développements, ce qui a été confirmé expérimentalement dans [161]. Pour l'encodage des paramètres du modèle, j'ai considéré un encodage de Huffman et l'utilisation d'un modèle de mélange de gaussiennes.

Une fois les paramètres NTF quantifiés, ils sont prêts à être encodés. Dans mon travail, j'ai considéré deux types d'encodage. Le premier est un codage de Huffman [108], le deuxième est un codage par modèle de mélange de gaussienne (MMG). Dans ce deuxième cas, abordé dans [161], nous avons considéré pour chaque matrice W, H et Q un MMG à 2 états, appris au sens du maximum de vraisemblance sur $\overline{\log W}$, $\overline{\log H}$ et $\overline{\log Q}$ ce qui produit en tout 15 paramètres à transmettre au décodeur (2 moyennes, 2 variances et 1 poids pour chacune de ces trois matrices). Il y a des avantages et des inconvénients à considérer l'une ou l'autre de ces techniques d'encodage. Si un encodage de Huffman est optimal, il nécessite la transmission d'un dictionnaire, ce qui peut être coûteux par rapport au faible nombre de paramètres requis par le MMG. Dans la

figure 10.3, j'ai représenté un exemple de modèle NTF appris sur trois sources audio ainsi que le MMG correspondant.

10.4 Algorithme de codage

De manière à conclure ce chapitre sur l'encodeur du système de séparation informée que je propose pour $S_d = M_d = 0$, j'ai résumé dans l'algorithme 10.1 les opérations correspondantes. Cet algorithme sera testé et évalué au chapitre 12.

Algorithme 10.1 Encodeur paramétrique pour la séparation informée selon le modèle gaussien, dans le cas $S_d = M_d = 0$.

Entrées :

- I signaux régulièrement échantillonnés de mélange $\tilde{\boldsymbol{x}}$
- J signaux régulièrement échantillonnés de sources $\tilde{\boldsymbol{s}}$
- Paramètres ρ et L_0 de tramage
- Famille paramétrique $\mathcal{P}$ utilisée pour les DSP des sources (incluant K pour le modèle NTF)
- Pas de quantification Δ_θ à utiliser pour le modèle de DSP des sources
- Filtres de mélanges s'ils sont connus. Leur ordre H au cas échéant.

Initialisation

- Construire les TFCT $\boldsymbol{x}$ et $\boldsymbol{s}$ des mélanges et des sources

Modèle de sources θ

- Si $\mathcal{P}$ est le modèle NTF
  - Estimer le modèle de sources $\theta = \{W, H, Q\}$ en utilisant l'algorithme 4.1
  - Quantifier uniformément $\log W$, $\log H$ et $\log Q$ en utilisant les pas de quantification 10.3.8
  - Encoder les paramètres en utilisant un codage de Huffman ou un MMG comme décrit en section 10.3.3
- Si $\mathcal{P}$ est le modèle CI
  - Compresser les log-spectrogrammes des sources en utilisant un algorithme de codage d'image (idéalement, utiliser une quantification uniforme au lieu de la quantification perceptuelle intégrée dans les codeurs standards)

Modèle de mélange

- Si les filtres de mélange sont inconnus, utiliser les résultats de la section 10.2.2 pour les estimer
- Les quantifier de manière uniforme sur 32 bits et utiliser un encodage de Huffman

Sortie

- Retourner le flux binaire correspondant à $\bar{\Theta} = \{\bar{\theta}, \bar{A}\}$

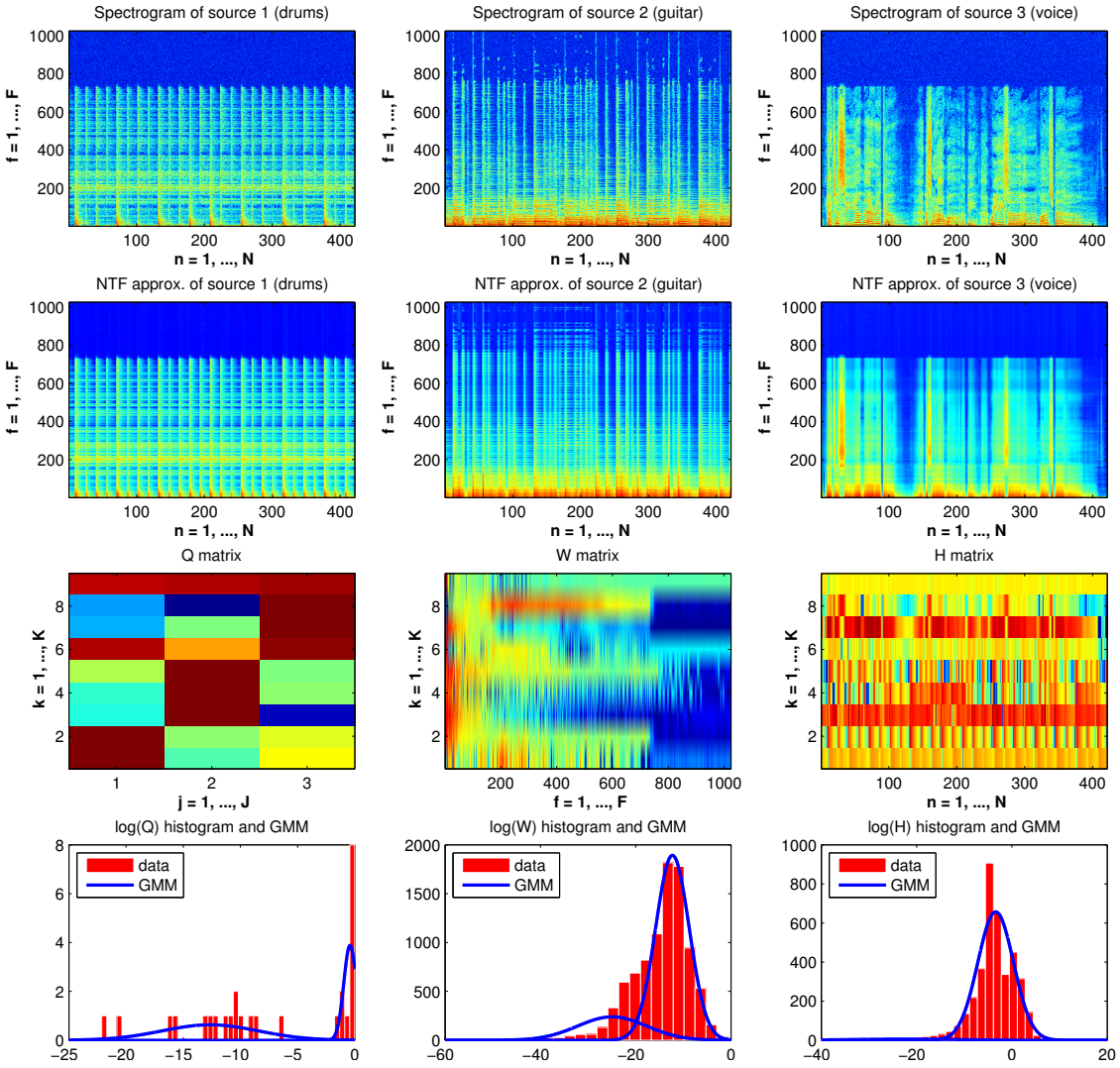

FIGURE 10.3: Spectrogrammes de trois sources $v(f,n,j)$ (première ligne), leurs approximations $\mathcal{P}(f,n,j \mid \theta)$ (deuxième ligne) selon un modèle NTF à $K = 9$ composantes dont les matrices W, H et Q sont représentées sur la troisième ligne. En quatrième ligne sont présentés les histogrammes de $\overline{\log W}$, $\overline{\log H}$ et $\overline{\log Q}$ ainsi que les MMG correspondants (d'après [161]).

Chapitre 11

Codeur informé paramétrique (cas général)

11.1 Introduction

11.1.1 Motivations

Dans le chapitre précédent, j'ai présenté le codeur paramétrique pour le cas particulier où aucun signal considéré ne fait intervenir de mixage diffus, ni dans les signaux observés au codeur, ni lors de la production des mélanges, ce qui se traduit par $S_d = 0$ et $M_d = 0$. En conséquence, le problème de l'estimation des paramètres à inclure dans l'information annexe s'est retrouvé considérablement simplifié. En effet, les signaux ponctuels à récupérer se confondent dans ce cas avec ceux observés au codeur ($\boldsymbol{s}_p = \boldsymbol{s}$). Ensuite, il est facile d'utiliser les résultats de la section 4.1 page 57 pour estimer les paramètres θ de sources. Les paramètres de mixage, réduits à des filtres de mélange, sont obtenus facilement par des techniques classiques de régression linéaire. Enfin, il n'est pas nécessaire d'estimer de quelconques filtres de formation de voie, puisqu'on a alors $\mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d = \varnothing$.

Dans un article de Nicolas Stürmel dont je suis un des co-auteurs [201], nous avons cherché à modéliser le mixage tel qu'effectué en studio. Le modèle convolutif y apparaît clairement comme souvent insuffisant. L'inclusion du modèle diffus dans le cadre qui y est proposé reste une piste de recherche.

Il y a cependant des inconvénients à se restreindre ainsi au cas des seuls mélanges convolutifs. Comme on l'a vu en section 6.1 page 83, ce n'est qu'au prix de l'hypothèse assez restrictive de filtres de mélange courts que l'assimilation 6.1.15 d'un mélange convolutif à un mélange instantané en fréquentiel est correcte. En section 6.2, j'ai présenté le formalisme diffus comme permettant justement de prendre en compte des filtres plus réalistes. Si un mixage ponctuel peut parfois déjà paraître discutable dans un contexte de propagation acoustique [47, 50, 49, 93] où il est pourtant justifié par des considérations physiques, il est à plus forte raison souvent irréaliste dans un contexte de production musicale, où l'opération de mixage peut faire intervenir des filtres d'une longueur considérable en raison d'une forte réverbération ou d'échos lointains ainsi que certains traitements non linéaires comme des *compressions*, qui sont autant de causes d'écarts à un simple modèle ponctuel.

C'est dans le but de pallier à ces inconvénients du modèle ponctuel que j'en ai présenté la généralisation diffuse en section 6.2. Je propose maintenant d'inclure cette extension dans le système de séparation de sources informée paramétrique. Cela pose un certain nombre de problèmes techniques. En premier lieu, l'apprentissage des modèles de sources ne se fait pas de manière aussi simple. Ensuite, l'estimation des paramètres de mélange est un peu plus complexe. Enfin, il est nécessaire d'estimer les filtres de formation de voie pour les sources de $\mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d$. Dans cette étude, je propose une approche simplifiée et sous-optimale pour l'ensemble de ces problèmes, en attendant que des travaux ultérieurs viennent leur donner une solution plus adéquate.

11.1.2 Stratégie adoptée

La stratégie globale que je propose pour l'encodeur paramétrique dans le cas général est résumée en figure 11.1 page ci-contre. De la même manière que dans le cas $S_d = M_d = 0$, j'adopte pour le cas général une approche générative plutôt qu'une approche discriminante. Cela se justifie d'autant plus ici que les signaux recherchés ne sont pas nécessairement observés à l'encodeur [1]. Il s'agira ainsi de déterminer l'information annexe Θ qui maximise :

La prise en compte du mixage diffus introduit des difficultés techniques au codeur. J'en propose ici une approche simplifiée.

$$p\left(\boldsymbol{s}_p, \boldsymbol{s}_d, \boldsymbol{x} \mid \Theta\right),$$

qu'on peut approximer comme :

$$p\left(\boldsymbol{s}_p, \boldsymbol{s}_d, \boldsymbol{x} \mid \Theta\right) \approx p\left(\boldsymbol{s}_p, \boldsymbol{s}_d \mid \theta\right) p\left(\boldsymbol{x} \mid \theta, \left\{A_j\left(f\right)\right\}_{f,j\in\mathcal{M}_c}, \left\{R_j\left(f\right)\right\}_{f,j\in\mathcal{M}_d}\right). \tag{11.1.1}$$

Pour l'optimisation du modèle de sources θ, je choisis de l'apprendre uniquement à partir des observations $\boldsymbol{s}_p$ et $\boldsymbol{s}_d$, alors qu'il intervient aussi dans la vraisemblance des mélanges. Mais même ainsi, le problème n'est pas trivial compte tenu du fait que les sources $\mathcal{S}_d$ sont observées par le biais d'un mélange diffus. Je propose alors de découpler cet apprentissage de θ en deux étapes. Tout d'abord, j'estimerai les DSP $P\left(f, n, j\right)$ de toutes les sources, puis ce n'est qu'alors que j'estimerai θ à partir de P. En d'autres termes, dans une première étape, j'estimerai les spectrogrammes des sources et dans une deuxième étape j'estimerai les paramètres permettant de correctement les approcher. Ce découplage est sous-optimal, parce qu'il serait possible de plutôt apprendre directement θ à partir des observations [2]. Cependant, il rend les optimisations correspondantes beaucoup plus simples. Je présenterai l'estimation de P à partir des observations en section 11.2 et celle des paramètres de sources en section 11.3.

Une fois θ appris, les paramètres de mixage seront déterminés par la seule observation des mélanges et de θ comme je le montrerai en section 11.4. Cette approche contraste donc avec la technique du chapitre 10 qui mettait plutôt en œuvre les mélanges et les sources. Cette différence se justifie par le fait que dans le cas général, les signaux sources menant aux mélanges ne sont pas nécessairement connus. S'ils le sont ($S_d = 0$), alors on peut utiliser cette connaissance pour mettre au point d'autres méthodes d'apprentissage similaires à celles du chapitre précédent. Je n'aborderai pas cette question ici.

Enfin, comme on peut le constater, les filtres de formation de voie permettant d'estimer les sources de $\mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d$ au décodeur, n'interviennent pas dans ce critère génératif 11.1.1. Je suggère donc de les apprendre indépendamment à partir de l'observation de $\tilde{\boldsymbol{s}}_p\left(\cdot, \mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d\right)$ et de $\tilde{\boldsymbol{x}}$ ainsi que de leur covariance 9.2.4, comme je le montrerai en section 11.5.

11.2 DSP des sources

Dans cette partie, j'aborde le problème de l'estimation des DSP des sources par le biais des signaux observés. Pour les sources ponctuelles $\boldsymbol{s}_p$, on a vu en section 3.3.2 page 53 et plus particulièrement en 3.3.5 page 54 que cela se fait très simplement en choisissant comme estimées les spectrogrammes v des signaux correspondants, entendus comme le module au carré de leurs TFCT.

La DSP des sources observées de manière ponctuelle est estimée trivialement par le spectrogramme des observations. Celle des sources diffuses nécessite un algorithme itératif.

1. Je rappelle que c'était déjà le cas pour $S_d = M_d = 0$ si $Z_y \neq 0$, c'est-à-dire si certaines sources étaient à récupérer sous forme d'images.
2. Dans le cas du modèle NTF, on peut par exemple utiliser à cet effet les techniques proposées dans [7, 163].

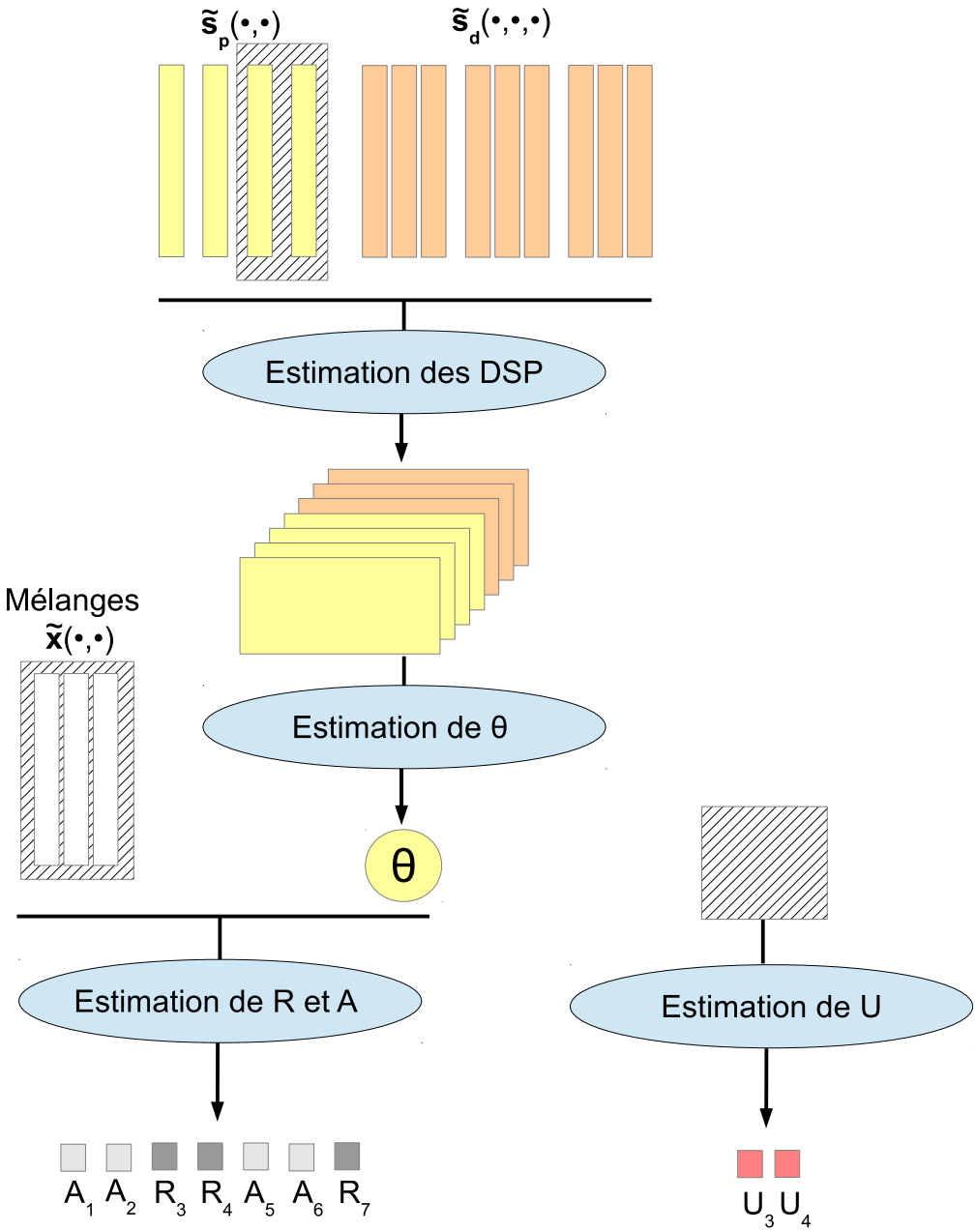

FIGURE 11.1: Stratégie générale proposée pour l'apprentissage de l'information annexe à partir des observations dans le cas général.
Dans cet exemple, on a $J = 7$ sources, dont les 4 premières sont observées de manière ponctuelle et les trois suivantes de manière diffuse, conduisant à $\mathcal{S}_p = \{1, 2, 3, 4\}$ et $S_d = \{5, 6, 7\}$. On suppose que les sources $\mathcal{M}_c = \{1, 2, 5, 6\}$ sont mixées de manière convolutive et les sources $\mathcal{M}_d = \{3, 4, 7\}$ sont mixées de manière diffuse. Enfin, les sources 3 et 4 sont à récupérer de manière ponctuelle alors qu'elles sont mixées de manière diffuse, d'où $\mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d = \{3, 4\}$.
La stratégie adoptée est la suivante : d'abord, on estime les DSP $P(f, n, j)$ des sources. Je détaille cette opération en section 11.2. Ensuite, on apprend les paramètres des sources à partir de ces DSP estimées. Cette étape est décrite en section 11.3. Muni de ces paramètres de sources et des mélanges, on apprend les paramètres de mixage. Cette étape est présentée en section 11.4. Enfin, on estime les filtres de formation de voie pour les sources de $\mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d$ à partir de leurs observations ponctuelles et des mélanges. Je montre comment en section 11.5.

Pour le cas des sources observées sous forme diffuse, l'estimation des DSP est moins triviale. Soit $j \in \mathcal{S}_d$ une de ces sources. Pour un point (f, n) donné, la covariance de $\boldsymbol{s}_d(f, n, \cdot, j)$ est donnée par 9.1.1 :

$$\mathbb{E}\left[\boldsymbol{s}_d(f, n, \cdot, j)\, \boldsymbol{s}_d(f, n, \cdot, j)^H\right] = P(f, n, j)\, R_j^{\text{obs}}(f),$$

où $R_j^{\text{obs}}(f)$ est la matrice de covariance spatiale de l'observation $\boldsymbol{s}_d(\cdot, \cdot, \cdot, j)$ pour l'indice de fréquence f. On dispose de l'observation de $\boldsymbol{s}_d$, mais à la fois $P(\cdot, \cdot, j)$ et R_j^{obs} sont inconnus. Si on suppose l'une de ces grandeurs connues, il est cependant facile d'estimer l'autre aux moindres carrés. Il est donc possible de mettre au point l'algorithme itératif 11.1, simple à implémenter et qui permet de déterminer à la fois $P(f, n, j)$ et $R_j^{\text{obs}}(f)$. Il faut noter qu'une régularisation peut être nécessaire dans l'implémentation de cet algorithme en cas de faible valeurs pour $\hat{P}$ lors de la mise à jour de $\widehat{R}_j^{\text{obs}}$, ce qui arrive en cas de silence des sources.

Algorithme 11.1 Estimation $\widehat{P}(f, n, j)$ de la DSP et des matrices de covariance spatiales $\widehat{R}_j^{\text{obs}}(f)$ d'un processus à partir de l'observation de son image $\boldsymbol{s}_d(f, n, \cdot, j)$ produite par mixage diffus.

Entrées :
– TFCT $\boldsymbol{s}_d(f, n, \cdot, j)$ d'une source diffuse j, de dimension $F \times N \times I$.

Initialisation :
– Définir $\widehat{P}(f, n, j) = \frac{1}{I} \sum_{i=1}^{I} |\boldsymbol{s}_d(f, n, i, j)|^2$

Répéter jusqu'à convergence :
– pour chaque f, $\widehat{R}_j^{\text{obs}}(f) \leftarrow \frac{1}{N} \sum_{n=1}^{N} \frac{\boldsymbol{s}_d(f,n,\cdot,j)\boldsymbol{s}_d(f,n,\cdot,j)^H}{\widehat{P}(f,n,j)}$
– pour chaque (f, n), $\widehat{P}(f, n, j) \leftarrow \frac{1}{I} \boldsymbol{s}_d(f, n, \cdot, j)^H\, \widehat{R}_j^{\text{obs}}(f)^{-1}\, \boldsymbol{s}_d(f, n, \cdot, j)$

Sortie :
– $\widehat{P}(\cdot, \cdot, j)$ et $\widehat{R}_j^{\text{obs}}$

11.3 Paramètres de sources

Comme on l'a vu en section 4.1 page 57, les paramètres de sources θ sont appris par approximation des spectrogrammes v des sources. Ici, je propose de les apprendre en utilisant les mêmes techniques, à la différence que les spectrogrammes v des observations sont remplacés par les estimées $\widehat{P}$ des DSP des sources, obtenues à l'étape précédente.

Dans le cas où on n'observe que des sources ponctuelles ($S_d = 0$), on retombe bien sur la stratégie optimale vue en section 10.2.1, puisque les estimées des DSP coïncident alors avec les spectrogrammes. Dans le cas contraire, l'intérêt de l'approche est de permettre l'utilisation des mêmes algorithmes pour l'apprentissage des modèles de sources.

11.4 Paramètres de mixage

Lorsqu'on dispose des paramètres θ de sources et donc des modèles $\mathcal{P}(\cdot \mid \theta)$ de DSP correspondants, il s'agit de déterminer les paramètres de mixage $\{A_j(f)\}_{f,j \in \mathcal{M}_c}$ et $\{R_j(f)\}_{f,j \in \mathcal{M}_d}$ permettant au mieux de rendre compte des mélanges $\boldsymbol{x}$ observés. Pour simplifier les notations, je désignerai par θ_M les paramètres de mixage :

$$\theta_M = \left\{\{A_j(f)\}_{f,j \in \mathcal{M}_c}, \{R_j(f)\}_{f,j \in \mathcal{M}_d}\right\}$$

La stratégie adoptée dans ce but repose sur l'application d'un algorithme de type Espérance-Maximisation (EM, [42]), fortement inspiré des travaux [155, 50], qui reposent eux-mêmes sur les

travaux précurseurs menés par FEDER et WEINSTEIN dans [68]. Dans cet algorithme, les variables $\boldsymbol{x}$ observées sont complétées par un lot de variables latentes non observées $\boldsymbol{y}$ de manière à former l'ensemble $\{\boldsymbol{x}, \boldsymbol{y}\}$ des *données complètes*[3]. Dans notre cas, ces variables latentes sont simplement constituées des images des sources. Dans la mesure où le mélange s'obtient de manière déterministe comme la somme des images $\boldsymbol{y}$, les données complètes considérées seront simplement $\boldsymbol{y}$ et il apparaît que notre problème devient exactement équivalent à celui considéré dans [50] (section 3.3), à la différence que les DSP $\mathcal{P}(\cdot \mid \theta)$ des sources ne sont pas à estimer puisqu'on les suppose déjà estimées et fixées.

Dans ces conditions, je propose d'appliquer exactement le même algorithme que celui proposé par DUONG *et al.* en introduisant cependant une petite modification, qui consiste à estimer dans un premier temps toutes les matrices de covariance spatiale $R_j(f)$ sous la forme de matrice à rang plein I lors de l'étape de Maximisation, et de les décomposer sous la forme $A_j(f) A_j(f)^H$ lorsque cela est nécessaire (pour $j \in \mathcal{M}_C$) seulement dans un deuxième temps. Cela permet de s'affranchir de la nécessite d'introduire un bruit additif dans le modèle pour pallier à d'éventuelles instabilités numériques. L'algorithme 11.2 page suivante résultant permet d'estimer les paramètres de mixage.

11.5 Filtres de formation de voie (sources ponctuelles mixées de manière diffuse)

Comme je l'ai expliqué en section 11.1.2, les filtres de formation de voie

$$\{U_j(f)\}_{f, j \in \mathcal{S}_p \cap \mathcal{M}_d \cap \mathcal{Z}_s}$$

ne sont pas inclus dans les paramètres qu'une approche générative peut optimiser. Je propose ainsi de les estimer indépendamment, une fois tous les autres paramètres de Θ déterminés. Pour ce faire, il suffit de considérer la principale expression 9.2.4 page 124 où ils interviennent comme la covariance entre les réalisations ponctuelles observées des sources de $\mathcal{S}_p \cap \mathcal{M}_d \cap \mathcal{Z}_s$ et les mélanges :

$$\mathbb{E}\left[\boldsymbol{s}_p(f,n,j)\, \boldsymbol{x}(f,n,\cdot)^H\right] = U_j(f)^\top R_j(f) P(f,n,j).$$

on peut remarquer que compte tenu de l'indépendance des sources, cette expression se confond avec :

$$\mathbb{E}\left[\boldsymbol{s}_p(f,n,j)\, \boldsymbol{y}(f,n,\cdot)^H\right] = U_j(f)^\top \mathbb{E}\left[\boldsymbol{y}(f,n,\cdot,j)\, \boldsymbol{y}(f,n,\cdot,j)^H\right],$$

ce qui indique que $U_j(f)$ n'est rien d'autre que le vecteur permettant d'esimer $\boldsymbol{s}_p(f,n,j)$ à partir de l'image $\boldsymbol{y}(f,n,\cdot,j)$ aux moindres carrés. Si on suppose fixée l'estimée $\hat{\boldsymbol{y}}(f,n,\cdot,j)$ de l'image, obtenue par filtrage de Wiener des mélanges, on peut estimer $U_j(f)$:

$$\forall j \in \mathcal{S}_p \cap \mathcal{M}_d \cap \mathcal{Z}_s, \forall f,\ \widehat{U_j}(f)^\top = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{s}_p(f,n,j)\, \hat{\boldsymbol{y}}(f,n,\cdot)^H \left(\boldsymbol{y}(f,n,\cdot,j)\, \boldsymbol{y}(f,n,\cdot,j)^H\right)^{-1}, \tag{11.5.1}$$

où je rappelle que $\cdot^*$ désigne la conjugaison complexe et $\cdot^H$ la conjugaison Hermitienne (conjuguée transposée). Si on désire plutôt transmettre ces filtres de formation de voie sous la forme de réponses impulsionnelles, on peut appliquer une transformée de Fourier inverse aux filtres estimés par 11.5.1.

11.6 Conclusion

Au cours des différentes sections de ce chapitre, j'ai proposé une heuristique pour l'apprentissage dans toute sa généralité de l'information annexe du système de séparation informée paramétrique présenté au chapitre 9. La quantification et l'encodage des paramètres ainsi appris peut se faire en utilisant les mêmes techniques que celles présentées en section 10.3.

3. Je rappelle que si les sources (ponctuelles ou diffuses) sont observées, leurs images $\boldsymbol{y}$ dans le mélange ne le sont pas.

Algorithme 11.2 Estimation des paramètres de mixage $\theta_M = \left\{ \{A_j(f)\}_{f,j\in\mathcal{M}_c}, \{R_j(f)\}_{f,j\in\mathcal{M}_d} \right\}$ à partir du modèle θ des DSP des sources et des mélanges $\boldsymbol{x}$.

Entrées :
- TFCT $\boldsymbol{x}(f,n,\cdot)$ des mélanges
- Paramètres θ des DSP des sources (alternativement, leur version quantifiée $\bar{\theta}$)
- Ensembles $\mathcal{M}_c$ et $\mathcal{M}_d$ des sources mixées de manière convolutive ou diffuse, respectivement.

Initialisation :
- Définir tous les $R_j(f)$ comme une matrice diagonale de dimension $I \times I$

Répéter jusqu'à convergence :
- Étape d'estimation : pour chaque (f,n,j) :
  1. $K(\boldsymbol{x}(f,n,\cdot), \boldsymbol{x}(f,n,\cdot)) = \sum_{j=1}^{J} \mathcal{P}(f,n,j \mid \theta) R_j(f)$
  2. Calcul du gain "de Wiener"
  $$G_j = \mathcal{P}(f,n,j \mid \theta) R_j(f) K(\boldsymbol{x}(f,n,\cdot), \boldsymbol{x}(f,n,\cdot))^{-1}$$
  de dimension $I \times I$
  3. Estimée des images :
  $$\widehat{\boldsymbol{y}}(f,n,\cdot,j) = G_j \boldsymbol{x}(f,n,\cdot)$$
  4. Estimée des covariances des images :
  $$\widehat{K}(\boldsymbol{y}(f,n,\cdot,j), \boldsymbol{y}(f,n,\cdot,j)) = \widehat{\boldsymbol{y}}(f,n,\cdot,j)\widehat{\boldsymbol{y}}(f,n,\cdot,j)^H + (I_I - G_j)\mathcal{P}(f,n,j \mid \theta) R_j(f),$$
  où je rappelle que I_I est la matrice identité de dimension $I \times I$.
- Étape de maximisation : pour chaque f et chaque j :
  1. $R_j(f) \leftarrow \frac{1}{N} \sum_{n=1}^{N} \frac{\widehat{K}(\boldsymbol{y}(f,n,\cdot,j), \boldsymbol{y}(f,n,\cdot,j))}{P(f,n,j|\theta)}$
  2. Si $j \in \mathcal{M}_c$:
     a) Calculer la décomposition en valeurs propres de $R_j(f)$.
     b) Définir $A_j(f)$ comme le vecteur propre de $R_j(f)$ associé à la plus grande valeur propre λ_{max}, multiplié par $\sqrt{\lambda_{max}}$.
     c) $R_j(f) \leftarrow A_j(f) A_j(f)^H$

Sortie :
- $\{A_j(f)\}_{f,j\in\mathcal{M}_c}$ et $\{R_j(f)\}_{f,j\in\mathcal{M}_d}$

Ce chapitre présente une technique complète d'estimation de l'information annexe, dans le cas général envisagé au chapitre 9. Il est probable que des travaux ultérieurs viendront améliorer la technique proposée, basée sur des heuristiques.

Contrairement au cas plus simple envisagé au chapitre précédent et dont j'ai considéré des cas particuliers dans [130, 137], l'heuristique que je viens de présenter ne bénéficie pas de garanties d'optimalité, puisqu'elle repose sur une série d'approximations qui mériteraient d'être étudiées de plus près et au cas échéant remplacées au profit de techniques plus adéquates. En revanche, elle offre une solution assez simple au problème du calcul de l'information annexe au niveau de l'encodeur dans un nombre très considérable de configurations, qui englobe l'ensemble des cas de figure abordés dans la littérature mais qui inclut aussi de multiples variantes inédites.

De manière à clôturer ce chapitre sur l'encodage de l'information annexe dans le cas général, je résume dans l'algorithme 11.3 l'ensemble des opérations que je viens de décrire et qui sont à implémenter au niveau du codeur. Les opérations effectuées au décodeur, présentées dans l'algorithme 9.1 page 125, restent inchangées. Cet algorithme sera mis en œuvre dans l'évaluation présentée au chapitre 16.

Algorithme 11.3 Encodeur paramétrique pour la séparation informée selon le modèle gaussien, cas général.

Entrées :
- I signaux régulièrement échantillonnés de mélange $\tilde{\boldsymbol{x}}$
- S_p signaux régulièrement échantillonnés : sources ponctuelles $\tilde{\boldsymbol{s}}_p$
- S_d groupes de I signaux régulièrement échantillonnés : sources diffuses $\tilde{\boldsymbol{s}}_d$
- Paramètres ρ et L_0 de tramage
- Famille paramétrique $\mathcal{P}$ utilisée pour les DSP des sources
- Pas de quantification Δ_θ à utiliser pour le modèle de sources
- Paramètres de mixage s'ils sont connus
- Ensemble $\mathcal{Z}_s$, $\mathcal{Z}_y$ et $\mathcal{Z}_\varnothing$ des signaux désirés au décodeur

Initialisation
- Construire les TFCT $\boldsymbol{x}$ et $\boldsymbol{s}$ des mélanges et des sources observées

DSP des sources
- Pour les sources $\boldsymbol{s}_p$, estimer leurs DSP par leurs spectrogrammes
- Pour les sources $\boldsymbol{s}_d$, utiliser l'algorithme 11.1

Modèle de sources θ
- Si $\mathcal{P}$ est le modèle NTF
  - Estimer le modèle de sources $\theta = \{W, H, Q\}$ en utilisant l'algorithme 4.1 où on remplace v par les DSP estimées à l'étape précédente
  - Quantifier uniformément $\log W$, $\log H$ et $\log Q$ en utilisant les pas de quantification 10.3.8
  - Encoder les paramètres en utilisant un codage de Huffman ou un MMG comme décrit en section 10.3.3
- Si $\mathcal{P}$ est le modèle CI
  - Compresser les log-DSP des sources en utilisant un algorithme de codage d'image (idéalement, utiliser une quantification uniforme au lieu de la quantification perceptuelle intégrée dans les codeurs standards)

Modèle de mélange
- Utiliser l'algorithme 11.2 pour estimer $\{R_j(f)\}_{f,j\in\mathcal{M}_d}$ et $\{A_j(f)\}_{f,j\in\mathcal{M}_c}$
- Les quantifier de manière uniforme sur 32 bits et utiliser un encodage de Huffman

Filtres de formation de voie
- Si cela est nécessaire ($j \in \mathcal{Z}_s \cap \mathcal{S}_p \cap \mathcal{M}_d$), estimer ces filtres en utilisant 11.5.1
- Les quantifier de manière uniforme sur 32 bits et utiliser un encodage de Huffman

Sortie
- Retourner le flux binaire correspondant à

$$\bar{\Theta} = \left\{\bar{\theta},\, \left\{\bar{R}_j(f)\right\}_{f,j\in\mathcal{M}_d},\, \left\{\bar{A}_j(f)\right\}_{f,j\in\mathcal{M}_c},\, \left\{\bar{U}_j(f)\right\}_{j\in\mathcal{Z}_s\cap\mathcal{S}_p\cap\mathcal{M}_d}\right\}$$

Chapitre 12

Évaluation

Dans ce chapitre, je présente les résultats des différentes campagnes d'évaluation que j'ai menées au cours de mon travail de doctorat sur le système proposé de séparation informée paramétrique. Ce travail d'évaluation s'est opéré sur les trois années durant lesquelles je me suis intéressé à cette problématique et a été effectué dans le cadre de plusieurs publications [130, 137, 134]. Ce sont les résultats publiés dans [134] que je vais détailler tout particulièrement ici. Cette évaluation est le fruit de la collaboration de l'ensemble des partenaires académiques du projet DReaM et constitue un état des lieux assez complet sur les performances et les débits que peuvent atteindre les systèmes paramétriques actuels. Il s'agit à ma connaissance de l'unique évaluation commune et complète de plusieurs systèmes de séparation informée sur la même base de données et avec les mêmes métriques. Je tiens à remercier ici tous ses participants pour leur enthousiaste collaboration.

12.1 Métriques

12.1.1 Qualité de séparation : principes généraux

L'évaluation de la séparation de sources est un problème non trivial qui a fait l'objet de plusieurs travaux. En premier lieu, une manière naturelle d'évaluer les résultats d'une séparation est de faire appel à une campagne d'évaluation perceptive, c'est-à-dire de demander à de nombreux sujets dans une situation d'écoute contrôlée de quantifier leur opinion sur la qualité des sons obtenus après séparation. En effet, seules de telles études permettent d'évaluer rigoureusement le véritable objectif de la séparation, qui est de permettre de récupérer des sources séparées qui soient agréables à entendre. Quantifier les critères selon lesquels un humain juge qu'un son est de bonne qualité est une tâche très difficile et il est toujours mieux de faire appel à une campagne d'évaluation perceptive lorsque cela est possible. C'est par exemple une étape obligée dans la définition de standards de compression audio [146, 147].

Si l'idéal serait d'avoir recours à une évaluation perceptive de la séparation, de telles évaluations sont difficiles à réaliser. C'est pour cette raison que des métriques objectives comme celles de BSSEval ou PEASS ont été proposées.

Malheureusement, les évaluations perceptives souffrent d'un inconvénient majeur : elles sont d'une grande difficulté de mise en œuvre. Pour qu'elles soient faites dans de bonnes conditions, il est en effet nécessaire de faire appel à de nombreux sujets, d'évaluer les scores sur une large quantité de données et enfin de procéder à un encadrement strict des conditions d'écoute. De telles procédures sont souvent coûteuses à mettre en place et requièrent énormément d'investissement pour être menées à bien. Je n'ai malheureusement pas eu le temps de réaliser une telle campagne d'évaluation durant ma thèse.

Pour pallier à la difficulté de faire appel à une évaluation perceptive pour l'évaluation de la séparation, plusieurs chercheurs se sont concentrés sur l'établissement de métriques permettant une évaluation objective [212, 59]. De telles métriques ont le grand avantage de simplement nécessiter l'application d'algorithmes qui aboutissent à la production d'un score. Elles sont donc très faciles à

utiliser dans un contexte d'évaluation, d'autant plus que des implémentations librement utilisables en sont disponibles. Le principe d'une évaluation objective de la séparation de sources est le suivant.

En premier lieu, une évaluation objective d'un algorithme de séparation nécessite la disponibilité de plusieurs morceaux musicaux dont on dispose de l'ensemble des pistes séparées. Pour un morceau donné, c'est en effet en comparant les pistes séparées avec les originales qu'on peut produire un score estimant la qualité de la séparation. De plus, la configuration informée requiert la connaissance des sources au codeur pour pouvoir être appliquée. La disponibilité de tels corpus est encore très restreinte et j'ai eu la chance de bénéficier en primeur de la base de données QUASI [1], qui contient les pistes séparées d'une quinzaine de morceaux de musique ainsi que leurs mélanges. Je reviendrai sur ce point en section 12.3.

Dans le cadre de mon travail sur la séparation informée, j'ai utilisé deux métriques principales. La première, le SDR (*Signal to Distortion Ratio*) provient de la bibliothèque BSSEval [212], très utilisée dans la communauté de la séparation de sources audio. La deuxième, le PSM (*Perceptual Similarity Measure*, [107]) provient de la communauté du codage audio.

12.1.2 BSSEval

La bibliothèque BSSEval (*Blind Source Separation Evaluation*) permet d'évaluer les performances de la séparation de sources audio. Si on dispose d'un lot de sources originales et d'un lot de sources estimées, elle renvoie pour chaque source un ensemble de trois métriques permettant d'évaluer la qualité de la séparation, qui s'expriment toutes trois en décibels (dB) :

- **Le SDR** (*Signal to Distortion Ratio*) peut se comprendre comme un rapport moyen entre la source estimée et l'erreur d'estimation. En ce sens, elle se rapproche d'un simple rapport signal à bruit. Cependant, ce n'est pas la source estimée telle que renvoyée par l'algorithme de séparation qui est utilisée pour calculer la métrique mais plutôt sa version filtrée qui permet au mieux de la rapprocher de la source originale [2]. L'idée derrière cette technique d'estimation est de ne pas pénaliser une séparation qui serait parfaite *à un filtre près*.
- **Le SIR** (*Signal to Interference Ratio*) permet de quantifier la quantité d'interférences entre les sources dans les estimées, c'est-à-dire à quel point on entendra les autres sources quand on écoutera une des pistes séparées.
- **Le SAR** (*Signal to Artefacts Ratio*) cherche à quantifier la quantité d'*artéfacts* présents dans les pistes séparées, comme le bruit musical, introduit par la mise à zéro de nombreux points TF dans les estimées, qui conduit à de brusques changements de phases et d'amplitudes des signaux, désagréables à l'oreille.

Cette bibliothèque d'évaluation fait aujourd'hui l'objet d'un consensus dans le domaine de la séparation de sources audio, pour plusieurs raisons. Tout d'abord, il a été montré [125] que parmi l'ensemble des métriques disponibles pour l'évaluation de la séparation de sources, ce sont celles de BSSEval qui sont le mieux corrélées avec les résultats d'évaluations perceptives. Bien que ce résultat doive être tempéré par les récentes recherches accomplies dans [59] qui tendent à montrer que de nouvelles métriques puissent être plus avantageuses sur ce point, il reste clair que leur validité est généralement acceptée. Une deuxième raison du succès de BSSEval est qu'il s'agit d'une technique d'évaluation relativement rapide, quoique loin d'être instantanée. Pour un extrait d'environ 20 secondes composé de 5 sources, il faut environ 5 secondes pour mener à bien l'évaluation. Enfin, un avantage déterminant pour la large diffusion de ces métriques a été la mise à disposition pour tous par Emmanuel Vincent de scripts Matlab implémentant ces méthodes [3].

12.1.3 PEMO-Q

Un des principaux inconvénients des métriques de BSSEval dans le contexte qui nous occupe est qu'elles ont été mises au point pour évaluer la séparation *aveugle* et non pas la séparation infor-

1. `http://www.tsi.telecom-paristech.fr/aao/en/2012/03/12/quasi/`
2. au sens des moindres carrés
3. `http://bass-db.gforge.inria.fr/bss_eval/`

mée. En effet, un problème qui est apparu lorsque nous nous sommes intéressés à la comparaison entre elles des différentes techniques de séparation informée est qu'elles obtenaient toutes d'excellents scores avec BSSEval, qui n'étaient vraisemblablement pas corrélés avec la qualité perceptive ressentie par les auditeurs lors de tests informels.

Les techniques de séparation informée permettent de minimiser des distorsions quadratique du type de celles considérées par BSSEval, sans nécessairement conduire à des gains perceptifs. Pour cette raison, j'ai utilisé PSM, qui est une mesure de qualité issue de la communauté du codage.

La principale raison de cette faiblesse est simple : l'approche informée permet de produire des pistes séparées qui minimisent la distorsion quadratique entre les sources originales et les estimées. Ce faisant, elles maximisent les métriques de BSSEval, sans nécessairement produire des pistes séparées d'une qualité parfaite. On touche ainsi aux limites d'une évaluation basée sur des critères quadratiques, bien connues dans le domaine du codage audio.

Pour cette raison, j'ai décidé dans [134, 136, 161] de ne plus me restreindre aux métriques de BSSEval, mais de leur adjoindre une métrique classique dans l'évaluation de la qualité des codeurs audio : PSM (Perceptual Similarity Measure), issue de la bibliothèque PEMO-Q [107]. Le mode d'utilisation de cette métrique est le même que pour BSSEval : à partir d'une source originale et d'une source estimée, un algorithme retourne un score représentatif de la qualité de l'estimation. Cette fois, le score est compris entre 0 (médiocre) et 1 (identique).

Une métrique telle que PSM, à la différence de celles de BSSEval, a été spécifiquement mise au point pour produire des scores qui imitent ceux donnés par des humains *lorsque les attentes en terme de qualité sont très élevées*, ce qui est le cas dans un contexte de codage. Dans la mesure où les pistes séparées obtenues par les techniques informées sont toutes de très bonne qualité, l'utilisation d'une telle métrique se justifie. Dans mes évaluations, j'ai utilisé l'implémentation de PSM qu'on peut trouver dans [59].

12.1.4 Débit total et courbes de débit-qualité

En dehors de l'évaluation de la qualité de la séparation, il est primordial pour une technique de séparation informée d'également prendre en compte le débit nécessaire à la transmission de l'information annexe Θ du codeur vers le décodeur, que j'appellerai "débit total" par la suite. Dans tout cet exposé, le débit total se définit comme le nombre total de bits requis pour transférer l'information annexe Θ, divisé par la durée en secondes des mélanges. Il sera fréquent d'exprimer ce débit en kilobits par seconde et par source, de manière à pouvoir le comparer aux grandeurs habituelles pour des codeurs audio standards [4].

L'importance de prendre en compte le débit dans l'évaluation de la séparation informée apparaît lorsqu'on se souvient qu'elle s'apparente à un type particulier de codage audio (voir section 9.4 page 126). En conséquence, ce n'est qu'à la lumière de l'évolution de la qualité de restitution des sources séparées en fonction du débit qu'on peut juger de la performance d'une méthode de séparation informée. Si le critère de qualité retenu dans ce but est l'erreur quadratique moyenne, cette relation débit-qualité est appelée la courbe *débit-distorsion* et joue un rôle fondamental dans la théorie du codage de source [89], sur laquelle je reviendrai en section 13.1 page 167. Dans notre cas, je considèrerai plutôt les courbes débit-SDR et débit-PSM.

12.2 Normalisation sur un corpus

Dans l'exposé des métriques objectives retenues pour l'évaluation en section 12.1 ci-dessus, on a pu remarquer que chaque source séparée de chaque mélange produit un score SDR et un score PSM. Face à l'abondance de ces scores dans le cas d'une évaluation de grande ampleur, un problème de synthèse de ces résultats se pose de manière aigüe.

Sur un même mélange, la stratégie courante consiste à moyenner les scores obtenus pour les différentes sources. Par contre, moyenner des scores de qualité tels que SDR ou PSM obtenus sur

4. On peut noter que je n'exprimerai pas ici le débit total en kbps/source/mélange, comme je l'ai fait dans [130, 137]. Cela ne se justifiait que par la manière particulière de transmettre l'information annexe que j'ai considérée dans ces études.

différents mélanges est une façon très discutable de résumer la qualité moyenne de séparation. En effet, l'expérience montre que tous les mélanges ne sont pas d'une égale complexité à séparer. Si un mélange est composé de peu de sources ou de sources dont le support fréquentiel est disjoint, alors il sera facile de procéder à une séparation et les scores obtenus seront très bons. Au contraire, si un mélange est constitué de nombreuses sources ou de sources qui se superposent fortement sur le plan temps-fréquence, alors la séparation sera plus difficile. Si on attribue le même poids à tous les scores de séparation obtenus indépendamment de cette difficulté de séparation, ce n'est somme toute pas la technique de séparation qu'on évalue, mais la complexité de la tâche.

Face à ce problème, plusieurs stratégies sont possibles. Si le nombre de mélanges considérés est faible, il est toujours possible de représenter les résultats pour chacun indépendamment de ceux des autres. C'est ce qui est souvent fait dans les campagnes d'évaluations [208, 209] et c'est ce que j'ai fait moi-même dans [130]. Si cette représentation a l'avantage de la simplicité, elle conduit souvent à des tableaux de chiffres dont l'interprétation est difficile.

De manière à pouvoir moyenner les performances de séparation obtenues sur plusieurs morceaux, j'ai introduit dans [137] l'idée de ne plus considérer les scores SDR et PSM absolus, mais de plutôt considérer leur écart δ_{SDR} et δ_{PSM} avec ceux obtenus lorsque l'information annexe est celle de la configuration oracle 9.1.7 page 122. Il est remarquable en effet que les performances obtenues par la configuration oracle donnent une indication très fidèle de la complexité du problème de séparation, en même temps qu'une borne supérieure des performances atteignables par une séparation paramétrique. De plus, j'ai remarqué que la variabilité entre les scores δ obtenus pour différents mélanges est bien plus faible que celle des scores absolus, rendant possible la comparaison entre différents mélanges.

Le δ_{SDR} et δ_{PSM} se définissent comme l'écart entre les scores de la séparation évaluée et ceux de la configuration oracle. Ces métriques compensent les larges variations des scores absolus et permettent de moyenner les performances entre extraits.

Pour chaque fichier considéré et chaque configuration de qualité, j'ai ainsi obtenu un score δ_{SDR} et δ_{PSM}, conduisant à l'obtention d'un nuage de points comme celui représenté en figure 12.1. De manière à pouvoir représenter cet ensemble de résultats sous une forme synthétique, j'ai utilisé une technique de lissage non-paramétrique [5], qui m'a permis de produire une seule courbe débit-δ_{SDR} et débit-δ_{PSM} pour chaque technique, qui résume d'une manière assez fiable l'évolution de la qualité de séparation obtenue par une technique par rapport au score de la configuration oracle gaussienne, en fonction du débit total.

12.3 Données

12.3.1 Nature des données

La configuration informée requiert la connaissance des sources au codeur pour pouvoir être appliquée. De la même manière, une évaluation objective nécessite elle aussi ces sources.

S'il est clair que les sources sont connues au moment de l'étape de production en studio d'un morceau de musique et qu'elles sont conservées ultérieurement, les usages font qu'elles sont jalousement gardées par les professionnels et les ayant-droits qui les considèrent souvent comme un trésor qu'il ne faut pas divulguer. En conséquence, il est difficile de réunir des grands corpus qui rassemblent des morceaux mixés accompagnés des pistes séparées qui les composent. La principale campagne internationale d'évaluation des performances de séparation aveugle de sources (*SIgnal Separation Evaluation Campaign*, SISEC), organisée depuis 2007 [6] par Emmanuel Vincent [208, 209] a l'avantage de permettre une comparaison internationale et objective

5. La technique utilisée est LOESS [36]. Elle estime localement les coefficients d'une droite aux moindres carrés pondérés. Elle fut mise au point spécialement dans le but de produire des courbes synthétiques à partir de nuages de points.

6. J'ai participé à SISEC en 2011 avec Zafar Rafii [177]. Nos performances ont été très honorables. Il est regrettable cependant qu'aucune campagne d'évaluation n'existe encore qui porte sur la séparation de morceaux entiers.

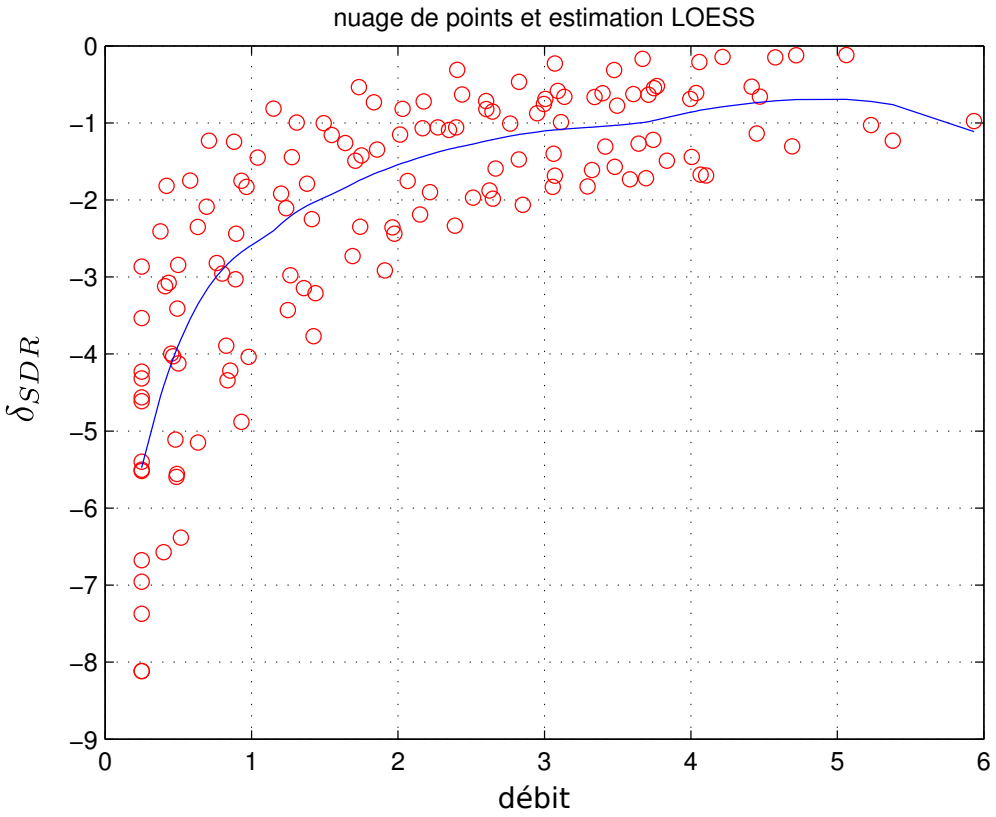

FIGURE 12.1: Exemple d'un nuage de points lissé en utilisant LOESS [36].

des différents algorithmes de séparation aveugle mais elle ne fait pour l'instant intervenir que 4 extraits de trente secondes dans la tâche portant sur la séparation de musique.

La base de données QUASI que j'ai utilisée pour mon évaluation a avant tout été rendue possible par la gracieuse mise à disposition sous licence libre de leurs pistes séparées par des artistes tels que ANOTHER DREAMER, SHANNON HURLEY, ULTIMATE NZ TOUR, JIM BIG EGO, PHOENIX, GLEN PHILLIPS, FARKA TOURÉ ou NINE INCH NAILS. Qu'ils soient ici remerciés pour ce geste très apprécié.

Il est possible de trouver sur Internet des pistes séparées qui circulent entre passionnés et qui proviennent souvent de jeux musicaux tels que GUITAR HERO ou ROCK BAND. Cependant, il n'est pas possible de les utiliser officiellement, ne serait-ce qu'à des fins de recherche, parce que leurs licences d'exploitation de le permettent pas. Fort heureusement, un nouveau type de licences se développe depuis quelques années qui permet l'exploitation et la diffusion du contenu musical à titre gracieux. Il existe ainsi une communauté grandissante d'artistes qui placent leurs œuvres sous une licence CREATIVE COMMONS ou LICENCE ART LIBRE, qu'on peut comprendre pour faire court comme l'équivalent des licences GPL ou LPGL en informatique[7]. Ainsi, il a été possible de réunir une quinzaine de morceaux avec toutes leurs pistes séparées, exploitables à des fins scientifiques. La base de données QUASI[8] est composée d'une quinzaine de morceaux complets, dont on dispose de toutes les pistes séparées, échantillon-

7. J'invite le lecteur intéressé à se renseigner sur les subtilités afférentes à ces types de licences et à l'alternative qu'elles représentent au système actuel de propriété intellectuelle des créations artistiques. Des bons pointeurs pour commencer sont les sites officiels correspondants `www.creativecommons.org` et `http://artlibre.org`. Je me suis beaucoup investi personnellement dans la promotion de ces licences au court de ces dix dernières années et je ne peux que constater leur importance dans un contexte scientifique où elles sont les seules aujourd'hui à permettre la disponibilité de données d'évaluation pour la séparation de sources audio.

8. `http://www.tsi.telecom-paristech.fr/aao/en/2012/03/12/quasi/`

nées à 44.1kHz ou 48kHz. Il s'agit à l'heure actuelle de la base de données la plus complète dans le domaine. Elle contient des morceaux de genres différents, qui vont du reggae à la bossa nova en passant par le rock, l'electro-pop et l'industriel.

12.3.2 Mélanges

Hormis les sources, ou *pistes séparées*, il est nécessaire de disposer des mélanges à partir desquels se fera la séparation. À partir des pistes séparées, deux approches principales sont possibles :

- On peut réaliser un mixage *de laboratoire*, qui consiste à produire les images des sources en appliquant des filtres de mélange connus. C'est en général la stratégie adoptée dans les études de séparation de sources audio [168, 166, 169, 170, 167, 165, 78, 101, 209]. Bien que ces mixages ne correspondent pas aux usages des professionnels de la musique, il faut noter qu'ils peuvent être de très bonne qualité. Ils se caractérisent en outre par le fait que les mélanges ne subissent pas de post-production.
- On peut réaliser un mixage *réaliste* des sources, au sens d'un mixage qui correspond aux usages des professionnels. Dans ce cas, les mélanges sont produits en utilisant des logiciels professionnels (DAW, Digital Audio Workstation) et sont d'une qualité perceptive supérieure. Par contre, ils ne respectent pas exactement les hypothèses faites par les méthodes de séparation et leurs paramètres sont généralement inconnus voire difficiles à modéliser. La base de données QUASI contient de nombreux exemples de mixages professionnels. Rares sont pour l'instant les études qui mettent en œuvre de tels mélanges dans les évaluations [130, 137, 201].

Si la deuxième option a l'attrait du réalisme par rapport aux applications visées, elle a l'inconvénient de ne pas être prise en charge par l'ensemble des techniques existantes. J'ai moi-même considéré le cas de mixages réalistes dans plusieurs évaluations publiées [130, 137] et je renvoie le lecteur à ces publications pour plus de détails.

De manière à permettre une comparaison de toutes les techniques de séparation informée disponibles à l'heure actuelle, il a été nécessaire de se restreindre aux configurations de mixage que toutes permettent de traiter. En conséquence, il a été choisi de se concentrer exclusivement sur deux types de mixages de laboratoire. Le premier est un mixage linéaire instantané, tel que décrit au chapitre 5. Le deuxième est un mixage convolutif, décrit au chapitre 6. Dans ce deuxième cas, les filtres de mélange sont connus et d'une réponse impulsionnelle de longueur $H = 200$. Ils proviennent de la base de données CIPIC [4] de filtres de spatialisation par fonction de transfert de tête (*Head Related Transfer Function*).

Les mixages considérés sont linéaires instantanés et convolutifs. Je remercie Sylvain Marchand d'avoir rendu possible cette opération de mixage de laboratoire, en fournissant à la fois la base de données des filtres de mélange et l'implémentation en Matlab permettant de l'exploiter.

L'évaluation a ainsi porté sur un extrait de 30 secondes de chacun des morceaux de la base de données, pour lequel deux mixages sont disponibles : le premier est linéaire instantané tandis que le deuxième est convolutif avec des filtres de mélange de longueur $H = 200$. Dans la suite, **les filtres de mélange sont supposés connus** par les techniques de séparation, puisque cette connaissance est nécessaire à certaines d'entre elles.

12.4 Configurations testées

L'ensemble des configurations permises par le formalisme que j'ai proposé au chapitre 9 est large. Je n'ai malheureusement pas eu le temps de pouvoir toutes les tester et d'ainsi compléter totalement l'évaluation du système proposé. Plusieurs raisons viennent expliquer ce constat. Tout d'abord, cette formulation générale est venue assez tard au cours de mon travail, lorsqu'il est devenu possible d'avoir sur le problème un recul suffisant pour formaliser ses différents cas de figure. Ensuite, chaque campagne d'évaluation est l'objet de plusieurs semaines de calculs en continu, principalement liés à l'obtention des scores SDR et PSM.

L'évaluation a porté sur 14 extraits de 30 secondes, mixés de deux manières différentes et avec 10 configurations différentes de qualité. Les scores correspondants ont été calculés pour 6 techniques de séparation informée paramétrique. En tout, j'ai calculé pour cette campagne les scores de séparation de plus de 8000 sources.

Ainsi, je ne présenterai ici en détail que des résultats de séparation informée paramétrique pour le cas $S_d = M_d = 0$ et l'encodeur évalué est donc celui présenté au chapitre 10 [9]. De plus, certaines des méthodes sont limitées à la récupération des sources ponctuelles. Pour pouvoir comparer entre elles les performances, on ne considèrera donc que le cas $Z_y = 0$. On peut noter toutefois que c'est le cas complémentaire $Z_s = 0$ où ce sont les images des sources qu'on souhaite récupérer qui m'a occupé pour les évaluations qu'on trouvera dans [130, 137].

Malgré ce bémol sur le nombre de configurations permises par le modèle qui n'ont pas encore fait l'objet d'une étude complète complète, il faut noter que la portée de la présente évaluation, résumée dans [134], est inédite. Chacune des 6 techniques comparées [137, 87, 200, 167, 223], auxquelles s'ajoute la configuration oracle, a été testée sur les 14 extraits de la base pour deux types de mixage et 10 configurations différentes de qualité pour l'information annexe. En tout, cela a mené au calcul d'environ 8000 scores pour chaque métrique, nécessitant plusieurs semaines de calcul et de mise en forme des résultats.

Sur les 6 techniques évaluées, 2 sont des variantes de l'approche que je propose pour la séparation informée. La première correspond à l'utilisation du modèle de sources CI et la deuxième à l'utilisation du modèle NTF. Pour la première, l'algorithme de compression d'images utilisé pour la quantification des log-spectrogrammes est JPEG [216] et les différents débits testés correspondent à 10 paramètres de qualité JPEG différents, de 2 à 70. Pour la deuxième, les dix paramètres de qualité correspondent à un nombre grandissant de composantes K dans le modèle NTF. La quantification des paramètres a été effectuée dans le domaine linéaire et l'encodage par l'algorithme de Huffman, comme cela a été fait dans [137] et non pas comme cela est suggéré en section 10.3.3. Il faut noter que dans tous les cas, l'évaluation du débit s'est faite d'une manière tout à fait réaliste, en constatant simplement la taille du fichier d'information annexe correspondant.

12.5 Résultats

12.5.1 Configuration oracle et temps de traitement

Pour commencer, je vais ici donner les scores obtenus par la configuration oracle. Ces résultats permettent de valider ou d'infirmer le formalisme gaussien que je présente pour la séparation informée, puisqu'ils constituent une borne supérieure aux performances qu'on peut atteindre en utilisant le système paramétrique proposé. Les pistes séparées sont obtenues en utilisant l'algorithme 9.1 page 125 avec l'information annexe de la configuration oracle 9.1.7 page 122, qui revient à utiliser les spectrogrammes originaux des sources pour la séparation.

Pour chacun des 14 extraits de la base et pour chacune des configurations de mixage, instantanée ou convolutive, on trouvera en figure 12.2 page suivante les scores PSM et SDR obtenus par une séparation oracle. En plus des performances de la configuration oracle, on trouvera sur cette même figure une représentation des temps de calculs à l'encodage et au décodage observés pour l'implémentation testée des différentes techniques de séparation. Ces figures permettent de se donner une idée de la complexité relative des opérations effectuées par les codeurs et décodeurs des différentes techniques. Il est cependant clair qu'aucune des techniques testées n'a bénéficié d'une implémentation optimisée et ces durées de traitement sont données à titre indicatif.

12.5.2 Courbes débit-qualité

Après les performances de la configuration oracle données, on trouvera ces courbes débit-δ_{PSM} et débit-δ_{SDR} pour les mélanges instantanés et convolutifs sur la figure 12.3 page 157.

9. J'ai cependant mis en œuvre celui présenté au chapitre 11 dans le contexte du codage informé, comme on le verra au chapitre 16.

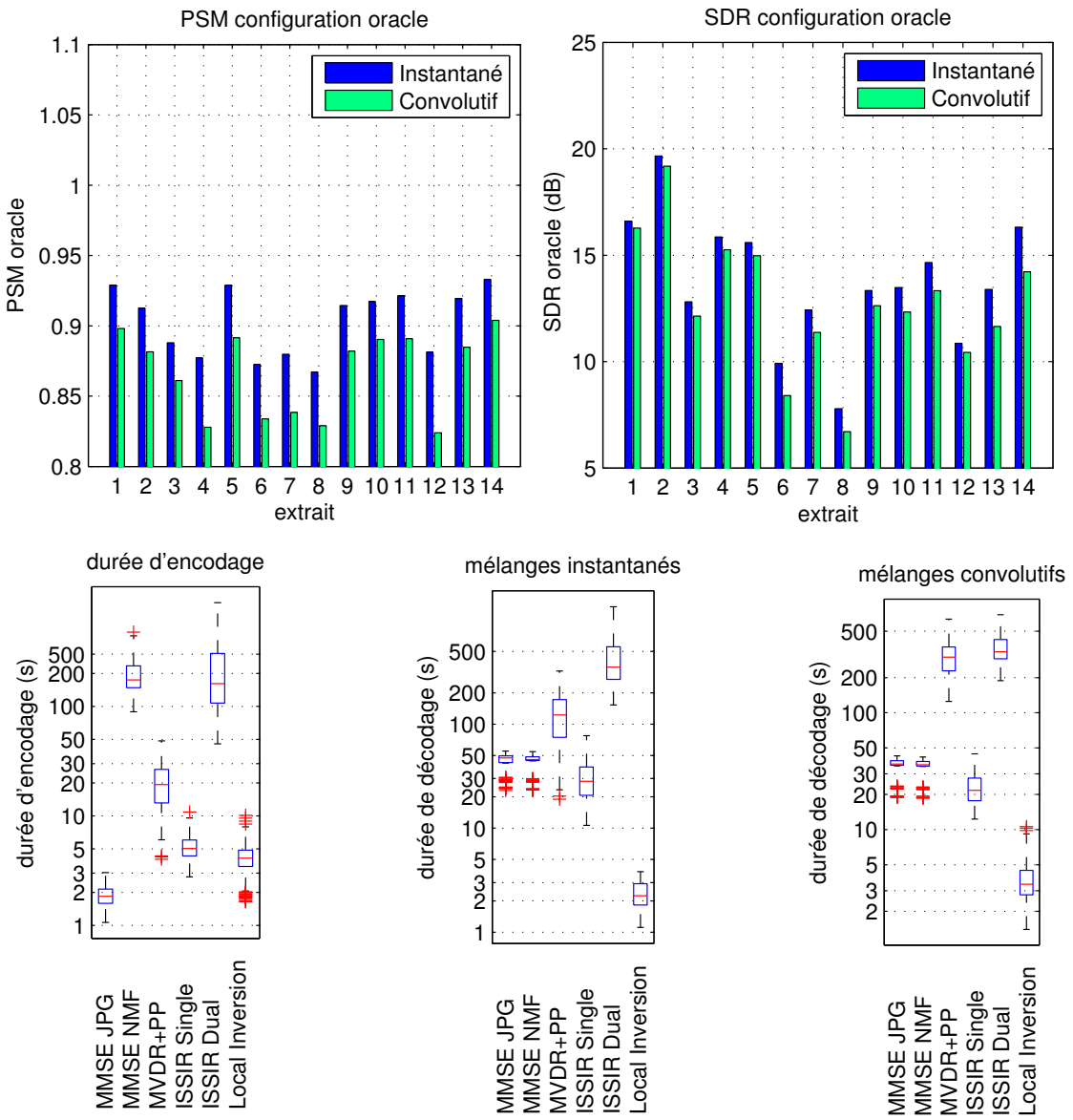

FIGURE 12.2: En haut : performances obtenues sur les 14 extraits de la base par la configuration oracle de l'information annexe, dans le cas d'un mixage instantané et d'un mixage convolutif. La première figure donne les scores PSM et la deuxième les scores SDR de l'oracle. En bas : distribution des durée d'encodage et de décodage observées par les implémentations actuelles des différentes techniques évaluées (d'après [134]).

12.6 Discussion

12.6.1 Performances de la configuration oracle

Les résultats donnés plus haut peuvent donner lieu à de multiples remarques. Tout d'abord, les performances obtenues par la configuration oracle confirment ce que j'ai dit en section 12.2 sur la variabilité de la difficulté du problème de séparation en fonction de l'extrait considéré. On remarque en effet sur cette figure que les performances absolues obtenues par l'oracle en termes de PSM et de SDR sont variables d'un extrait à l'autre. Il est donc bien nécessaire de prendre

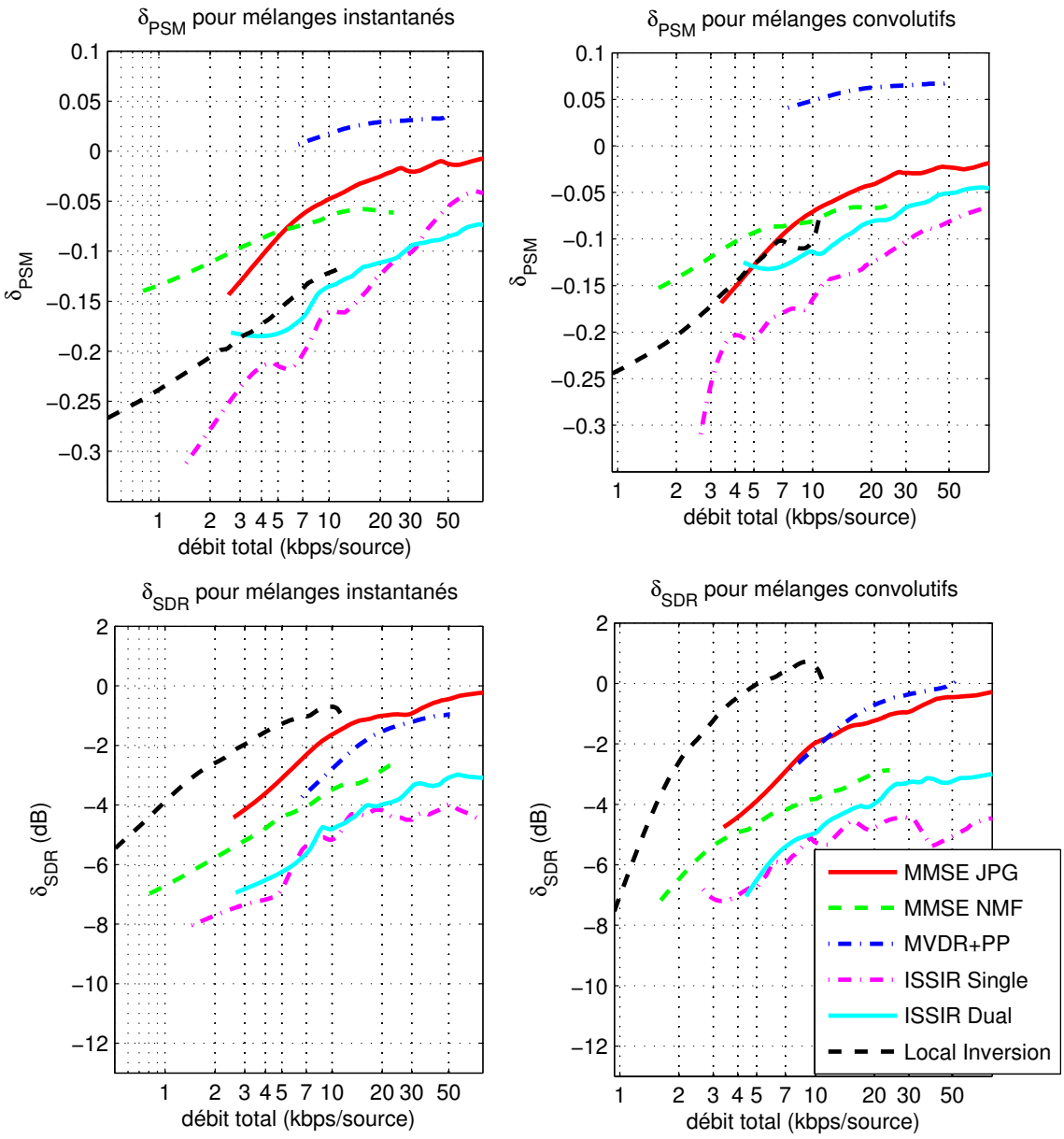

FIGURE 12.3: Courbes de débit-qualité obtenues par les différentes techniques de séparation paramétrique considérées. Chacune de ces courbes correspond au lissage par LOESS [36] d'un nuage de points de plus de 500 éléments. (d'après [134]).

en compte cette difficulté variable dans l'interprétation des scores obtenus. Par exemple, un score SDR de 7dB sur l'extrait 8, difficile, pourra être considéré comme très bon, tandis qu'il est moyen pour l'extrait 2, considérablement plus simple à séparer. Ce constat justifie l'introduction des métriques différentielles δ_{SDR} et δ_{PSM} et présente un intérêt méthodologique, puisqu'il n'est pas encore largement admis que des scores de séparation ne peuvent pas être simplement moyennés.

Dans le même ordre d'idées, on constate une baisse générale des performances de séparation pour les mélanges convolutifs par rapport aux mélanges instantanés. Cette baisse est de quelques dB pour les scores SDR obtenus par la configuration oracle et de quelques pourcents pour le score PSM. Bien qu'elle demeure toute relative, il me semble intéressant de la souligner. De mon point de vue, elle s'explique principalement par le fait que même dans le cas d'un mélange contrôlé en

laboratoire, l'équation de mélange 6.1.15 page 88 sur laquelle repose la séparation de mélanges convolutifs de PGLS demeure une approximation, contrairement à celle 5.2.3 page 77 du mélange instantané, qui est exacte.

Quoi qu'il en soit, les performances obtenues par la configuration oracle consignées en figure 12.2 sont excellentes. Dans le cas d'un mélange instantané tout comme dans celui d'un mélange convolutif, les scores PSM obtenus ne sont jamais inférieurs à 0.8 et peuvent donc être qualifiés de très bons. Ceux obtenus dans le cas instantané sont plutôt proches de 0.9 en général. Lors de tests informels d'écoute, la qualité des sources séparées obtenues est excellente et très largement suffisante pour la plupart des applications envisagées. Ce résultat très fort vient confirmer la validité du formalisme proposé pour la séparation informée.

12.6.2 Courbes de débit-qualité des méthodes proposées

Les méthodes proposées sont désignées par MMSE-NMF et MMSE-JPG dans les figures 12.2 et 12.3. Elles correspondent à l'utilisation du modèle de source NTF et CI, respectivement.

Un premier constat qu'on peut faire à la vue des résultats présentés en figure 12.3 est que les performances obtenues par ces méthodes sont toujours inférieures à celles de la configuration oracle, qui correspondent à la ligne horizontale d'ordonnée $\delta_{SDR} = \delta_{PSM} = 0$. Ce résultat était attendu puisque ces techniques de séparation reposent sur l'utilisation d'approximations pour le modèle de sources $\mathcal{P}(\cdot \mid \theta)$ qui ne peuvent être plus fidèles que l'utilisation des véritables spectrogrammes faite par l'oracle. Quel que soit le débit disponible, les méthodes paramétriques proposées ne permettent pas de dépasser un certain seuil de performances. Même si ce seuil est très bon, cela constitue un inconvénient de l'approche, comme je l'ai souligné en section 9.4 page 126. Il est remarquable que ce constat s'applique à toutes les techniques évaluées, même si MVDR+PP [87] ou la variante de l'inversion locale présentée dans [223] présentent une borne plus élevée.

Les méthodes paramétriques que je propose produisent d'excellentes sources séparées à un débit de $1 - 10$kbps/source.

Malgré cet inconvénient de l'approche paramétrique d'offrir des performances bornées, les débits obtenus et la qualité de séparation correspondante sont tout à fait remarquables. On voit en effet sur la figure 12.3 que les débits totaux observés pour la transmission de l'information annexe s'échelonnent d'environ 1kbps/source à 50kbps/source pour les algorithmes que je propose. Bien entendu, la zone d'utilisation la plus intéressante est constituée par les débits les plus faibles. Il est intéressant de constater que pour des débits inférieurs à 7kbps/source, ce sont les méthodes que j'ai implémentées qui présentent les meilleures courbes de débit-qualité. Ce résultat doit cependant être tempéré par le fait que beaucoup des différences entre les méthodes sur ce point sont plutôt relatives à l'implémentation qu'à des limitations intrinsèques. En particulier, plusieurs d'entre elles peuvent bénéficier de la très bonne compressibilité du modèle de sources NTF ou JPG pour améliorer leurs résultats [10].

Enfin, si on considère un instant les durées d'encodage et de décodage des modèles de sources NTF et CI, présentées en figure 12.2, comme un critère à prendre en compte en pratique, il apparaît que si le modèle CI produit des sources séparées d'une qualité légèrement moins bonne sur le plan perceptif, il est d'une incomparable rapidité d'encodage par rapport au modèle NTF. Si cela est dû en partie à la maturité des implémentations de l'algorithme de compression JPEG par rapport à mon implémentation Matlab de l'algorithme 4.1 page 67, il n'en demeure pas moins que la complexité de ce dernier est intrinsèquement beaucoup plus grande que celle de CI. Ces considérations peuvent avoir leur importance lorsqu'il s'agit de faire un choix sur la technique à implémenter dans un système industriel. Quel que soit le modèle choisi, on peut remarquer que le décodage se fait très rapidement pour les deux modèles [11].

10. J'ai transmis à tous les partenaires du projet DReaM ainsi que diffusé largement sur Internet les algorithmes permettant d'apprendre ces modèles sur des signaux observés.

11. Depuis cette évaluation, Shuhua Zhang a eu la gentillesse de me transmettre sa propre implémentation du décodeur implémentant l'algorithme 9.1 page 125, plus rapide d'un ordre de grandeur par rapport à celle dont je présente les performances ici.

12.6.3 Comparaison avec les autres techniques

Si on veut mettre en perspective les résultats obtenus par les méthodes paramétriques que je propose par rapport à ceux des autres techniques des partenaires académiques de DReaM, plusieurs constats s'imposent.

En premier lieu, il apparaît clairement que le domaine de la séparation informée paramétrique offre aujourd'hui des méthodes implémentables en pratique qui permettent la récupération au décodeur de sources séparées de qualité pour un débit de l'ordre de 5kbps/source. Quand on considère que les techniques originales proposées par Parvaix en [169, 167] nécessitaient un débit de l'ordre de 64kbps pour 5 sources mixées en mono, on mesure l'ampleur des progrès accomplis en quelques années. Je note d'ailleurs que la technique d'inversion locale évaluée ici a bénéficié depuis les travaux de Parvaix d'un important travail d'optimisation et d'amélioration accompli par Shuhua Zhang et Laurent Girin [12]. Ainsi, toutes les approches s'avèrent aujourd'hui compétitives en termes de débit par rapport à l'encodage indépendant des sources par un compresseur audio classique. Cela vient confirmer la validité générale de l'approche informée.

Ensuite, si elles offrent d'excellentes performances, on peut noter que les techniques paramétriques que je propose ne bénéficient pas, loin s'en faut, de la meilleure qualité objective de séparation à tous points de vue. Alors que l'inversion locale démontre de meilleurs scores SDR, l'approche MVDR+PP présente de meilleurs scores PSM. Comme on l'a vu en section 1.3.3, les bonnes performances de l'inversion locale en termes de SDR se font au prix de la présence dans les estimées d'un bruit musical, ennuyeux pour certaines applications [13]. Quant à la technique MVDR+PP [87], il s'agit d'une technique itérative basée sur une formation de voie à distorsion minimale [27].

La technique ISSIR de reconstruction itérative a souffert dans cette évaluation du fait qu'elle se concentre sur le cas des mélanges monophoniques ($I = 1$) et qu'elle n'exploite donc pas comme les autres l'information spatiale pour la séparation. Une évaluation dans sa configuration optimale d'utilisation $I = 1$ aurait démontré dans ce cas sa supériorité par rapport à la technique que je propose ici [200].

12.6.4 Comparaison avec SAOC

J'ai cherché à inclure des implémentations optimisées de SAOC dans les évaluations. À ma connaissance, aucun encodeur SAOC optimisé n'est encore disponible sur le marché et aucun des chercheurs travaillant sur ce standard que j'ai contactés ne m'a répondu positivement.

On peut être surpris du fait que le système de codage SAOC [104, 61, 153, 60, 66] n'ait pas fait l'objet d'une évaluation au même titre que les autres systèmes proposés, compte tenu de la similarité de sa problématique avec celle de la séparation informée. Je suis le premier à regretter ce manque, mais j'ai dû m'y résoudre du fait de l'absence d'implémentations disponibles d'un codeur SAOC optimisé et du manque de temps manifeste des différents auteurs travaillant sur ce système à qui j'ai proposé une collaboration.

On peut cependant établir sur un plan théorique plusieurs constats sur les similitudes et les différences entre le système paramétrique proposé et SAOC. Tout d'abord, les procédures de séparation de SAOC sont basées sur le même formalisme gaussien que celui que j'ai exploité ici [61, 153, 15, 66]. En conséquence, les techniques de séparation considérées dans SAOC apparaissent comme très proches de celles que je considère moi-même. Cependant, le système que je propose se distingue de SAOC par plusieurs points importants :

– Les représentations temps-fréquence que j'ai utilisées contiennent un nombre F de bandes environ 10 fois supérieur à celui considéré par SAOC. Alors qu'une cinquantaine de bandes sont

12. Non seulement la version évaluée ici permet un débit variable, mais elle offre des performances considérablement supérieures.

13. Le bruit musical produit par l'inversion locale peut néanmoins souvent être considéré comme très raisonnable et a été beaucoup réduit dans [223] par l'abandon de la mise à zéro pure et simple des sources "inactives".

suggérées pour SAOC [61, 15], j'en utilise en général 512. Cette différence a son importance, puisqu'elle permet une très bonne séparation des différentes sources, dans la mesure où les recouvrements spectraux sont bien plus courants sur 64 bandes que sur 512. On pourrait bien sûr utiliser autant de bandes de fréquences dans SAOC que dans le système que je propose, mais cela mènerait à des débits prohibitifs, ce qui m'amène à mon deuxième point.
- SAOC ne considère pas de modèle de sources permettant d'encoder les dépendances à long terme des DSP. En fait, il est restreint à l'utilisation d'un modèle à très court terme dont les paramètres sont les DSP moyennes des signaux sur chaque bande de fréquence considérée, directement quantifiées et encodées par des techniques *ad-hoc*. Le formalisme que je suggère permet au contraire d'utiliser n'importe quel modèle pour la DSP des sources, dont le modèle NTF, très puissant, qui permet d'énormément réduire la redondance dans les paramètres à transmettre. Sa force en pratique est de parvenir à encoder sur 6kbps/objet pour 512 canaux ce que SAOC encode pour 64 canaux.
- SAOC est restreint aux mélanges linéaires instantanés alors que l'approche que je propose, on l'a vu, permet une grande souplesse dans la modélisation du processus de mixage. De plus, on ne peut pas étendre SAOC aux mélanges convolutifs puisque cela reviendrait à ne pouvoir accepter que des filtres de mélange extrêmement courts pour que l'approximation 6.1.15 page 88 soit exacte, compte tenu de la grande largeur de chaque bande de fréquence considérée.
- Les débits totaux considérés par SAOC s'avèrent à peu près équivalents à ceux démontrés par les systèmes évalués, bien que les techniques proposées permettent des débits inférieurs. [61] évoque en effet le chiffre de 3kbps/source/canal de mélange auxquels s'ajoute un coût fixe de 3kbps. Pour un mélange stéréo composé de 5 sources, cela correspond à un débit total de 6.6kbps/source. Il serait bien sûr nécessaire de pouvoir évaluer la qualité correspondante des sources séparées de manière à pouvoir tirer une conclusion, ce qui ne m'a pas été possible en l'absence de codeur disponible ou de signaux obtenus après séparation par SAOC.
- Contrairement à l'approche que je propose dans cette partie, SAOC inclut un module de codage du résiduel [66, 61, 60], qui permet de ne pas être restreint à des performances bornées. Il s'agit donc d'un avantage de SAOC sur le système proposé ici. Ce système est cependant ad-hoc puisqu'il procède à l'encodage par MPEG2-AAC de l'erreur de reconstruction des sources [61, 66, 60]. Je reviendrai plus longuement sur cette idée en section 14.3.2, lorsque j'aborderai le codage informé.

En tout état de cause, il faut nuancer ces trois premiers points par le fait que SAOC est apparu très tôt : les premiers travaux dans cette direction datent de 2007. Les modèles que je propose ont bénéficié du travail effectué depuis par une large communauté de chercheurs dans le domaine de la séparation de sources et de la modélisation paramétrique des DSP de signaux audio. SAOC était alors déjà en cours de normalisation. De manière à être juste dans l'appréciation de la situation, je pense qu'on peut voir le modèle proposé comme mettant en œuvre les idées déjà présentes dans SAOC en les généralisant à un cadre applicatif plus large.

12.6.5 Conclusion générale

Cette évaluation assez complète de plusieurs techniques paramétriques de séparation informée a permis de mettre en lumière plusieurs faits marquants.

- Les débits requis pour une bonne séparation informée paramétrique s'échelonnent entre 1kbps/source et 10kbps/source.
- La qualité des sources estimées est bonne.
- Les performances de toutes les méthodes paramétriques sont bornées et elles ne permettent donc pas d'applications haute-fidélité. Par contre elles sont tout à fait adéquates à des applications d'écoute active.
- Les durées d'encodage sont variables en fonction des méthodes, mais le décodage peut être effectué en temps réel.

Conclusion de la troisième partie

En partie précédente, j'ai introduit un formalisme général pour la séparation de sources. Dans cette partie, j'en ai considéré une configuration originale, appelée séparation informée. Le principe de la séparation informée est de pouvoir utiliser des paramètres qui ont été appris sur les sources à séparer elles-mêmes. Une telle situation peut se produire dans un contexte de codage-décodage, où les sources ne sont disponibles qu'à la première étape de codage, durant laquelle des paramètres peuvent être calculés. Le décodeur, muni de la seule connaissance des mélanges et de ces paramètres, peut effectuer une séparation pour récupérer les sources.

La formalisation faite des différentes configurations de séparation informée permet d'envisager tous les cas de figure considérés dans la littérature.

Pour commencer, je me suis efforcé dans le chapitre 9 de formaliser les différentes configurations rencontrées dans le domaine de la séparation informée. Ainsi, j'ai distingué trois composantes majeures dans la caractérisation d'un tel problème. Tout d'abord, j'ai mis en lumière le fait que chaque signal observé au codeur peut être monocanal ou multicanal. Dans le premier cas, je l'ai modélisé comme la réalisation du processus source et dans le deuxième cas comme une image diffuse de cette source. Ensuite, chaque source fait l'objet d'un mixage en vue de la production des mélanges. Ainsi, j'ai envisagé la possibilité pour chaque source d'être mixée de manière convolutive ou diffuse. Enfin, une tâche de séparation informée se caractérise par la nature des signaux qu'on souhaite récupérer au décodeur : on peut souhaiter récupérer les sources ponctuelles, par exemple en vue d'une respatialisation, ou bien se contenter des images des sources si l'objectif est de les supprimer du mélange comme c'est le cas dans une application de type karaoké.

Une fois cette formalisation établie, j'ai montré que les techniques présentées dans la partie précédente fournissent une solution naturelle à l'ensemble de ces problématiques. J'ai ainsi montré qu'au décodeur, on peut utiliser à l'identique les résultats de mon étude sur la séparation de processus gaussiens pour récupérer les sources. Le travail du décodeur consiste alors simplement à choisir pour estimées des sources leur valeur la plus probable *a posteriori*. Cette séparation suppose que l'information annexe calculée au codeur est constituée des différentes DSP des sources ainsi que des paramètres de mixage. Il se trouve que le codeur peut très facilement estimer ces valeurs, puisqu'il dispose des signaux sources.

Pour effectuer une séparation de PGLS, le décodeur a besoin des DSP des sources, qui sont lourdes à encoder. J'ai proposé de les approcher de manière à pouvoir les transférer efficacement. La séparation se fait en estimant les sources par leur valeur la plus probable *a posteriori*.

Malheureusement, il n'est pas possible de transférer facilement du codeur vers le décodeur une quantité aussi importante de paramètres que la valeur de la DSP de chaque source pour chaque trame et chaque fréquence. En conséquence, j'ai montré qu'il est possible de mettre en œuvre les différents modèles prévus à cet effet et que j'avais présentés en première partie. Si on utilise un tel modèle pour les DSP des sources, il devient possible de compresser très efficacement l'information nécessaire pour que le décodeur puisse effectuer la séparation.

Pour obtenir un système opérationnel de séparation informée utilisant le modèle gaussien proposé, il ne reste donc plus qu'à expliquer comment le codeur peut procéder à l'apprentissage des différents paramètres nécessaires à la séparation. J'ai commencé par aborder ce problème dans le chapitre 10 dans un cas particulier où tous les mixages considérés sont convolutifs. Ce faisant, j'ai montré comment les paramètres de sources et de mixage peuvent être appris et comment il faut

les encoder pour produire une information annexe à transférer au décodeur qui soit la plus petite possible.

Une fois ce cas particulier des seuls mixages convolutifs traité, je me suis penché sur le cas général dans le chapitre 11 qui permet de prendre en compte toutes les configurations permises par la formalisation envisagée. Dans ce cas général, l'architecture du codeur que j'ai proposée permet d'estimer les différents paramètres et de les transmettre efficacement au décodeur.

Enfin, j'ai procédé dans le chapitre 12 à une large campagne d'évaluation pour laquelle les différentes approches de séparation informée existantes dans la littérature ont été comparées sur le même corpus et en utilisant les mêmes métriques. Cette évaluation a permis de montrer que les systèmes actuels sont très efficaces pour récupérer au décodeur des sources séparées de qualité en utilisant pour l'information annexe des débits de l'ordre de $5-10$kbps/source. Parmi ces systèmes, ceux que j'ai proposés se comportent de manière exemplaire.

Quoiqu'il en soit, cette large campagne d'évaluation a permis de constater que si ces systèmes s'avèrent souvent très efficaces, il souffrent tous d'un inconvénient majeur : leurs performances sont bornées. En effet, il est impossible avec une telle approche de la séparation informée de dépasser les meilleures performances que l'algorithme de séparation envisagé peut atteindre. Quelle que soit la valeur de cette borne, qui peut dépendre de la méthode de séparation choisie, il sera impossible au système considéré de la dépasser.

J'ai établi un lien original entre les systèmes existants de séparation informée et le codage audio paramétrique.

J'ai interprété ce résultat en établissant un lien original entre ces systèmes de séparation informée et les codeurs audio paramétriques. Tous deux font en effet l'hypothèse que les signaux à récupérer obéissent parfaitement au modèle choisi. Si le modèle sous-jacent à un codeur audio paramétrique repose la plupart du temps sur des hypothèses portant sur la nature des signaux à coder, celui des systèmes de séparation informée suppose en somme que les mélanges et les sources sont parfaitement décrits par les paramètres choisis, ce qui est faux en général.

Pour cette raison, j'ai nommé *paramétriques* tous les systèmes informés qui se contentent au décodeur d'effectuer une séparation des mélanges avec des paramètres calculés et transmis par le codeur.

Quatrième partie

Séparation informée par codage

Introduction

Dans la partie précédente, j'ai proposé un système de séparation informée dont on a vu en section 9.4 qu'il est équivalent à un codeur audio *paramétrique*. En effet, il commence par faire sur les sources à compresser l'hypothèse qu'elles sont correctement expliquées selon un modèle particulier. Ensuite, les paramètres de ce modèle sont estimés au codeur et utilisés au décodeur pour reconstruire le signal.

Le modèle supposé sur les sources dans ce codeur audio paramétrique ne porte pas comme c'est d'habitude le cas [33] sur une forme particulière des données à transmettre. Au lieu de cela, il suppose connu à la fois au codeur et au décodeur un ensemble de signaux appelés *mélange*s. Muni de ces signaux, il repose sur l'hypothèse qu'ils sont correctement modélisés comme un mixage des sources à transmettre [14]. C'est en structurant de manière paramétrique la covariance entre sources et mélanges qu'est rendu possible le codage des sources : elles sont estimées au décodeur de manière paramétrique par séparation des mélanges aux moindres carrés.

Dans un codeur audio paramétrique classique, c'est lorsque le signal à compresser n'obéit pas correctement au modèle de production choisi qu'apparaissent de fortes distorsions sur les signaux reconstruits. De manière analogue, ce sera ici lorsqu'on ne peut plus considérer que les mélanges s'expliquent correctement comme un mixage des signaux sources selon les paramètres choisis. Cela peut arriver si les paramètres sont mal estimés, si le modèle de mixage est inadéquat ou bien si les mélanges ne sont vraiment pas produits par mixage des sources. Dans tous les cas, outre le risque d'une inadéquation avec le modèle et donc de la présence d'une forte distorsion, la limitation fondamentale d'un codeur paramétrique est qu'il présente des performances *bornées*, dépendantes du signal à encoder. Quel que soit le débit disponible pour l'encodage, il ne sera pas possible en général d'atteindre n'importe quelle qualité. J'ai mis en évidence cet inconvénient de toutes les techniques paramétriques évaluées au chapitre 12 : à partir d'un certain débit, leurs performances ne s'améliorent plus. Cela peut être gênant dans des applications où une haute-fidélité est requise, pour lesquelles il est désirable de pouvoir *garantir* une certaine qualité des signaux séparés, ce qui n'est pas possible en l'état.

Je remercie Alexey Ozerov pour l'impulsion très importante qu'il a donnée à mes travaux et pour la source d'inspiration constante que nos discussions ont représenté pour moi. Ce qui suit est un résumé de nos articles communs sur le sujet [160, 136, 161].

Dans cette partie, la problématique de la séparation informée de sources audio est considérablement étendue. En fait, on va voir à présent qu'il est possible de la considérer comme une instance particulière d'application de la théorie du *codage de source* [89] et d'en déduire une nouvelle manière d'aborder le problème qui bénéficie d'avantages importants. Cette extension permet principalement de garantir une qualité de séparation aussi bonne que désiré, pour peu qu'un débit suffisant soit disponible. À contrario, elle permet de garantir une performance optimale pour un débit donné.

Le codage de source se concentre sur la manière optimale de transmettre un message caractérisé par ses propriétés statistiques depuis un *codeur* vers un *décodeur* en minimisant à la fois un critère de distorsion et le débit nécessaire à la transmission. En substance, le codage de source est le formalisme théorique qui s'attache à donner des bornes au débit minimal auquel on peut espérer transmettre fidèlement les réalisations $\tilde{\boldsymbol{s}}$ d'un processus *source*, caractérisées par une distribution de probabilité $p\left(\tilde{\boldsymbol{s}} \mid \Theta\right)$. Elle fournit en outre des méthodes qui permettent d'atteindre ces bornes

14. Une approche discriminante telle qu'évoquée en section 10.1 page 129 supposerait plutôt que les sources sont formées à partir des mélanges par filtrage.

dans certains cas et constitue ainsi la clé de voûte de tous les systèmes modernes de communication numérique.

Le chaînon manquant entre séparation informée et codage de source m'a été indiqué par Alexey Ozerov qui, au courant de mes travaux, m'a suggéré d'orienter mes recherches dans la direction suivante. Si on suppose connue une information $\tilde{\boldsymbol{x}}$ à la fois au codeur et au décodeur, le codage *informé* consistera à appliquer les résultats du codage de source pour la transmission de $\tilde{\boldsymbol{s}}$, à la différence que la distribution considérée pour l'encodage sera $p(\tilde{\boldsymbol{s}} \mid \tilde{\boldsymbol{x}}, \Theta)$ et non plus seulement $p(\tilde{\boldsymbol{s}} \mid \Theta)$. Je formaliserai cela plus en détail en section 14.1.1 page 183, mais on peut montrer que la connaissance de $\tilde{\boldsymbol{x}}$ permet en général de réduire le débit nécessaire à la transmission de $\tilde{\boldsymbol{s}}$. Pour peu que $\tilde{\boldsymbol{x}}$ apporte beaucoup d'information sur $\tilde{\boldsymbol{s}}$, le gain correspondant en débit sera considérable. Au pire, $\tilde{\boldsymbol{x}}$ est indépendant de $\tilde{\boldsymbol{s}}$, auquel cas la situation se ramène au codage de source classique.

Comme on s'en doute, le choix des notations dans cette courte présentation est délibéré : le formalisme de séparation de processus gaussiens que j'ai présenté en partie II fournit précisément une telle distribution *a posteriori* $p(\tilde{\boldsymbol{s}} \mid \tilde{\boldsymbol{x}}, \Theta)$ des sources étant donnés les mélanges. La portée de ce constat est grande, parce qu'il ouvre les portes à l'utilisation d'une théorie extrêmement mature telle que celle du codage de source pour la résolution du problème posé par la séparation informée.

Cette partie est organisée de la manière suivante. Au chapitre 13, je présente la théorie du codage de source. Cette théorie est appliquée au cas particulier où les mélanges sont connus à la fois au codeur et au décodeur au chapitre 14. Je détaille la structure du codeur et du décodeur correspondants au chapitre 15 et le système est évalué au chapitre 16, où j'en démontre la supériorité sur le système paramétrique de la partie précédente [15].

15. Tout au long de cette partie, D désigne un réel positif appelé distorsion et non plus la dimension du domaine de définition des signaux, qui est fixée à 1 dans tous les cas.

Chapitre 13

Codage de source

Dans ce chapitre, je vais présenter les résultats de la théorie du codage de source importants pour la suite de mon exposé. Je me limiterai ici à la présentation de certains principes généraux importants et de certains résultats particuliers utiles pour la compréhension et l'implémentation de la méthode de codage proposée. La présentation simplifiée de la théorie débit-distorsion faite aux sections 13.2 et 13.3 est inspirée des notes de cours de Bastiaan Kleijn [120] sur le sujet.

Il faut avant tout voir ce chapitre comme résumant les bases théoriques nécessaires à la compréhension de la technique de codage des sources informée par leurs mélanges, que je décrirai plus loin aux chapitres 14 et 15. Le lecteur désireux de trouver une présentation solide et complète de cette théorie peut consulter le texte classique [89] disponible gratuitement en ligne [1].

13.1 Variables discrètes

13.1.1 Entropie et théorème du codage de source

Soit $\overline{s}$ une variable aléatoire à valeurs dans un ensemble fini $\mathcal{C}_N = \{c_1, \ldots, c_N\}$ de N éléments. Je noterai $\overline{\boldsymbol{s}}$ une de ses réalisations et je suppose connue la loi de probabilité de $\overline{s}$:

$$\forall c_n \in \mathcal{C}_N,\, P_{\overline{s}}(n) = P\left(\overline{\boldsymbol{s}} = c_n\right).$$

L'entropie de $\overline{s}$, notée $H(\overline{s})$, est définie par :

$$H(\overline{s}) = -\sum_{n=1}^{N} P_{\overline{s}}(n) \log_2 P_{\overline{s}}(n), \tag{13.1.1}$$

et s'exprime en *bits* ou en *nats* si on utilise plutôt le log naturel dans sa définition. $H(\overline{s})$ peut se comprendre comme l'incertitude qu'on a sur la valeur de $\overline{\boldsymbol{s}}$ avant son observation. L'entropie d'une variable aléatoire discrète est nécessairement positive. Ceci se vérifie facilement en considérant que $\forall n,\, P_{\overline{s}}(n) \leq 1$.

Par exemple, si $\mathcal{C}_1$ est composé d'un seul élément c dont la réalisation est certaine, il n'y a aucune incertitude sur la valeur $\overline{\boldsymbol{s}} = c$. Dans ce cas, on vérifie que $H(\overline{s}) = 0$. À l'opposé, si $\mathcal{C}_N$ est constitué de N éléments équiprobables ($P_{\overline{s}}(n) = \frac{1}{N}$), alors l'incertitude sur la valeur de $\overline{\boldsymbol{s}}$ est maximale. On vérifie en effet qu'on a dans ce cas $H(\overline{s}) = \log_2 N$ bits et que l'entropie d'une variable aléatoire discrète ne peut pas être plus grande.

Supposons maintenant qu'un informateur ait pu observer la valeur de $\overline{\boldsymbol{s}}$ et souhaite nous l'indiquer de la manière la plus concise possible. Pour ce faire, nous conviendrons ensemble d'un *code*. Par définition, un code associe à chaque élément $c_n \in \mathcal{C}_N$ une suite de 0 et de 1 de longueur r_n qui portera le nom de *symbole*. Le problème qu'aborde la théorie du codage de source est de trouver le code qui permet de réduire au maximum la longueur moyenne R du symbole par lequel

1. ee.stanford.edu/~gray/it.pdf

l'observateur nous indiquera la valeur de $\overline{\boldsymbol{s}}$:

$$R = \sum_{n=1}^{N} r_n P_{\overline{s}}(n)\,. \tag{13.1.2}$$

Puisqu'il s'agit d'un nombre de bits moyen par symbole, R peut se comprendre comme le *débit moyen* par symbole. En outre, on conviendra d'un code sans ambigüité, c'est-à-dire pour lequel chaque symbole permet d'identifier de manière unique un élément de $\mathcal{C}_N$.

C'est le sens du théorème du codage de source que de faire le lien entre le débit moyen minimal $R^\star$ qu'on peut effectivement atteindre et l'entropie de la variable aléatoire considérée :

$$H(\overline{s}) \leq R^\star \leq H(\overline{s}) + 1. \tag{13.1.3}$$

Ce résultat parait intuitif : si l'incertitude est faible quant à la valeur prise par $\overline{\boldsymbol{s}}$, le symbole se réduira la plupart du temps à une courte confirmation du résultat attendu. Si au contraire l'incertitude est grande, il y aura besoin de plus d'informations pour décrire le message. Le fait que l'entropie s'exprime en bits se comprend alors : elle est directement liée à la longueur moyenne des symboles binaires nécessaires au codage de réalisations indépendantes d'une variable aléatoire discrète.

13.1.2 Entropie conditionnelle et information mutuelle

Considérons à présent deux variables aléatoires discrètes $\overline{s}$ et $\overline{x}$ à valeurs dans le même ensemble fini $\mathcal{C}_N$ et $\overline{\boldsymbol{s}}$ et $\overline{\boldsymbol{x}}$ une de leurs réalisations. Supposons qu'on connaisse leur probabilité jointe [2] $p(\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}} \mid \Theta)$. On peut définir de la même manière qu'en 13.1.1 leur entropie jointe $H(\overline{s}, \overline{x})$, qui correspondra à l'incertitude portant sur la valeur jointe $(\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}})$ avant observation. On aura encore $H(\overline{s}, \overline{x}) \geq 0$.

Puisqu'on connaît la probabilité jointe $p(\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}} \mid \Theta)$, on connaît aussi la probabilité conditionnelle [3] $p(\overline{\boldsymbol{s}} \mid \overline{\boldsymbol{x}}, \Theta)$. Une grandeur importante dans ce contexte est l'entropie conditionnelle $H(\overline{s} \mid \overline{x})$ de $\overline{s}$ sachant $\overline{x}$, qui se définit comme :

$$H(\overline{s} \mid \overline{x}) = -\sum_{\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}} \in \mathcal{C}_N} p(\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}} \mid \Theta) \log_2 p(\overline{\boldsymbol{s}} \mid \overline{\boldsymbol{x}}, \Theta)\,. \tag{13.1.4}$$

L'entropie conditionnelle s'interprète comme l'incertitude qui demeure en moyenne sur la valeur $\overline{\boldsymbol{s}}$ si la réalisation de $\overline{x}$ est observée. Il s'agit encore d'une grandeur positive. Par exemple, si la valeur $\overline{\boldsymbol{s}}$ s'obtient de manière déterministe à partir de $\overline{\boldsymbol{x}}$, on aura $H(\overline{s} \mid \overline{x}) = 0$. À l'opposé, si $\overline{x}$ et $\overline{s}$ sont indépendantes, on aura $H(\overline{s} \mid \overline{x}) = H(\overline{s})$.

L'entropie conditionnelle vérifie $H(\overline{s} \mid \overline{x}) \leq H(\overline{s})$ et le théorème du codage de source 13.1.3 s'applique pour le codage de $\overline{s} \mid \overline{x}$. Donc, si $\overline{\boldsymbol{x}}$ est connu au codeur et au décodeur, prendre en compte cette connaissance ne peut que baisser le débit nécessaire à la transmission de $\overline{\boldsymbol{s}}$. C'est le principe du codage *a posteriori* qu'on reverra au chapitre suivant.

On peut montrer le résultat important suivant :

$$H(\overline{s}, \overline{x}) = H(\overline{x}) + H(\overline{s} \mid \overline{x})\,, \tag{13.1.5}$$

où toutes les grandeurs sont positives, ainsi que :

$$H(\overline{s} \mid \overline{x}) \leq H(\overline{s})\,,$$

qui se comprend aisément puisque cela signifie que l'incertitude qu'on a sur la valeur de $\overline{\boldsymbol{s}}$ ne peut que baisser si on connaît celle de $\overline{\boldsymbol{x}}$. Au pire, elle reste identique si $\overline{s}$ et $\overline{x}$ sont indépendantes.

2. Comme d'habitude, Θ désigne un lot d'hyperparamètres caractérisant au cas échéant la distribution $p(\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}} \mid \Theta)$.

3. $p(\overline{\boldsymbol{s}} \mid \overline{\boldsymbol{x}}, \Theta)$ s'obtient à partir de $p(\overline{\boldsymbol{s}}, \overline{\boldsymbol{x}} \mid \Theta)$ en ne gardant que les entrées pour lesquelles $\overline{x} = \overline{\boldsymbol{x}}$ et en les normalisant pour qu'elles somment à 1. Voir la section 2.2.1 page 22.

Une dernière grandeur très importante pour la suite est celle d'*information mutuelle* $I(\overline{s}, \overline{x})$, qui se définit simplement comme :

$$\begin{aligned} I(\overline{s}, \overline{x}) &= H(\overline{s}) - H(\overline{s} \mid \overline{x}) \\ &= H(\overline{x}) - H(\overline{x} \mid \overline{s}). \end{aligned} \tag{13.1.6}$$

– L'information mutuelle $I(\overline{s}, \overline{x})$ entre deux variables aléatoires peut donc se comprendre comme mesurant la *réduction moyenne d'incertitude* sur la valeur de $\overline{\boldsymbol{s}}$ apportée par la connaissance de $\overline{\boldsymbol{x}}$ et vice-versa.
– Il s'agit d'une grandeur positive :

$$I(\overline{s}, \overline{x}) \geq 0.$$

1. Si elle est nulle, alors $H(\overline{s}) = H(\overline{s} \mid \overline{x})$ et la connaissance de $\overline{\boldsymbol{x}}$ n'aura pas d'intérêt particulier pour le codage de $\overline{\boldsymbol{s}}$. Dans ce cas, $\overline{s}$ et $\overline{x}$ sont indépendantes.
2. Si au contraire elle n'est pas négligeable, alors il sera intéressant de prendre en compte une éventuelle connaissance de $\overline{\boldsymbol{x}}$ pour coder $\overline{\boldsymbol{s}}$, parce que le théorème du codage de source nous indique dans ce cas que le débit nécessaire est plus petit.

13.1.3 Codage sans perte

Considérons toujours une variable aléatoire discrète $\overline{s}$ à valeurs dans un ensemble discret $\mathcal{C}_N$, dont la loi de probabilité $P_{\overline{s}}(n)$ est connue. Supposons qu'on dispose d'une séquence $\overline{\boldsymbol{s}^L}$ de L réalisations indépendantes de $\overline{s}$ à encoder :

$$\overline{\boldsymbol{s}^L} = [\overline{\boldsymbol{s}_1}, \dots \overline{\boldsymbol{s}_L}]. \tag{13.1.7}$$

Le théorème du codage de source nous indique que le débit moyen minimal $R^\star$ pour le faire vérifie 13.1.3. Cependant, il n'indique pas de manière pratique comment réaliser un tel codage et c'est le champ de recherche du *codage sans perte* qui s'attelle à fournir une réponse concrète à cette question. D'une manière générale, le principe d'un codeur sans perte est d'attribuer à une valeur très probable un symbole court et un symbole plus long à une valeur peu probable. Ainsi, les symboles émis sont d'une longueur R définie en 13.1.2 minimisée en moyenne.

Je considère deux techniques de codage sans perte : le codage de Huffman et le codage arithmétique. Le premier est utilisé pour coder des réalisations indépendantes d'une même variable aléatoire. Le deuxième est utilisé pour l'encodage de longues séquences de réalisations indépendantes de variables qui n'ont pas forcément la même loi.

Il existe de nombreuses manières connues de mettre au point des codes permettant d'atteindre la borne 13.1.3 pour des séquences à encoder asymptotiquement longues. Dans cette section, j'évoque les deux principaux algorithmes mis en œuvre dans le cadre de mon travail, qui sont le codage de Huffman et le codage arithmétique. Dans la mesure où ces algorithmes sont omniprésents dans l'informatique moderne, je ne les présenterai pas en détail mais je renvoie le lecteur intéressé à la consultation d'un des nombreux livres, articles, ou sites internet qui leur sont consacrés [4].

Codage de Huffman L'algorithme de Huffman [108], déjà évoqué en section 10.3, est optimal dans le sens où on démontre qu'il est impossible de mettre au point un code non ambigu présentant une longueur moyenne R par symbole plus courte que celle obtenue par cet algorithme.

Dans la mesure où il s'agit de l'algorithme de codage le plus connu et que de très nombreuses présentations en sont faites dans la littérature, je n'en détaillerai pas le principe outre mesure. Je me contenterai d'indiquer qu'il nécessite une étape préliminaire de construction d'un *dictionnaire* à

4. Les pages Wikipédia `http://en.wikipedia.org/wiki/Huffman_coding` et `http://en.wikipedia.org/wiki/Arithmetic_coding` me semblent de bons points de départ.

partir de la probabilité $P_{\overline{s}}(n)$ de la source, qu'il faut transmettre au décodeur. Ce dictionnaire indique le symbole correspondant à chaque valeur possible de la source s. Les étapes d'encodage et de décodage consistent simplement à remplacer les valeurs observées par leur code et réciproquement.

Les avantages et les inconvénients de la procédure sont résumés dans le tableau 13.1.

Les avantages d'un encodage de Huffman
– Il est optimal : on ne peut pas mettre au point un code présentant une longueur moyenne plus courte.
– Les procédures d'encodage et de décodage sont simples.
Les inconvénients d'un encodage de Huffman :
– Il nécessite la construction d'un dictionnaire donnant le symbole attribué à chaque réalisation possible de la variable. La taille de ce dictionnaire peut devenir prohibitive si on considère des variables vectorielles.
– À chaque valeur possible, on associe un symbole qui a un nombre entier de bits. Cela peut être gênant si l'entropie de la variable aléatoire est faible.
– Il est nécessaire de transmettre le dictionnaire du codeur vers le décodeur.
– Toutes les valeurs à encoder doivent être des réalisations indépendantes de la *même* variable aléatoire.

TABLE 13.1: Avantages et inconvénients d'un encodage de Huffman

Codage arithmétique Le codage arithmétique [180, 171] est une procédure spécifiquement conçue pour l'encodage de *séquences* $\overline{\boldsymbol{s}^L}$ du type de 13.1.7, plutôt que de réalisations $\overline{\boldsymbol{s}_l}$ particulières. Il se démarque de l'encodage de Huffman par le fait qu'il n'attribue pas un symbole indépendant à chaque élément de la séquence, mais que c'est plutôt à $\overline{\boldsymbol{s}^L}$ dans son ensemble qu'il attribue une représentation binaire.

Le principe du codage arithmétique peut se comprendre simplement si on introduit la fonction de répartition $F_{\overline{s}}(c_n)$, ou $F_{\overline{s}}(n)$ d'une variable aléatoire discrète $\overline{s}$, illustrée en figure 13.1.

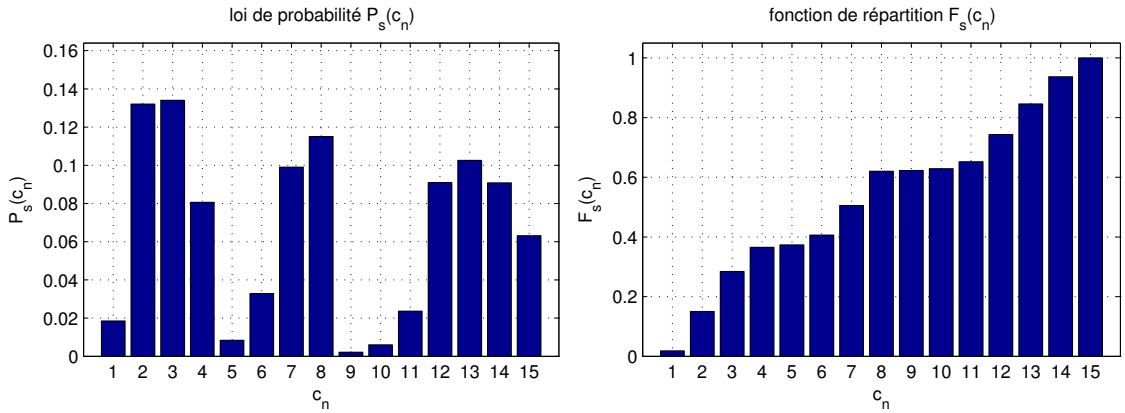

FIGURE 13.1: Loi de probabilité $P_{\overline{s}}(c_n)$ et la fonction de répartition $F_{\overline{s}}(c_n)$ correspondante.

Définition 5. Soit $\overline{s}$ une variable aléatoire discrète à valeurs dans un ensemble discret $\mathcal{C}_N = \{c_1, \ldots, c_N\}$, de loi de probabilité $P_{\overline{s}}(c_n)$. Sa fonction de répartition $F_{\overline{s}}(c_n) \in [0\,;\,1]$ est donnée

par :

$$\forall n = 1, \dots N,\ F_{\overline{s}}(c_n) = \sum_{k=1}^{n} P_{\overline{s}}(c_k),$$

que je noterai indifféremment $F_{\overline{s}}(c_n)$ ou $F_{\overline{s}}(n)$.

Le principe du codage arithmétique peut être introduit en considérant pour commencer une seule réalisation $\overline{\boldsymbol{s}}$ de $\overline{s}$. Muni de la fonction de répartition, on voit bien sur la figure 13.1, qu'à chaque valeur c_n pour $\overline{\boldsymbol{s}}$ correspond un unique intervalle

$$[F_{\overline{s}}(n-1)\ ;\ F_{\overline{s}}(n)] \subset [0\,;\,1],$$

où on a pris soin de définir $F_{\overline{s}}(0) = 0$. On peut donc identifier de manière unique une réalisation $\overline{\boldsymbol{s}} = c_n$ de $\overline{s}$ par le centre q_n de cet intervalle :

$$q_n = \frac{F_{\overline{s}}(n-1) + F_{\overline{s}}(n)}{2}$$

où il suffit pour l'encoder d'arrondir la représentation binaire de q_n à une précision en bits de

$$\lceil -\log_2 (F_{\overline{s}}(n-1) - F_{\overline{s}}(n)) \rceil + 1 = \lceil -\log_2 P_{\overline{s}}(c_n) \rceil + 1, \tag{13.1.8}$$

dans la mesure à cela garantit qu'aucun autre $\{q_{n'}\}_{n' \neq n}$ n'a la même représentation. La longueur moyenne du code correspondant sera donc de :

$$R = 1 + \sum_{n=1}^{N} P_{\overline{s}}(n) \lceil -\log_2 P_{\overline{s}}(n) \rceil,$$

ce qui donne :

$$H(\overline{s}) + 1 \leq R \leq H(\overline{s}) + 2.$$

On voit que cette manière d'encoder une seule réalisation $\overline{\boldsymbol{s}}$ offre une longueur moyenne en bits moins bonne que la borne théorique 13.1.3 et en particulier moins bonne qu'un codage de Huffman.

L'intérêt de cette approche apparaît si au lieu de considérer une seule réalisation $\overline{\boldsymbol{s}}$ de la variable aléatoire, on en considère une séquence $\overline{\boldsymbol{s}^L}$, dont tous les éléments sont indépendants. Même s'il est grand, il y a un ensemble fini $\mathcal{C}_N^L$ de N^L séquences possibles et puisqu'on suppose que tous les éléments d'une séquence sont indépendants, on peut calculer la probabilité d'une séquence donnée $\overline{\boldsymbol{s}^L}$ à partir de celles de ses constituants par :

$$P\left(\overline{\boldsymbol{s}^L}\right) = \prod_{l=1}^{L} P(\overline{s_l}). \tag{13.1.9}$$

On définit alors une nouvelle variable aléatoire discrète $\overline{s^L}$ de loi définie en 13.1.9, dont $\overline{\boldsymbol{s}^L}$ est une réalisation. Si on considère la fonction de répartition $F_{\overline{s^L}}\left(\overline{\boldsymbol{s}^L}\right)$ correspondante, le codage arithmétique de la séquence consiste à la décrire dans son ensemble comme le centre de l'intervalle de $[0\,;\,1]$ correspondant à $\overline{\boldsymbol{s}^L}$ pour la fonction de répartition $F_{\overline{s^L}}$. Soit k l'indice de $\overline{\boldsymbol{s}^L}$ dans $\mathcal{C}_N^L$. De manière identique à 13.1.8, la précision en bits nécessaire à identifier cet intervalle de manière unique parmi tous les autres est de :

$$\left\lceil -\log_2 \left(F_{\overline{s^L}}(k-1) - F_{\overline{s^L}}(k)\right) \right\rceil + 1$$

et le débit total R_{tot} moyen nécessaire à encoder cette séquence vérifie donc :

$$1 + H\left(\overline{\boldsymbol{s}^L}\right) \leq R_{\text{tot}} \leq 2 + H\left(\overline{\boldsymbol{s}^L}\right).$$

Dans la mesure où tous les éléments de $\overline{\boldsymbol{s}^L}$ sont supposés indépendants, on a :

$$\frac{1}{L} + \frac{1}{L}\sum_{l=1}^{L} H(\overline{s_l}) \leq \frac{R_{\text{tot}}}{L} \leq \frac{2}{L} + \frac{1}{L}\sum_{l=1}^{L} H(\overline{s_l}).$$

Si la distribution de tous les $\overline{\boldsymbol{s}_l}$ est identique, on voit que le débit moyen $R = \frac{R_{\text{tot}}}{L}$ nécessaire à l'encodage de chaque élément de la séquence vérifie asymptotiquement :

$$\lim_{L\to\infty} R = H(\overline{s}).$$

En d'autres termes, un encodeur arithmétique permet asymptotiquement d'atteindre la limite théorique 13.1.3 pour l'encodage de séquences.

De manière intéressante, il n'est pas nécessaire de supposer que tous les éléments $\overline{\boldsymbol{s}_l}$ de la séquence sont des réalisations de la même variable aléatoire, mais juste qu'ils sont indépendants.
Dans le cas où chaque élément $\overline{\boldsymbol{s}_l}$ de la séquence est une réalisation d'une variable aléatoire $\overline{s_l}$ dont on connaît la loi $P_l(\cdot)$, on montre [224] que le débit effectif moyen R_l pour chaque symbole est de :

$$R_l = -\log_2 P_l(\overline{\boldsymbol{s}_l}). \tag{13.1.10}$$

Comme on le voit, le codage arithmétique devient très intéressant lorsqu'on considère l'encodage d'une longue séquence $\overline{\boldsymbol{s}^L}$ de réalisations indépendantes de variables aléatoires discrètes, chacune définie par une loi de probabilité connue.

Pour une raison de concision de mon exposé, je ne précise pas la manière d'implémenter une telle procédure d'encodage, qui présente des enjeux pratiques importants liés à la précision finie des calculs menés sur un ordinateur. En effet, la probabilité $P\left(\overline{\boldsymbol{s}^L}\right)$ de chaque séquence possible est extrêmement faible. De telles difficultés peuvent cependant être surmontées parfaitement en pratique par des astuces de remise à l'échelle et l'encodage arithmétique est couramment utilisé dans des applications [224, 158, 159, 161].

Pour finir, je résume les avantages et les inconvénients du codage arithmétique dans le tableau 13.2 et cela conclut cette présentation du codage de sources discrètes.

Les avantages d'un encodage arithmétique
- Il permet l'encodage d'une séquence $\overline{\boldsymbol{s}^L} = [\overline{\boldsymbol{s}}_1, \ldots, \overline{\boldsymbol{s}}_L]$ de réalisations indépendantes de variables aléatoires de lois *différentes* $P_l(\cdot)$.
- On montre que cet encodage peut se faire de manière séquentielle, où on fournit à l'algorithme chaque symbole $\overline{\boldsymbol{s}}_l$ de la séquence ainsi que la loi $P_l(\cdot)$ de la variable aléatoire dont il est une réalisation.
- Asymptotiquement, le débit effectif R_l pour l'encodage de chaque symbole d'une séquence $\overline{\boldsymbol{s}^L}$ est donné en bits par :
$$R_l = -\log_2 P(\overline{\boldsymbol{s}}_l).$$

Les inconvénients d'un encodage arithmétique :
- Il nécessite la connaissance de la distribution $P_l(\cdot)$ de chaque symbole.
- Les procédures d'encodage et de décodage sont délicates à implémenter.

TABLE 13.2: Avantages et inconvénients d'un encodage arithmétique

13.2 Théorie débit-distorsion

13.2.1 Entropie différentielle et information mutuelle

La notion d'entropie peut se généraliser au cas où la variable aléatoire considérée s est à valeurs dans $\mathbb{R}^J$ et non pas dans un ensemble discret $\mathcal{C}_N$. La notion équivalente dans ce cas est celle d'*entropie différentielle*. Supposons une fois de plus qu'on connaisse la densité de probabi-

lité $p(\boldsymbol{s} \mid \theta)$ de s. L'entropie différentielle de s, notée $h(s)$, est donnée par :

$$\begin{aligned} h(s) &= -\int_{\boldsymbol{s}\in\mathbb{R}^J} p(\boldsymbol{s} \mid \theta) \log p(\boldsymbol{s} \mid \theta)\, d\boldsymbol{s} \\ &= -\mathbb{E}_s\left[\log p(\boldsymbol{s} \mid \theta)\right]. \end{aligned} \tag{13.2.1}$$

Compte tenu du fait que je me concentrerai bientôt exclusivement sur le cas des densités gaussiennes pour lequel 13.2.1 est défini, je supposerai que cette intégrale existe. J'utiliserai souvent le terme d'entropie pour désigner l'entropie différentielle lorsque le contexte est clair. Comme on le voit, la définition 13.2.1 de l'entropie différentielle de variables à valeurs continues est similaire à celle 13.1.1 de l'entropie de variables discrètes. En particulier, on peut montrer qu'*une translation laisse l'entropie différentielle inchangée.*

Cependant, leurs propriétés diffèrent significativement. En particulier, l'entropie différentielle d'une variable aléatoire continue peut être négative. De plus, le théorème du codage de source 13.1.3 ne se généralise pas tel quel dans le cas continu.

On peut montrer que l'entropie d'une variable aléatoire distribuée selon une loi gaussienne multivariée de matrice de covariance K est donnée (en bits) par :

$$h(s) = \frac{1}{2}\log_2\left((2\pi e)^J |K|\right), \tag{13.2.2}$$

où $|K|$ est le déterminant de K.

L'information mutuelle $I(s,x)$ entre deux variables aléatoires continues à valeurs dans $\mathbb{R}^J$ se définit de la même manière qu'en 13.1.6 :

$$I(s,x) = h(s) - h(s \mid x). \tag{13.2.3}$$

Une propriété intéressante de cette quantité est qu'elle est toujours positive, même dans le cas de variables aléatoires continues :

$$I(s,x) \geq 0.$$

L'information mutuelle $I(s,x)$ entre deux variables s et x dans le cas continu peut être interprétée de la même manière que dans le cas discret comme quantifiant la réduction moyenne d'incertitude qu'apporte l'observation de l'une sur l'autre. Pour le montrer, je vais considérer l'exemple suivant, important pour la suite de cet exposé.

Exemple. Soient s et x deux variables aléatoires conjointement gaussiennes et corrélées, telle que x est la somme de s avec bruit additif ϵ gaussien :

$$x = s + \epsilon.$$

La distribution jointe de s et x est donnée par :

$$\begin{bmatrix} \boldsymbol{s} \\ \boldsymbol{x} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_s^2 & \sigma_s^2 \\ \sigma_s^2 & \left(\sigma_s^2 + \sigma_\epsilon^2\right) \end{bmatrix}\right),$$

On peut utiliser les résultats 2.2.8 exploités déjà de très nombreuses fois dans cet exposé pour montrer que la distribution *a posteriori* de $\boldsymbol{s}$ étant donné $\boldsymbol{x}$ est :

$$\boldsymbol{s} \mid \boldsymbol{x} \sim \mathcal{N}\left(\frac{\sigma_s^2}{\sigma_s^2 + \sigma_\epsilon^2}\boldsymbol{x},\ \frac{\sigma_s^2 \sigma_\epsilon^2}{\sigma_s^2 + \sigma_\epsilon^2}\right).$$

Calculons à présent l'information mutuelle $I(s,x)$ entre s et x en utilisant l'expression 13.2.2 de l'entropie différentielle d'une variable aléatoire gaussienne :

$$\begin{aligned} I(s,x) &= h(s) - h(s \mid x) \\ &= \left(\frac{1}{2}\log 2\pi e\sigma_s^2\right) - \left(\frac{1}{2}\log 2\pi e\frac{\sigma_s^2\sigma_\epsilon^2}{\sigma_s^2+\sigma_\epsilon^2}\right) \qquad (13.2.4) \\ &= -\frac{1}{2}\log\left(\frac{\sigma_\epsilon^2}{\sigma_s^2+\sigma_\epsilon^2}\right), \qquad (13.2.5) \end{aligned}$$

où j'ai utilisé en 13.2.4 le fait que l'entropie différentielle est invariante par translation. Comme on le voit à l'examen de 13.2.5, si la variance σ_ϵ^2 du bruit additif devient très grande devant celle de s, alors l'information mutuelle entre s et x tend vers 0. Dans le cas contraire, elle grandit jusqu'à devenir infinie si $x = s$.

13.2.2 fonction débit-distorsion

Soit s une variable aléatoire à valeurs dans $\mathbb{R}^J$ dont on connaît la densité de probabilité $p(\boldsymbol{s} \mid \Theta)$ et soit

$$\boldsymbol{s}^L = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_L]$$

un ensemble de L réalisations indépendantes de cette variable aléatoire, c'est-à-dire tel que :

$$p\left(\boldsymbol{s}^L \mid \Theta\right) = \prod_{l=1}^{L} p(\boldsymbol{s}_l \mid \Theta).$$

Supposons qu'un observateur connaisse cet ensemble $\boldsymbol{s}^L$ de L réalisations indépendantes de s et qu'il souhaite transmettre ces valeurs à un tiers. Dans la mesure où chacune de ces variables est *réelle*, il n'est pas possible dans le cas général de le faire sans erreur en un temps fini [5]. C'est ce qui explique le fait que le cas des variables aléatoires continues ne se règle pas de la même manière que celui des variables discrètes par un théorème de codage de source sans perte analogue à 13.1.3.

Cependant, des solutions apparaissent si on est prêt à accepter une *distorsion* dans la reconstruction du message transmis. Par exemple, on peut considérer que pour l'application envisagée, il sera suffisant de connaître les 10 premières décimales de chaque $\boldsymbol{s}_l$ plutôt que sa valeur réelle. Dans ce cas, la situation devient radicalement différente, puisqu'on est prêt à se contenter d'une reconstruction

$$\overline{\boldsymbol{s}^L} = [\overline{\boldsymbol{s}_1}, \ldots, \overline{\boldsymbol{s}_L}]$$

qui ne prend qu'un nombre fini de valeurs possibles (ici, 10^{11L} pour L variables comprises entre 0 et 1 dont on souhaite les 10 premières décimales).

Ces considérations justifient d'introduire une opération de *quantification*, par laquelle on associe au message initial $\boldsymbol{s}^L$ à valeurs dans $\left(\mathbb{R}^J\right)^L$ un message *quantifié* $\overline{\boldsymbol{s}^L}$, qui prend ses valeurs dans un ensemble discret $\mathcal{C}_J^L$ de points de reconstructions, avec $\mathcal{C}_J \subset \mathbb{R}^J$ un ensemble discret de réels.

Présenté de cette manière, l'objectif de la *théorie débit-distorsion* devient de déterminer quel débit on peut espérer atteindre pour le transfert de $\overline{\boldsymbol{s}^L}$ si on se donne une distorsion moyenne maximale à respecter entre le message original et sa reconstruction. Pour formaliser ces deux objectifs contradictoires, je vais préciser successivement ce qu'on entend par distorsion et par débit :

– On introduit une *fonction de distorsion* $d^L\left(\boldsymbol{s}^L, \overline{\boldsymbol{s}^L}\right)$ qui quantifie l'écart entre le message original $\boldsymbol{s}^L$ à transmettre et sa reconstruction $\overline{\boldsymbol{s}^L}$. Il s'agit d'une grandeur positive qui s'annule si ses deux opérandes sont égales. On peut définir des fonctions de distorsion de nombreux types, mais il est classique de se limiter à celles qui s'expriment sous la forme :

$$d^L\left(\boldsymbol{s}^L, \overline{\boldsymbol{s}^L}\right) = \sum_{l=1}^{L} d(\boldsymbol{s}_l, \overline{\boldsymbol{s}_l}),$$

5. On connaît le temps qu'il faut pour énumérer toutes les décimales de π...

c'est-à-dire qui peuvent se décomposer comme la somme des L distorsions individuelles des échantillons du message. Le premier objectif de l'opération de quantification sera de garantir que $\lim_{L\to\infty} \frac{1}{L} d^L\left(\boldsymbol{s}^L, \overline{\boldsymbol{s}^L}\right)$ soit inférieur à une certaine distorsion D donnée :

$$\lim_{L\to\infty} \frac{1}{L} d^L\left(\boldsymbol{s}^L, \overline{\boldsymbol{s}^L}\right) \leq D \tag{13.2.6}$$

– Par ailleurs, on appellera *débit* le nombre moyen de bits par échantillon nécessaire à la transmission de $\overline{\boldsymbol{s}^L}$. Si R_{tot} est le débit total, on définira le débit moyen par symbole par $R = \frac{R_{\text{tot}}}{L}$. Le deuxième objectif de l'opération de quantification sera de permettre un débit R minimal.

Étant donnée la distribution $p(\boldsymbol{s} \mid \Theta)$ de la source, on définira la fonction débit-distorsion $R(D)$ comme indiquant pour chaque distorsion D le débit moyen par symbole minimal requis asymptotiquement pour transmettre une séquence de réalisations indépendantes avec une distorsion moyenne inférieure ou égale à D.

Pour définir formellement cette fonction débit-distorsion [120, 89], il est nécessaire de définir l'étape de quantification par laquelle on assigne un échantillon quantifié $\overline{\boldsymbol{s}}$ à une réalisation $\boldsymbol{s}$ de la source. En toute généralité, elle peut se définir comme une probabilité conditionnelle $p(\overline{\boldsymbol{s}} \mid \boldsymbol{s})$. Au cas où une reconstruction $\overline{\boldsymbol{s}}$ est associée de manière déterministe à une réalisation $\boldsymbol{s}$ selon :

$$\overline{\boldsymbol{s}} = \mathcal{Q}\{\boldsymbol{s}\}, \tag{13.2.7}$$

cette probabilité devient triviale : $p(\overline{\boldsymbol{s}} \mid \boldsymbol{s}) = \delta(\overline{\boldsymbol{s}}, \mathcal{Q}\{\boldsymbol{s}\})$.

A présent, si on désigne par $B(D)$ l'ensemble des quantifications qui permettent une distorsion *moyenne* inférieure à D :

$$B(D) = \{p(\overline{\boldsymbol{s}} \mid \boldsymbol{s}) \mid \mathbb{E}[d(\overline{\boldsymbol{s}} \mid \boldsymbol{s})] \leq D\}, \tag{13.2.8}$$

alors le théorème du débit-distorsion indique que :

$$R(D) = \inf_{p(\overline{\boldsymbol{s}}|\boldsymbol{s}) \in B(D)} I(s, \overline{s}). \tag{13.2.9}$$

Pour une quantification $p(\overline{\boldsymbol{s}} \mid \boldsymbol{s})$ donnée, si elle appartient à $B(D)$, elle produit par construction 13.2.8 une distorsion moyenne inférieure à D. Par ailleurs, les reconstructions $\overline{\boldsymbol{s}}$ qu'elle produit ne peuvent pas être transmis avec un débit moyen inférieur à $I(s, \overline{s})$. En effet, le théorème du codage de source 13.1.3 indique que ce débit est minoré par l'entropie $H(\overline{s})$. Or, cette entropie vérifie :

$$\begin{aligned} H(\overline{s}) &\geq H(\overline{s}) - H(\overline{s} \mid s) \\ &= I(s, \overline{s}) \end{aligned} \tag{13.2.10}$$

et $I(s, \overline{s})$ constitue donc bien une borne inférieure au débit possible. Dans le cas où la quantification est déterministe 13.2.7, on a $H(\overline{s} \mid s) = 0$ et cette borne est stricte. La réciproque est en revanche moins immédiate, qui établit que $R(D)$ constitue une borne inférieure au débit moyen nécessaire à l'encodage de séquences asymptotiquement longues. J'admettrai ce résultat fort de la théorie du codage de source. On démontre par ailleurs que la fonction de débit-distorsion est décroissante en fonction de D.

Un autre résultat très important que j'admettrai aussi est que si elle est une borne inférieure aux débits qu'on peut espérer pour une distorsion donnée, on démontre qu'il existe des procédures déterministes de quantification du type de 13.2.7 capables de l'atteindre. Malheureusement, les preuves impliquées ne sont pas constructives et la mise au point de procédures de quantification permettant de se rapprocher de la borne reste un domaine de recherche important. De plus, ces résultats tiennent surtout asymptotiquement, c'est-à-dire lorsque la longueur L des séquences à coder devient suffisante.

Une autre grandeur théorique importante reliée à la fonction débit-distorsion $R(D)$ est la fonction distorsion-débit $D(R)$: elle indique pour un débit R donné quelle distorsion minimale on peut espérer atteindre asymptotiquement par une procédure de quantification. On montre que $D(R)$ est simplement l'inverse de $R(D)$ dès lors que $R(D)$ est strictement décroissante [120].

13.2.3 Le cas gaussien scalaire pour une distorsion quadratique

La fonction débit-distorsion 13.2.9 définit une borne inférieure au débit qu'on peut espérer atteindre asymptotiquement pour le codage de séquences de réalisations indépendantes d'une variable aléatoire s continue, dont on connaît la densité de probabilité $p(\boldsymbol{s} \mid \Theta)$. Comme le suggère sa définition 13.2.9, elle ne s'exprime en général pas sous la forme d'une expression analytique simple.

Il y a cependant certains cas où une expression simple de la fonction débit-distorsion est disponible. Fort opportunément pour nous, c'est ce qui se passe si s est une variable gaussienne et si le critère de distorsion d est l'erreur quadratique. Puisque c'est précisément le cas pratique qui va nous intéresser, je donne ici les résultats correspondants.

Dans le cas d'une variable aléatoire gaussienne scalaire ($J = 1$) de variance σ^2, on montre [120] que la fonction débit-distorsion prend une forme simple, ainsi que la fonction distorsion-débit :

$$R(D) = \begin{cases} \frac{1}{2}\log\left(\frac{\sigma^2}{D}\right) & \text{si } D \leq \sigma^2 \\ 0 & \text{si } D \geq \sigma^2 \end{cases} \tag{13.2.11}$$

$$D(R) = \sigma^2 \exp(-2R). \tag{13.2.12}$$

On constate que le débit moyen nécessaire pour transmettre la valeur d'une variable gaussienne avec une distorsion supérieure à sa variance est nul. Ceci se comprend aisément dans la mesure où la variance se définit comme la distorsion de la variable autour de sa moyenne. Si $D \geq \sigma^2$, il suffit de toujours choisir la moyenne comme reconstruction et on obtient une quantification qui respecte la contrainte. Aucun débit n'est alors nécessaire, si on suppose moyenne et variance connues au décodeur. Les fonctions $R(D)$ et $D(R)$ d'une variable gaussienne scalaire sont représentées en échelle semi-logarithmique sur la figure 13.2.

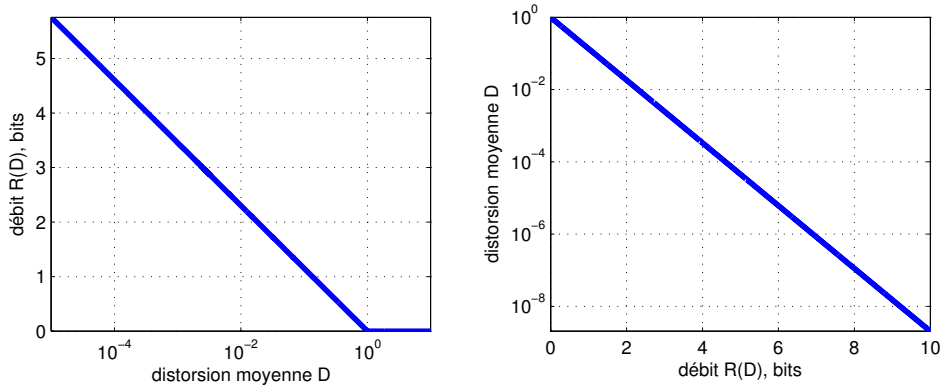

FIGURE 13.2: fonctions débit-distorsion $R(D)$ (à gauche) et distorsion-débit $D(R)$ (à droite) pour une variable aléatoire gaussienne scalaire de variance $\sigma^2 = 1$, avec une échelle logarithmique pour D. On a simplement $D(R) = \frac{1}{R(D)}$. Il est possible en théorie d'atteindre asymptotiquement n'importe quel couple (D, R) au-dessus de ces fonctions.

13.2.4 Le cas vectoriel gaussien pour une distorsion quadratique

Considérons à présent le cas d'une variable gaussienne vectorielle $s = [s_1, \ldots, s_J]^\top$. Je suppose pour commencer que tous les s_j sont indépendants, de variance σ_j^2. On montre [120] que dans ce cas, la fonction de débit-distorsion $R(D)$ s'obtient par :

$$R = \sum_{j=1}^{J} R_j = \sum_{j=1}^{J} \max\left(0, \frac{1}{2}\log\frac{\sigma_j^2}{D}\right) \tag{13.2.13}$$

Ces relations s'interprètent en disant que pour une variable s_j, si sa variance est inférieure à D, alors il ne sera pas nécessaire de dépenser du débit pour la transmettre puisque reconstruire sa moyenne sera suffisant. Pour les variables telles que $\sigma_j^2 > D$, le débit est réparti en fonction de $\frac{\sigma_j^2}{D}$: plus la variance d'une variable sera grande devant D, plus il sera nécessaire de lui attribuer du débit pour la transmettre correctement.

On établit une expression analytique simple pour la fonction débit-distorsion dans le cas de variables aléatoires gaussiennes multivariées.

Dans le cas où les variables s_j sont corrélées, la distribution jointe $p(\boldsymbol{s} \mid \Theta)$ est une gaussienne multivariée dont je note K la matrice de covariance. On peut se ramener au cas précédent en considérant non pas la quantification directe des $\boldsymbol{s}_j$, mais plutôt celle de la transformée de Karhunen-Loeve de $\boldsymbol{s}$. Plus précisément, puisque la matrice K est définie positive, on peut en effectuer une décomposition du type :

$$K = U \Lambda U^H,$$

où U est une matrice orthonormée et $\Lambda = \operatorname{diag}[\lambda_1, \ldots, \lambda_J]$ est diagonale. On définit alors la transformée de Karhunen-Loeve de $\boldsymbol{s}$ par :

$$\mathcal{K}(\boldsymbol{s}) = U^H \boldsymbol{s},$$

et on vérifie que ses éléments sont décorrélés, donc indépendants ici puisque $\boldsymbol{s}$ est gaussien :

$$\begin{aligned} \mathbb{E}\left[\mathcal{K}(\boldsymbol{s})\,\mathcal{K}(\boldsymbol{s})^H\right] &= \mathbb{E}\left[U^H \boldsymbol{s}\boldsymbol{s}^H U\right] \\ &= U^H \mathbb{E}\left[\boldsymbol{s}\boldsymbol{s}^H\right] U \\ &= U^H U \Lambda U^H U \\ &= \Lambda. \end{aligned}$$

Or, on a vu en section 10.3.2 qu'une transformée orthonormale laisse la distorsion quadratique moyenne inchangée. Donc on se ramène au cas précédent en quantifiant $\mathcal{K}(\boldsymbol{s})$ au lieu de $\boldsymbol{s}$ et en utilisant les valeurs propres λ_j en lieu et place des variances individuelles σ_j^2.

13.2.5 Conclusion

La théorie débit-distorsion permet d'identifier le débit minimal théorique requis pour transmettre une variable quantifiée présentant avec le message original une distorsion moyenne maximale donnée. À contrario, elle permet aussi d'obtenir la distorsion moyenne minimale qu'on peut espérer atteindre à débit fixé.

Dans le cas gaussien, on dispose d'une expression analytique simple de ces deux fonctions débit-distorsion $R(D)$ et distorsion-débit $D(R)$, indiquant les bornes sur les performances de codage qu'on peut espérer atteindre. C'est précisément ce cas gaussien qui sera exploité dans la suite pour le système que je propose de codage informé par les mélanges.

En effet, on verra au chapitre suivant que ces considérations permettent de fournir une réponse élégante à la question du débit minimal nécessaire à la récupération au décodeur de signaux qui soient d'une qualité donnée. On peut d'ores et déjà sentir que cette qualité ne sera plus bornée par des estimateurs oracle, mais que la distorsion pourra au contraire être rendue arbitrairement petite, pour peu que le débit disponible soit suffisant. Ceci contraste nettement avec les limitations de l'approche paramétrique présentée en partie III.

13.3 Théorie haute-résolution

La théorie débit-distorsion présentée en section 13.2 fournit des bornes absolues sur les performances qu'on peut espérer d'un système de codage de variables aléatoires continues. En ce sens, elle est intéressante pour les applications car elle permet de déterminer le régime de fonctionnement limite qu'il est théoriquement possible d'atteindre, indépendamment d'une quelconque procédure de quantification adoptée, potentiellement sous-optimale. Ainsi, une approche pourra bénéficier de

bornes théoriques très intéressantes sans qu'un système de codage opérationnel soit encore disponible qui permette de les atteindre. Une telle théorie permet en somme d'identifier des directions intéressantes de recherche.

Cependant, lorsqu'il s'agit de mettre au point des stratégies concrètes d'encodage, la théorie débit-distorsion s'avère moins intéressante. Bien qu'elle garantisse qu'il est possible asymptotiquement d'atteindre les bornes qu'elle précise en utilisant une procédure déterministe de quantification, elle n'indique pas une manière concrète de le faire. On souhaiterait disposer d'une théorie qui nous indique des procédures effectives de quantification qui permettent à la fois de se rapprocher de la fonction débit-distorsion et de s'affranchir de la considération de séquences asymptotiquement grandes.

13.3.1 Motivations et simplifications

La *théorie haute-résolution* (ou hypothèse haute résolution) fournit une approche simple pour déterminer des approximations utiles de la fonction débit-distorsion en même temps qu'une méthode concrète pour l'atteindre si les débits considérés sont suffisamment élevés. Je considèrerai ici le seul cas d'une quantification scalaire, c'est-à-dire appliquée à la réalisation $\boldsymbol{s}$ d'une variable aléatoire s à valeurs dans $\mathbb{R}$ dont on connaît la densité de probabilité $p(\boldsymbol{s} \mid \Theta)$. De plus, je me limiterai à la distorsion quadratique.

La théorie haute-résolution se concentre sur un type particulier de quantification, dite *quantification régulière* et illustrée en figure 13.3.

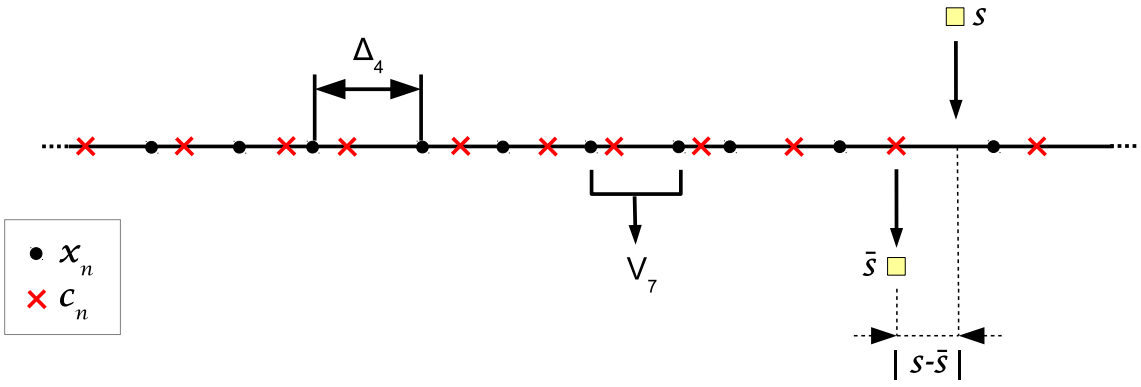

FIGURE 13.3: Quantification régulière : $\mathbb{R}$ est découpé en N intervalles adjacents V_n. Pour chacun, on choisit un point de reconstruction c_n. La quantification $\bar{\boldsymbol{s}} = \mathcal{Q}\{\boldsymbol{s}\}$ d'une observation est le point de reconstruction de l'intervalle auquel appartient $\boldsymbol{s}$. Ici, $\mathcal{Q}\{\boldsymbol{s}\} = c_{10}$.

Tout d'abord, on se donne une suite croissante $\{x_n\}$ de $N+1$ réels, avec $x_0 = -\infty$ et $x_N = +\infty$ qui définit N intervalles V_n adjacents, aussi appelés *cellules* :

$$\forall n = 1, \ldots, N,\ V_n = [x_{n-1}\,;\, x_n]\,,$$

et on note $\Delta_n = x_n - x_{n-1}$ la longueur de chacune de ces cellules. Ensuite, pour chaque cellule, on choisit un réel $c_n \in V_n$ appelé point de reconstruction.

Une quantification régulière $\mathcal{Q}$ consiste à assigner à une observation $\boldsymbol{s}$ quelconque le point de reconstruction c_k de la cellule à laquelle elle appartient. On note

$$\bar{\boldsymbol{s}} = Q\{\boldsymbol{s}\}$$

La loi de probabilité de $\bar{s}$ peut se calculer à partir de celle de s :

$$P(\bar{\boldsymbol{s}} = c_n) = \int_{\boldsymbol{s} \in V_n} p(\boldsymbol{s} \mid \Theta)\, d\boldsymbol{s}. \tag{13.3.1}$$

Le débit moyen minimal R requis pour encoder $\overline{\boldsymbol{s}}$ obéit au théorème du codage de source 13.1.3, puisqu'il s'agit d'une variable aléatoire discrète.

Étant donnée une quantification régulière $\mathcal{Q}$, l'hypothèse fondamentale de la théorie haute-résolution est qu'au sein de chaque cellule n donnée, la densité de probabilité $p(\boldsymbol{s} \mid \Theta)$ reste constante :

$$\forall \boldsymbol{s} \in V_n,\, p(\boldsymbol{s} \mid \Theta) \approx p(c_n \mid \Theta).$$

On peut interpréter cette hypothèse en voyant qu'elle devient valide lorsque les cellules sont petites devant les variations de $p(\boldsymbol{s} \mid \Theta)$, et donc nombreuses. Sous cette hypothèse, de nombreux résultats très intéressants sont démontrés [120] :

– La distorsion D_n minimale par cellule est atteinte si c_n est le centre de la cellule V_n
– D_n dépend uniquement de la longueur Δ_n de la cellule n (sauf pour $n = 1$ et $n = N$ où $D_n = \infty$) :

$$D_n = \frac{\Delta_n^2}{12} \tag{13.3.2}$$

– Si on veut minimiser l'entropie $H(\overline{s})$, (*entropie contrainte*)

1. il faut choisir un pas de quantification constant

$$\forall n,\, \Delta_n = \Delta. \tag{13.3.3}$$

2. l'entropie $H(\overline{s})$ est liée à Δ par :

$$\Delta = 2^{-H(\overline{s})+h(s)}, \tag{13.3.4}$$

3. ce qui donne en utilisant 13.3.2 :

$$H(\overline{s}) = h(s) - \frac{1}{2}\log 12D, \tag{13.3.5}$$

où $H(\overline{s})$ est directement relié au débit nécessaire à l'encodage de $\overline{s}$ en vertu du théorème de codage de source 13.1.3.

Chercher à minimiser l'entropie $H(\overline{s})$ des réalisations quantifiées revient à minimiser le débit nécessaire pour leur transmission. En conséquence, sous hypothèse de haut-débit, cette configuration dite à *entropie contrainte* démontre que c'est une quantification uniforme qui permet de minimiser à la fois la distorsion et le débit d'une variable aléatoire scalaire de densité de probabilité quelconque. Ce résultat sera central dans la suite de mon exposé, puisqu'il fournit une stratégie très simple pour procéder à l'encodage.

On peut noter que dans le cas gaussien, on a d'après 13.2.11 et 13.3.5 :

$$\begin{aligned} H(\overline{s}) &= \frac{1}{2}\log\left(\frac{\sigma^2}{D}\right) + \frac{1}{2}\log\frac{\pi e}{6} \\ &\approx R(D) + 0.25. \end{aligned} \tag{13.3.6}$$

La fonction débit-distorsion *sous hypothèse haute-résolution* et en utilisant une quantification scalaire est supérieure d'environ 0.25bits à la fonction théorique $R(D)$.

Souvent, on cherche plutôt en pratique à optimiser les performances à bas débit. On pourrait alors s'interroger sur l'intérêt d'une approximation qui tient lorsqu'ils sont élevés. Il est cependant établi expérimentalement que les procédures d'encodage correspondant à l'hypothèse de haute-résolution donnent des performances effectives très bonnes, même à bas débit [224, 158, 159].

13.3.2 Encodage d'une séquence de vecteurs gaussiens

De manière à illustrer l'intérêt de l'hypothèse haute-résolution, je vais mettre en pratique les résultats ci-dessus de quantification à entropie contrainte au cas gaussien qui m'occupera durant toute la suite de mon exposé.

La théorie haute-résolution est appliquée à l'encodage optimal d'une séquence de L vecteurs gaussiens indépendants dont on connaît les moyennes et matrices de covariance.

Supposons qu'on observe une séquence $\boldsymbol{s}^L$ de L réalisations indépendantes de variables gaussiennes multivariées :

$$\boldsymbol{s}^L = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_L\},$$

où chaque $\boldsymbol{s}_l$ est un vecteur de dimension $J \times 1$ dont on connaît la moyenne $\boldsymbol{\mu}_l$ et la matrice de covariance K_l, de dimension $J \times 1$ et $J \times J$, respectivement :

$$\boldsymbol{s}_l \sim \mathcal{N}(\boldsymbol{\mu}_l, K_l).$$

L'objectif est d'encoder $\boldsymbol{s}^L$ de manière à garantir une distorsion moyenne inférieure à D et à minimiser le débit moyen nécessaire pour transmettre cette information.

Sous hypothèse de haute-résolution, la méthode à adopter dans le cas $J = L = 1$ est d'effectuer une quantification uniforme, comme le suggère la discussion plus haut. En sortie, on obtient une valeur quantifiée, dont l'entropie est donnée par 13.3.6. Il sera dur en pratique d'encoder de manière optimale cette seule réalisation.

Dans le cas $J > 1$ et $L > 1$, c'est-à-dire où chaque $\boldsymbol{s}_l$ est un vecteur, on peut appliquer la même stratégie qu'en 13.2.4 pour en décorréler les composantes en utilisant la transformée de Karhunen-Loeve. Cette approche conduit à l'obtention de JL variables gaussiennes indépendantes.

Même si des variables aléatoires sont indépendantes, la manière optimale de les quantifier est d'utiliser une quantification jointe, c'est-à-dire vectorielle [158]. Ce résultat de la théorie de la quantification est illustré en figure 13.4. Cependant, la mise en place d'une telle procédure est difficile en pratique et on choisit souvent d'avoir plutôt recours à une quantification indépendante de chacune des variables.

Même si des variables aléatoires sont indépendantes, la manière optimale de les quantifier est d'utiliser une quantification vectorielle. Ici, j'utiliserai une stratégie sous-optimale classique qui consiste à utiliser des quantifications scalaires indépendantes.

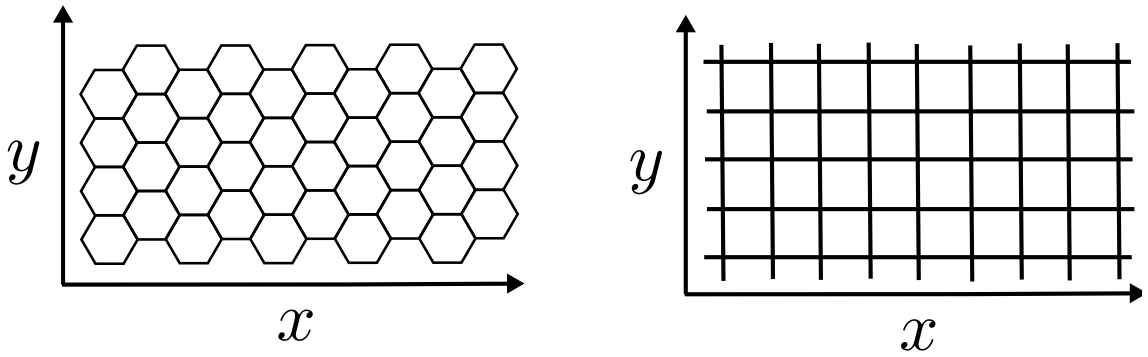

FIGURE 13.4: Pour deux variables aléatoires indépendantes x et y, la procédure optimale de quantification est vectorielle, telle que représentée à gauche. La procédure qui consiste à quantifier chaque variable indépendamment, comme représenté à droite, est sous optimale. Elle est cependant très simple à implémenter et donc couramment utilisée en pratique.

Si on décide de procéder à une quantification indépendante de chacune des composantes après transformée de Karhunen-Loeve, alors sous hypothèse de haute-résolution et à entropie contrainte, la stratégie optimale pour le faire est d'encoder chacune de ces variables par un quantificateur uniforme.

Dans tous les cas, après quantification, on obtient JL variables quantifiées (et donc discrètes), dont on peut calculer analytiquement la loi de probabilité. La situation devient alors propice à l'utilisation d'un encodeur arithmétique pour coder l'ensemble de cette séquence et le débit correspondant est très proche des bornes théoriques pour peu que JL soit grand, ce qui sera souvent le cas en pratique. Ces opérations sont résumées dans l'algorithme 13.1 page suivante, qui a déjà été présenté plusieurs fois dans la littérature [224, 158, 159, 161].

Algorithme 13.1 Algorithme optimal de quantification à entropie contrainte d'une séquence de L réalisations indépendantes de variables aléatoires gaussiennes multivariées dont moyennes et covariances sont connues au codeur et décodeur, sous hypothèse de haute-résolution et d'utilisation d'une quantification scalaire.

Entrées

– Séquence $\boldsymbol{s}^L$:

$$\boldsymbol{s}^L = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_L\}$$

où chaque $\boldsymbol{s}_l$ est un vecteur de dimension $J \times 1$.

– Vecteur moyenne $\boldsymbol{\mu}_l$ et matrice de covariance K_l caractérisant la distribution gaussienne de chaque vecteur $\boldsymbol{s}_l$

– Distorsion moyenne maximale D souhaitée, ou alternativement le pas de quantification $\Delta = \sqrt{12D}$ d'après 13.3.2.

Pour chaque $l = 1, \ldots, L$

– Calculer la décomposition de K_l en valeurs propres :

$$K_l = U_l \mathrm{diag}\left(\left[\lambda_{l1}, \ldots, \lambda_{lJ}\right]\right) U_l^H$$

– Calculer la transformée de Karhunen-Loeve $\boldsymbol{\xi}_l$ de $\boldsymbol{s}_l - \boldsymbol{\mu}_l$:

$$\boldsymbol{\xi}_l = U_l^H \left(\boldsymbol{s}_l - \boldsymbol{\mu}_l\right)$$

Sa $j^{\text{ème}}$ entrée $\boldsymbol{\xi}_{lj}$ est distribuée selon une loi gaussienne centrée de variance λ_{lj} :

$$\boldsymbol{\xi}_{lj} \sim \mathcal{N}\left(0, \lambda_{lj}\right).$$

– **Pour chaque** $j = 1, \ldots, J$:

– Quantifier $\boldsymbol{\xi}_{lj}$ en utilisant une quantification uniforme de pas de quantification Δ, pour produire $\overline{\boldsymbol{\xi}}_{lj}$

$$\overline{\boldsymbol{\xi}}_{lj} = \mathcal{Q}\left\{\boldsymbol{\xi}_{lj}\right\}.$$

Je désigne par $\mathcal{C}_N = \{c_n\}_{n\in\mathbb{N}}$ les points de reconstruction correspondants.

– La loi de probabilité $P_{lj}(\cdot)$ de la variable discrète $\overline{\boldsymbol{\xi}}_{lj}$ est donnée par 13.3.1 :

$$\forall n,\ P_{lj}(c_n) = \int_{c_n - \frac{\Delta}{2}}^{c_n + \frac{\Delta}{2}} \mathcal{N}\left(\xi \mid 0, \lambda_{lj}\right) d\xi, \tag{13.3.7}$$

Encodage arithmétique

– $\overline{\boldsymbol{\xi}}$ est une séquence de LJ réalisations indépendantes $\overline{\boldsymbol{\xi}}_{lj}$ de LJ variables discrètes dont on connait les LJ lois $P_{lj}(\cdot)$. On peut appliquer un encodage arithmétique sans perte pour l'encoder, vu en section 13.1.3.

– Le débit effectif moyen pour chaque symbole $\overline{\boldsymbol{\xi}}_{lj}$ est de $-\log_2 P_{lj}\left(\overline{\boldsymbol{\xi}}_{lj}\right)$, donné en 13.3.7. Le débit total est donné, en bits, par *la fonction débit-distorsion opérationnelle* [122, 159] :

$$R_s(D) = -\log_2 p\left(\boldsymbol{s}^L\right) - \frac{LJ}{2}\log_2 12D \tag{13.3.8}$$

Reconstruction après décodage

– Après décodage, on dispose de tous les $\overline{\boldsymbol{\xi}}_{lj}$.

– La reconstruction $\overline{\boldsymbol{s}_l}$ de chaque $\boldsymbol{s}_l$ est donnée par

$$\overline{\boldsymbol{s}_l} = U_l \overline{\boldsymbol{\xi}}_l + \boldsymbol{\mu}_l \tag{13.3.9}$$

et présente une distorsion inférieure à D.

13.3.3 Applications pour le codage audio

L'algorithme présenté plus haut permet un encodage optimal d'une séquence de vecteurs gaussiens. Pour conclure ce chapitre sur le codage de source, je vais à présent montrer brièvement comment l'algorithme 13.1 peut être utilisé en pratique pour l'encodage de signaux audio.

Soit $\tilde{\boldsymbol{s}}(\cdot)$ un signal audio, qu'on modélise comme un PGLS. Sa TFCT $\boldsymbol{s}(f,n)$ est une matrice de dimension $F \times N$. Pour chaque point (f,n), $\boldsymbol{s}(f,n)$ est une variable gaussienne scalaire (de dimension 1×1) et on a vu en section 3.3 que tous les $\{\boldsymbol{s}(f,n)\}_{f,n}$ sont indépendants dans le modèle PGLS.

Supposons à présent qu'un modèle de source $\mathcal{P}(\cdot \mid \theta)$ soit disponible, qui permette de poser [6] :

$$\boldsymbol{s}(f,n) \sim \mathcal{N}_c\left(0, \mathcal{P}(f,n \mid \theta)\right).$$

On peut appliquer directement l'algorithme 13.1 pour le codage de $\tilde{\boldsymbol{s}}$, dans la mesure où $\boldsymbol{s}$ peut être compris comme une séquence de NF variables aléatoires gaussiennes indépendantes [7]. Cette application a été considérée dans [158, 159].

6. Voir les sections 3.3 page 52 et 4.1 page 57.

7. Si la transformée fréquentielle utilisée est celle de Fourier, on code indépendamment sa partie réelle et sa partie imaginaire, indépendantes. On a donc plutôt une séquence de $2FN$ variables à encoder, en imposant pour chacune une distorsion maximale de $\frac{D}{2}$.

Chapitre 14

Codage informé par les mélanges

Dans ce chapitre, j'applique la théorie du codage de source présentée au chapitre précédent dans une situation originale où le codeur et le décodeur disposent de la même donnée tierce, exploitée pour le codage. Je présente les principes généraux de cette situation de codage de sources *a posteriori* et de son cas particulier de séparation informée par codage.

Dans un premier temps, je montre en section 14.1 sur la base de la théorie du codage de source que la prise en compte de données *tierces* connues à la fois du codeur et du décodeur peut produire un gain théorique en débit pour la transmission d'une information. Ce gain en débit est directement lié à l'information mutuelle entre sources et données tierces. J'ai appelé cette configuration *codage de sources a posteriori*[1].

Dans ce chapitre, j'introduis CISS comme un formalisme puissant pour la séparation de sources informée, qui généralise les approches paramétriques présentées en partie précédente. CISS est le fruit de ma collaboration avec Alexey Ozerov et a été présenté pour la première fois dans [160], puis dans [136, 161].

En pratique, pour qu'un codeur *a posteriori* soit efficace, il est nécessaire de disposer d'une distribution des sources *a posteriori* qui exploite l'information mutuelle entre sources et données tierces. C'est ainsi que j'introduis en section 14.2 le cas particulier où les sources sont des processus gaussiens et où les données tierces sont des mélanges de réalisations de ces processus. Dans ce cas, l'ensemble de la discussion de la partie II peut être mis à profit pour fournir des distributions *a posteriori* des sources étant donnés leurs mélanges, utilisables pour un codage *a posteriori*. C'est ainsi que j'introduis la séparation informée par codage, (CISS, *Coding-Based Informed Source Separation*) qui fait l'objet de la suite de cette partie.

En section 14.3, je mets en perspective cette nouvelle approche pour la séparation informée avec celle proposée en partie III et je montre qu'elle en est une généralisation. Je la compare également aux techniques telles que SAOC qui mettent en œuvre une étape de codage du résiduel et je montre que CISS bénéficie de ce point de vue de propriétés d'optimalité intéressantes. Je conclus sur quelques perspectives relatives à CISS, en particulier sur la possibilité d'y inclure une fonction de distorsion perceptive.

14.1 Codage de sources *a posteriori*

14.1.1 Principes généraux

Dans le chapitre précédent, on a vu que la théorie du codage de source offre le cadre idéal par lequel aborder une situation classique de communication[2], représentée en figure 14.1. Un codeur observe la réalisation $\boldsymbol{s}$ d'un processus source, caractérisée par une distribution $p\left(\boldsymbol{s} \mid \bar{\Theta}\right)$ dont on lui fournit les paramètres quantifiés $\bar{\Theta}$. Son objectif est de transmettre à un décodeur le flux binaire

1. Dans la littérature, cette configuration porte également le nom de codage de Berger-Flynn-Gray [222, 207].
2. Dans toute cette étude, je considère que le flux binaire est transmis sans erreur du codeur vers le décodeur et donc que le canal de transmission est idéal. En conséquence, je ne m'intéresse pas au problème du *codage de canal*.

le plus petit possible permettant de reconstruire une approximation $\bar{\boldsymbol{s}}$ qui présente avec $\boldsymbol{s}$ une distorsion minimale.

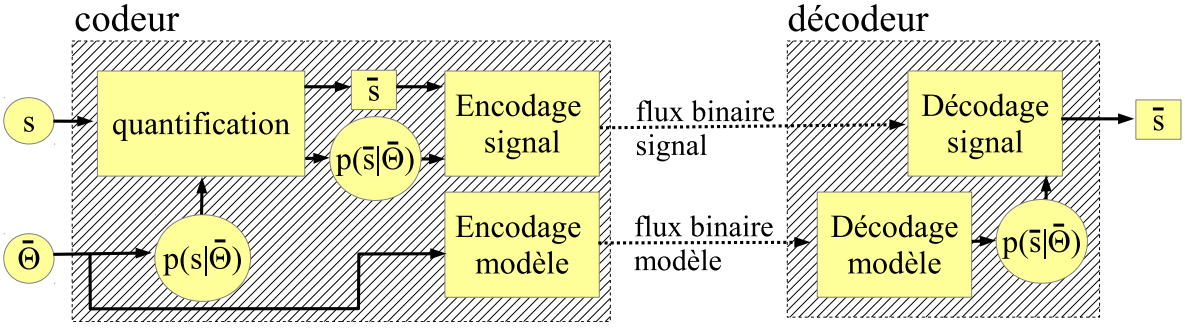

FIGURE 14.1: Situation classique de codage de source. Un codeur observe la réalisation $\boldsymbol{s}$ d'une source ainsi qu'un lot de paramètres quantifiés $\bar{\Theta}$ décrivant la distribution de la source. Il procède dans un premier temps à la quantification du signal. Ensuite, le signal et le modèle quantifiés sont encodés et transmis au décodeur. Après décodage du modèle, le décodeur procède au décodage du signal quantifié et produit un signal reconstruit $\bar{\boldsymbol{s}}$.

Comme on le voit sur la figure 14.1, le flux binaire transmis du codeur vers le décodeur se décompose comme la combinaison d'un flux binaire *modèle*, permettant de reconstruire $\bar{\Theta}$, et d'un flux binaire *signal*, permettant de reconstruire $\bar{\boldsymbol{s}}$ en conjonction avec la distribution $p\left(\bar{\boldsymbol{s}} \mid \bar{\Theta}\right)$. Le débit total R_{tot} se décompose ainsi comme la somme du débit modèle R_Θ et du débit signal [3] R_s :

$$R_{\text{tot}} = R_\Theta + R_s. \tag{14.1.1}$$

En utilisant les résultats théoriques et pratiques de la théorie débit-distorsion, j'ai montré au chapitre 13 qu'il est possible de garantir une distorsion moyenne entre le signal original $\boldsymbol{s}$ et sa reconstruction $\bar{\boldsymbol{s}}$ inférieure à une valeur D donnée, tout en utilisant à cette fin un débit signal minimal. On établit de plus les bornes de performance de ce genre de système sous la forme d'un débit en deçà duquel on ne peut pas garantir une distorsion moyenne inférieure à D. La question délicate de la répartition optimale d'un débit total entre signal et modèle sera évoqué rapidement en section 15.2. Je peux me contenter pour l'instant d'en dire qu'une fois estimé, le modèle est encodé indépendamment du signal.

Je propose à présent d'étudier une variante de ce schéma de communication, illustrée en figure 14.2, que j'appellerai codage de source *a posteriori*, aussi appelé codage de BERGER-FLYNN-GRAY dans la littérature [222, 207]. La subtilité introduite par le codage *a posteriori* par rapport au codage classique réside dans le fait que le codeur et le décodeur disposent de la connaissance à l'identique d'une *donnée tierce* $\boldsymbol{x}$, modélisée comme la réalisation d'une variable aléatoire x. On suppose ainsi qu'un même signal $\boldsymbol{x}$ est disponible à la fois à l'encodeur et au décodeur, sans qu'il soit nécessaire de considérer son transfert de l'un vers l'autre [4].

De la même manière que dans le cas classique, l'objectif du codeur est de générer un flux binaire de débit minimal qui permette au décodeur de produire une reconstruction $\bar{\boldsymbol{s}}$ de $\boldsymbol{s}$ dont la distorsion moyenne est inférieure à un seuil. Alternativement, on pourra chercher à produire une distorsion minimale à débit fixé.

Si on songe au diagramme à haut niveau de la séparation informée présenté en figure 1.1 page 8, on s'aperçoit qu'elle peut se comprendre comme un problème de codage *a posteriori* pour lequel l'information tierce $\boldsymbol{x}$ correspond aux mélanges des sources. Avant de considérer ce cas de figure plus longuement en section 14.2, je vais commencer par montrer les avantages du codage *a posteriori*.

3. Dans les calculs, les débits sont donnés en bits. Ce n'est que lors des évaluations que je ramène ces débits totaux à une unité du type kbps/source comme au chapitre 12.

4. On peut envisager le problème d'aussi encoder $\boldsymbol{x}$, mais je ne le considèrerai pas dans cet exposé.

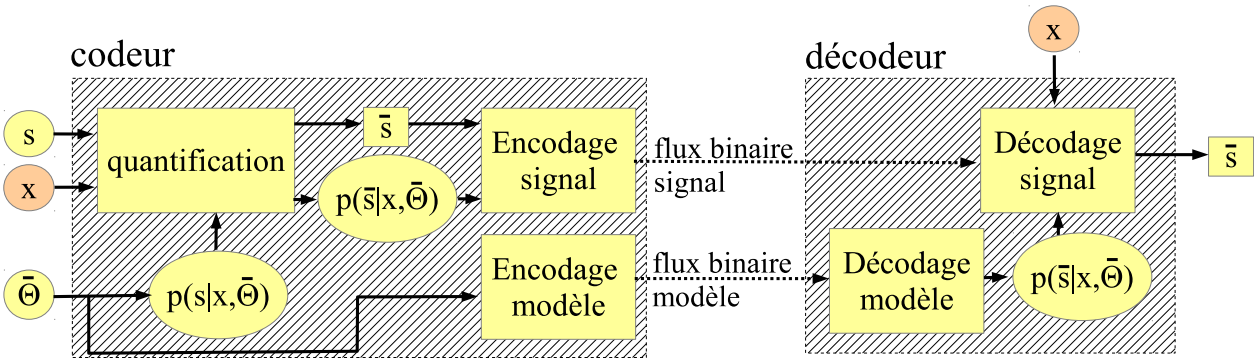

FIGURE 14.2: Situation de codage *a posteriori*, ou de codage de BERGER-FLYNN-GRAY. Par rapport à la situation classique de la figure 14.1, le codeur et le décodeur observent tous deux la même donnée tierce $\boldsymbol{x}$. Dans cette situation de codage *a posteriori*, toute l'analyse probabiliste mise en œuvre peut supposer donnée la connaissance de $\boldsymbol{x}$. Ainsi, la distribution des sources $p\left(\boldsymbol{s} \mid \bar{\Theta}\right)$ peut être remplacée par une distribution *a posteriori* $p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right)$ des sources étant donné $\boldsymbol{x}$.

14.1.2 Gain en débit d'un codage *a posteriori*

Dans le cas de la séparation informée, la donnée tierce $\boldsymbol{x}$ est constituée des mélanges des sources. En toute généralité, $\boldsymbol{x}$ est simplement la réalisation d'une variable aléatoire x.

Supposons que s soit une variable aléatoire vectorielle de dimension J. Le principal argument en faveur du codage *a posteriori* par rapport à un codage de source classique réside dans la réduction de débit que la connaissance de l'information tierce $\boldsymbol{x}$ peut permettre pour la transmission de $\bar{\boldsymbol{s}}$. Il semble en effet intuitif que si on dispose d'une information corrélée avec les signaux sources, alors le débit nécessaire pour leur transmission sera plus faible. Cette intuition se vérifie de manière triviale sous hypothèse de haute résolution à entropie contrainte, puisque l'entropie $H\left(\bar{s} \mid x\right)$ du message à transmettre est alors donnée non plus par 13.3.5, mais par :

$$H\left(\bar{s} \mid x\right) = h\left(s \mid x\right) - \frac{J}{2} \log 12D. \tag{14.1.2}$$

Dans la mesure où $I\left(s, x\right) = h\left(s\right) - h\left(s \mid x\right) \geq 0$, ce débit est nécessairement plus faible que $H\left(\bar{s}\right)$ et il est donc intéressant de prendre en compte la connaissance de $\boldsymbol{x}$ pour procéder à l'encodage de $\boldsymbol{s}$. Au pire, s et x sont indépendants, auquel cas $h\left(s\right) = h\left(s \mid x\right)$ et on se ramène au codage de source classique. Ce résultat peut s'illustrer dans le cas gaussien qui va m'intéresser particulièrement.

On a vu en section 13.2.1 que l'entropie d'une variable aléatoire gaussienne multivariée s de matrice de covariance K, de dimension $J \times J$, est donnée en nats par :

$$h\left(s\right) = \left(\frac{1}{2} \log \left(2\pi e\right)^{J} \left|K\right|\right).$$

Pour peu que s et x soient conjointement gaussiens, $s \mid x$ est gaussien également, de matrice de covariance K_{post}. On peut ainsi calculer le gain en débit provoqué par la prise en compte de la connaissance de x comme $H\left(\bar{s}\right) - H\left(\bar{s} \mid x\right)$, qui est nécessairement positif puisqu'il s'agit de $I\left(\bar{s}, x\right)$. Sous hypothèse de haute-résolution et à entropie contrainte, ce gain en débit s'obtient en utilisant 14.1.2 et 13.3.5 :

$$I\left(\bar{s}, x\right) = H\left(\bar{s}\right) - H\left(\bar{s} \mid x\right) = h\left(s\right) - h\left(s \mid x\right) = \frac{1}{2} \log \left(\frac{\left|K\right|}{\left|K_{\text{post}}\right|}\right) \geq 0.$$

Cette discussion permet en passant de prouver que le déterminant de $\left|K_{\text{post}}\right|$ est nécessairement plus petit que celui de $\left|K\right|$. Ce constat s'applique à l'ensemble des distributions *a posteriori* vues

en partie II. De manière intéressante, on a vu que ce résultat est immédiat à démontrer en utilisant des arguments issus de la théorie de l'information, mais moins trivial lorsqu'on l'envisage sous un angle algébrique.

Dans le cas $J = 1$, on peut reprendre l'exemple page 173 dans un contexte de codage *a posteriori*. Dans cet exemple, x est donné comme la somme de s avec un bruit additif ϵ indépendant de s. On a vu que dans ce cas, on a $|K| = \sigma_s^2$ et $|K_{post}| = \frac{\sigma_s^2 \sigma_\epsilon^2}{\sigma_s^2 + \sigma_\epsilon^2}$. Par conséquent, sous hypothèse de haute-résolution à entropie contrainte, il vient :

On quantifie facilement le gain en débit provoqué par la prise en compte de $\boldsymbol{x}$ dans le cas gaussien sous hypothèse haute-résolution. Ce gain est important si la source s à transmettre présente avec x une corrélation non négligeable.

$$H(\bar{s}) - H(\bar{s} \mid x) = \frac{1}{2} \log \left(1 + \frac{\sigma_s^2}{\sigma_\epsilon^2} \right).$$

Comme on le voit, si la variance σ_ϵ^2 du bruit additif est grande devant celle de la source à transmettre, le gain en débit provoqué par la prise en compte de l'observation $\boldsymbol{x}$ devient négligeable. En revanche, il devient très important lorsque la variance de ce bruit est faible devant σ_s^2.

Sans faire l'hypothèse de haute résolution, on peut s'interroger sur la relation entre la fonction débit-distorsion *conditionnelle* $R(D \mid x)$ dans un cas de codage *a posteriori* par rapport à celle $R(D)$ d'un codage de source classique, vue en section 13.2.2. Ces développements ont fait l'objet de certains travaux théoriques [222, 207], et il est mentionné dans ces publications que $R(D \mid x) \leq R(D)$.
Je n'ai trouvé nulle part de démonstration rigoureuse de cette affirmation, mais je conjecture qu'il sera en fait intéressant de prendre en compte la connaissance de $\boldsymbol{x}$ pour le codage de s dès lors qu'on peut exprimer $p(x \mid s, \bar{s})$ comme le produit de deux fonctions $f_1(x, s)$ et $f_2(x, \bar{s})$:

$$p(x \mid s, \bar{s}) = f_1(x, s) f_2(x, \bar{s}).$$

En effet, cette condition est nécessaire et suffisante [179] pour avoir :

$$I(s, \bar{s} \mid x) \leq I(s, \bar{s}) \tag{14.1.3}$$

quelle que soit la distribution jointe $p(s, \bar{s})$. Or, c'est par $I(s, \bar{s} \mid x)$ au lieu de $I(s, \bar{s})$ qu'est définie la fonction débit-distorsion conditionnelle $R(D \mid x)$ [207]. Dans les applications généralement considérées comme celle qui nous préoccupe ici, la variable x est indépendante de $\bar{s}$ étant donné s :

$$p(x \mid s, \bar{s}) = p(x \mid s),$$

et 14.1.3 s'applique donc.

14.2 Codage informé par les mélanges

14.2.1 Principes

Si on l'aborde sous l'angle du codage de source, le problème de la séparation informée que j'ai présenté en section 1.3 page 7 se comprend comme un cas particulier de codage *a posteriori*. Nous avons appelé cet angle d'approche la séparation informée par codage (CISS : Coding Based Informed Source Separation [160, 136, 161]). En effet, on a vu qu'un scénario de séparation informée se déroule en deux temps.

Durant l'étape de *codage*, on dispose des signaux sources $\boldsymbol{s}$ à transmettre ainsi que de leurs mélanges $\boldsymbol{x}$. On produit alors une information annexe Θ qu'on envoie à un *décodeur*. Le décodeur, muni de Θ et de $\boldsymbol{x}$ seulement, produit une estimée $\bar{\boldsymbol{s}}$ des sources. Si on adopte un formalisme probabiliste pour exprimer les distributions $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$ des sources $\boldsymbol{s}$ étant donnés les mélanges $\boldsymbol{x}$, ce problème devient équivalent à celui du codage *a posteriori* illustré en figure 14.2.

La principale différence avec l'approche paramétrique vue en partie III est qu'à présent, la distribution *a posteriori* $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$ n'est plus calculée dans le but de produire une seule estimée $\hat{\boldsymbol{s}}$

selon un critère comme le maximum *a posteriori* ou l'erreur quadratique minimale [5]. En effet, on a vu en partie III qu'une telle approche présente des performances bornées. Cela se comprend puisqu'elle ne peut pas garantir une distorsion inférieure à la borne de Cramér-Rao de l'estimateur qu'elle utilise [6].

La séparation informée est un cas de codage *a posteriori* où la donnée tierce est constituée des mélanges des sources.

Au lieu de cela, cette distribution $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$ est utilisée pour procéder à un codage optimal des sources $\boldsymbol{s}$. L'intérêt de la manœuvre réside dans le fait qu'on peut maintenant garantir que les sources reconstruites présentent avec les originales une distorsion moyenne inférieure à un seuil donné, tout en ayant minimisé le débit requis pour la transmission de l'information annexe correspondante [7].

14.2.2 Distributions *a posteriori* des sources

La partie I de cet exposé a porté sur la présentation d'un modèle de source puissant, selon lequel les signaux observés $\tilde{\boldsymbol{s}}$ sont des réalisations de processus gaussiens. J'ai montré que ce modèle est suffisamment flexible pour rendre compte d'une large diversité de formes d'ondes tout en permettant de caractériser un processus par un nombre restreint de paramètres θ par le biais d'une distribution $p(\tilde{\boldsymbol{s}} \mid \theta)$.

L'approche par codage nécessite d'observer au codeur les signaux à récupérer au décodeur. Je me restreins ici au cas simple où ce sont des sources ponctuelles observées qui doivent être récupérées.

En partie II, j'ai montré selon plusieurs modèles de mixage que des mélanges de processus gaussiens demeurent des processus gaussiens et qu'il est possible d'obtenir facilement les distributions $p(\tilde{\boldsymbol{s}} \mid \tilde{\boldsymbol{x}}, \Theta)$ des sources étant donnés leurs mélanges [8]. Tout l'effort déployé dans cette partie II sur le thème de la séparation de processus gaussiens n'avait somme toute comme objectif que l'établissement de ces distributions *a posteriori* dans de nombreuses configurations possibles du formalisme proposé. Le fait qu'on puisse en déduire la valeur des sources la plus probable *a posteriori* n'en est toujours apparue que comme un corolaire. L'intérêt d'avoir présenté l'objectif de la séparation de sources comme le calcul de telles distributions *a posteriori* est qu'il m'est à présent possible de les mettre toutes en œuvre dans un contexte de codage.

Selon la formalisation du chapitre 9 des problèmes de séparation informée, je me restreindrai pour la suite au cas où toutes les sources sont observées au codeur sous forme ponctuelle ($S_d = 0$), où tous les mixages sont ponctuels ($M_d = 0$) et où toutes les sources sont à récupérer sous forme ponctuelle ($Z_y = 0$). Dans ces conditions, le modèle utilisé est celui de la section 6.1 page 83.

14.2.3 Codage *a posteriori* de PGLS

Dans le système que je propose, les sources sont modélisées comme des PGLS. Tous les vecteurs $\{\boldsymbol{s}(f,n,\cdot)\}_{f,n}$ de leurs TFCT sont donc supposés indépendants. On peut par conséquent les considérer comme une *séquence* de NF vecteurs indépendants de dimension $J \times 1$ dont chacun est caractérisé par une distribution gaussienne. Cette distribution est soit la distribution *a priori* :

$$\boldsymbol{s}(f,n,\cdot) \mid \theta \sim \mathcal{N}_c\left(0, \operatorname{diag}\mathcal{P}(f,n,\cdot \mid \theta)\right) = \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{prior}}(f,n,\cdot), K_{\text{prior}}(f,n,\cdot,\cdot)\right)$$

ou *a posteriori* :

$$\boldsymbol{s}(f,n,\cdot) \mid \boldsymbol{x}(f,n,\cdot), \Theta \sim \mathcal{N}_c\left(\boldsymbol{\mu}_{\text{post}}(f,n,\cdot), K_{\text{post}}(f,n,\cdot,\cdot)\right)$$

5. qui se confondent dans le cas gaussien.

6. Cette borne de Cramer-Rao est donnée analytiquement dans le cas gaussien par la matrice de covariance de la distribution *a posteriori*.

7. Alternativement, on peut garantir la distorsion la plus petite possible à débit fixé pour un modèle de source et de mixage donnés.

8. 5.1.3, 5.2.6, 5.3.3, 6.1.8, 6.1.19, 6.1.19, 6.2.8, 9.2.5 sont toutes des exemples rencontrés au cours de ce texte de telles distributions *a posteriori* des sources étant donnés les mélanges.

avec $\boldsymbol{\mu}_{\text{post}}(f,n,\cdot)$ et $K_{\text{post}}(f,n,\cdot,\cdot)$ définis par 6.1.20 et 6.1.21 page 89 respectivement [9].

Le cadre est alors propice à l'utilisation des résultats de la section 13.3.2 pour l'encodage de la séquence des $\boldsymbol{s}(f,n,\cdot)$ en utilisant l'une ou l'autre de ces distributions pour chacun des éléments de la séquence. Quoiqu'il en soit, le codage *a posteriori* se fait alors, comme le codage *a priori*, selon les opérations décrites par l'algorithme 13.1 page 181. Sous hypothèse de haute-résolution il permet de garantir que la reconstruction $\bar{\boldsymbol{s}}$ des sources au décodeur présente une distorsion inférieure à un seuil donné D, tout en minimisant le débit nécessaire à sa transmission. L'étape d'encodage arithmétique mise en jeu a été décrite en section 13.1.3 page 169. Étant données les sources estimées $\bar{\boldsymbol{s}}(f,n,\cdot)$, les formes d'ondes recherchées sont calculées au décodeur par TFCT inverse.

On peut déduire de ces considérations les différentes étapes requises par une architecture de codage *a posteriori* :

1. Étant donnés les signaux sources et mélanges, estimer le modèle Θ et le quantifier pour obtenir $\bar{\Theta}$.
2. En utilisant le formalisme de séparation de PGLS vu en partie II, déduire pour chaque (f,n) la moyenne $\boldsymbol{\mu}_{\text{post}}(f,n,\cdot)$ et la covariance $K_{\text{post}}(f,n,\cdot,\cdot)$ de la distribution $p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right)$.
3. Choisir une distorsion moyenne D à ne pas dépasser.
4. Encoder la séquence des NF vecteurs $\boldsymbol{s}(f,n,\cdot)$ dans un flux signal en utilisant l'algorithme 13.1. Le débit-signal correspondant est noté R_s.
5. Adjoindre le flux modèle, de débit R_Θ, qui permet la reconstruction du modèle $\bar{\Theta}$.
6. Au décodeur, reconstruire $\bar{\Theta}$, puis les sources en utilisant 13.3.9 avec les composantes $\overline{\boldsymbol{\xi}}_l$ quantifiées.

14.3 Un changement de perspective

14.3.1 Le cas paramétrique

Dans cette section, je propose de montrer que l'approche paramétrique détaillée en partie III peut s'expliquer comme un cas particulier de codage *a posteriori*.

Dans l'approche paramétrique, la procédure de codage se limite à estimer puis encoder les paramètres $\bar{\Theta}$ de l'information annexe par lesquels les distributions *a posteriori* $p\left(\boldsymbol{s}(f,n,\cdot) \mid \boldsymbol{x}(f,n,\cdot), \bar{\Theta}\right)$ peuvent être reconstruites au décodeur. La procédure de décodage se résume alors à estimer les sources par leur moyenne *a posteriori*. En d'autres termes, le codeur paramétrique fait intervenir un débit signal nul ($R_s = 0$) et tout le débit requis pour le transfert de l'information annexe est donc compris par le débit modèle

$$R_{\text{tot}} = R_\Theta.$$

On peut voir ce régime de fonctionnement comme optimal sous un angle de codage de sources dès lors qu'on est prêt à tolérer une distorsion supérieure à la plus grande de toutes les valeurs propres des matrices $K(f,n,\cdot,\cdot)$ définies en 6.1.21 page 89.

Pour le montrer, commençons par définir :

$$K(f,n,\cdot,\cdot) = U_{fn}\text{diag}\left[\lambda_{fn,1},\ldots,\lambda_{fn,J}\right]U_{fn}^H$$

comme la décomposition en valeurs propres de $K(f,n,\cdot,\cdot)$. Soit

$$D_{\text{param}} \geq D_{\text{lim}} = \max_{f,n,j} \lambda_{fn,j}$$

un réel positif plus grand que la plus grande de toutes les valeurs propres $\lambda_{fn,j}$.

Chaque élément $\boldsymbol{s}(f,n,\cdot)$ de la séquence à encoder étant distribué selon une loi gaussienne de moyenne $\boldsymbol{\mu}(f,n,\cdot)$ et de covariance $K(f,n,\cdot,\cdot)$, je peux appliquer à chacun les résultats de

9. On peut ici mettre en œuvre n'importe laquelle des autres distributions *a posteriori* rencontrées en partie II pour la séparation de PGLS.

la section 13.2.4 page 176 pour déduire le débit moyen minimal requis pour avoir une distorsion quadratique moyenne maximale de D_{param}. On aura en appliquant 13.2.13 :

$$R_s\left(D_{\text{param}}\right) = \sum_{f,n,j} \max\left(0, \frac{1}{2}\log\frac{\lambda_{fn,j}}{D_{\text{param}}}\right) = 0.$$

La théorie débit-distorsion nous indique que le débit minimal théorique pour reconstruire les sources avec une distorsion moyenne de D_{param} est nul, pour chaque (f, n). Cela se comprend aisément en considérant que les $\lambda_{fn,j}$ correspondent aux variances des signaux à encoder. Il suffit donc de choisir leur moyenne comme estimées pour avoir une distorsion de $\lambda_{fn,j}$, inférieure à la distorsion demandée. Par contre, pour toute valeur plus petite que D_{lim}, le débit requis sera strictement positif. On peut déduire de cette discussion le résultat suivant :

Le cas paramétrique peut être conçu comme un cas particulier de CISS pour lequel on tolère une distorsion maximale plus grande que la plus grande des valeurs propres de tous les $K\left(f, n, \cdot, \cdot\right)$. Si on cherche une distorsion plus petite avec le même modèle probabiliste de sources et les mêmes paramètres $\bar{\Theta}$, il faut avoir recours à des méthodes de codage et ne plus estimer les sources par leur simple moyenne *a posteriori*.

Il faut cependant tempérer ce résultat par le constat que CISS nécessite la disponibilité à l'encodeur des signaux à récupérer au décodeur. Or, tous les cas de figures supportés par le codeur paramétrique et présentés au chapitre 9 ne rentrent pas dans ce cadre. En particulier, on a vu au chapitre 11 que le codeur paramétrique permet d'estimer des signaux sources non observés comme c'est le cas pour une source diffuse mixée de manière convolutive [10]. Si l'application met en jeu ce cas de figure, on ne pourra pas avoir recours à CISS pour de telles sources. Il faut cependant garder à l'esprit que dans de nombreux cas, récupérer les signaux observés au codeur sera l'objectif à atteindre.

14.3.2 Codage informé ou codage du résiduel ?

Comme je l'ai déjà évoqué, certaines techniques de séparation informée de l'état de l'art mettent en œuvre une étape de codage. C'est par exemple le cas de la technique hybride proposée dans [170] ou bien de SAOC [61, 66]. Dans tous les cas, on peut comprendre la stratégie correspondante de la manière suivante.

Le codage de source n'a été utilisé en séparation informée que pour transmettre l'erreur d'estimation, indépendamment de l'étape de séparation. Une telle approche est sous-optimale.

Pour commencer, un système de séparation informée paramétrique est utilisé pour récupérer des estimées $\hat{\boldsymbol{s}}$ des sources. Pour SAOC, ce système ressemble à celui que j'ai présenté en partie III, avec toutes les réserves qu'on peut faire sur ce point et mentionnées en section 12.6.4 page 159. Pour [170], la méthode paramétrique utilisée est l'inversion locale, présentée en section 1.3.3 page 9. Ce n'est que dans un deuxième temps qu'un algorithme classique de codage de forme d'onde est utilisé, pour transmettre l'erreur d'estimation $\boldsymbol{s} - \hat{\boldsymbol{s}}$. Pour ce faire, on considère l'utilisation d'un codeur de forme d'onde bas-débit AAC [147], appliqué indépendamment à l'erreur d'estimation faite sur chaque source. Toutes ces techniques procèdent donc à un *codage du résiduel*.

Contrairement à une technique de codage du résiduel où ce sont les erreurs d'estimation qui font l'objet d'un codage, CISS procède directement à l'encodage des signaux sources eux-mêmes. C'est la distribution utilisée pour cet encodage qui dépend des mélanges. Cette remarque a trois conséquences importantes qui marquent la supériorité au moins théorique de CISS sur une approche de codage du résiduel :

10. Par contre, il est possible d'utiliser CISS pour le codage de sources ponctuelles mixées de manière diffuse, pour peu qu'on souhaite les estimer telles qu'observées au codeur. Je ne me préoccuperai pas de ce cas de figure ici.

- CISS ne nécessite pas le transfert vers le décodeur de paramètres supplémentaires décrivant la distribution des signaux résiduels. Ceux nécessaires à la séparation sont suffisants. Dans une approche informée utilisant le codage de résiduel, les paramètres de séparation de sources ne sont plus utilisés lors de l'étape de codage et ce sont d'autres paramètres qui sont estimés, utilisés et transmis par un codeur tel que AAC.
  On peut voir CISS comme une technique de codage de résiduel pour laquelle ce sont les mêmes paramètres qui régissent la séparation paramétrique et la distribution des signaux résiduels.
- Contrairement à des techniques de codage du résiduel qui encodent indépendamment chacune des erreurs d'estimation, CISS exploite les corrélations *a posteriori* des sources pour leur encodage, ce qui correspond à un gain important en pratique. Pour le comprendre, on peut considérer le cas simple d'un seul mélange produit par la somme de deux sources. Étant donné le mélange, il n'est nécessaire que de coder l'une d'entre elles, puisque l'autre peut être récupérée par soustraction de la source codée du mélange. CISS considère le cas général de toute corrélation *a posteriori* des sources.
- Bien que je n'aie pas encore évoqué ce point (je le ferai en section suivante), on peut d'ores et déjà noter que CISS permet d'effectuer un codage perceptif des sources en utilisant leur distribution *a posteriori*. Cela se fait en pratique en pondérant différemment la distorsion utilisée en fonction du point (f, n) [174, 205]. Dans une telle approche, c'est bien la perception des sources elles-mêmes qu'on prend en compte pour l'encodage. Au contraire, une technique de codage du résiduel applique le modèle perceptif non pas aux sources, mais à l'erreur de reconstruction. Or, rien ne garantit qu'avoir une différence imperceptible sur les résiduels conduise à une différence imperceptible entre sources originales et estimées, ni l'optimalité de cette approche quand bien même ce serait le cas.

Ces trois points montrent la supériorité conceptuelle de CISS par rapport à une technique de codage du résiduel.

Cependant, il reste entendu que des considérations pratiques peuvent justifier l'utilisation conjointe d'une séparation paramétrique suivie d'un encodage du résiduel. En particulier, il est possible de disposer d'une technique de séparation informée efficace mais qui ne conduit pas à l'obtention immédiate d'une distribution *a posteriori* $p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right)$, indispensable pour CISS. C'est le cas de l'inversion locale envisagée dans [170]. Dans cette situation, il est clair qu'un codage du résiduel reste une option tout à fait valable [11].

Dans la mesure où il permet de garantir un débit minimal pour une distorsion fixée quelconque, CISS apparaît comme un candidat idéal dans de nombreux scénarios de séparation informée, en particulier ceux nécessitant une qualité de restitution garantie.

Ceci étant dit, il faut noter que l'utilisation de la distribution *a posteriori* pour l'encodage des sources plutôt qu'une distribution *a priori* ne conduit pas à une complexité notablement plus grande de CISS par rapport à des algorithmes de codage de source tels que ceux considérés dans [157, 121, 159], qui peuvent opérer en temps réel [12].

14.3.3 Perspectives : codage perceptif

Le principal défaut de CISS tel que je l'ai présenté jusqu'à présent par rapport à un codeur audio classique réside dans le fait que la distorsion considérée entre le signal original $\boldsymbol{s}$ et sa reconstruction $\bar{\boldsymbol{s}}$ est l'erreur quadratique. Ainsi, ce n'est qu'en fonction de ce critère que j'ai jusqu'à présent développé le formalisme de codage de sources *a posteriori* qu'est CISS. Or, il est bien connu qu'un tel critère ne rend compte que passablement de la différence perçue par l'ouïe humaine entre un son et sa reconstruction [205]. C'est pour cette raison que de multiples méthodes de codage

11. On peut également remarquer qu'il est possible d'appliquer CISS en utilisant une autre technique de séparation que les méthodes suggérées ici, pour peu qu'elle conduise à l'obtention d'une distribution *a posteriori* des sources. De ce point de vue, il est tout à fait possible d'entrevoir des applications de CISS pour lesquelles le modèle de source *a posteriori* est différent de tous ceux présentés en partie II.
12. au décodage tout au moins.

de source ont été mises au point qui permettent de minimiser le débit pour d'autres fonctions de distorsion [174, 157, 129] et qui ont été utilisées pour le codage de signaux audio [157, 76].

CISS permet d'intégrer de manière naturelle les avancées récentes —et moins récentes— en codage de source qui utilisent des fonctions de distorsion perceptives.

De mon point de vue, c'est un des principaux avantages de CISS d'établir sur des bases solides un lien entre le problème de la séparation informée et celui du codage de source. C'est ainsi que toute avancée significative dans le domaine du codage de source audio pourra être utilisée de manière directe dans le cadre de CISS. L'algorithme 13.1 page 181 d'encodage utilisé en pratique en est un exemple. Il en va de même pour le codage perceptif.

Je ne présenterai pas plus avant dans cet exposé l'utilisation de fonctions de distorsion perceptives dans CISS, qui fait l'objet de travaux en cours et que nous avons déjà évoqué dans [161]. Je note cependant que ces extensions ne présentent pas de difficulté particulière compte tenu des récentes avancées effectuées dans le domaine et je souligne encore une fois le fait qu'un codage perceptif effectué dans le cadre de CISS garantira une bonne reconstruction des sources elles-mêmes, contrairement à un codage perceptif du résiduel, qui ne pourra que garantir une bonne reconstruction perceptive de l'erreur d'estimation, ce qui est différent.

Chapitre 15

Codeur et décodeur informés

Après en avoir présenté le principe sur des bases plutôt théoriques au chapitre précédent, je précise dans ce chapitre les opérations effectuées en pratique par le codeur et le décodeur CISS, en vue de leur implémentation. Dans la mesure où l'estimation du modèle est très similaire à celle effectuée dans le cas paramétrique et que la technique de codage met en œuvre l'algorithme 13.1 déjà évoqué, je me contenterai ici d'organiser ces éléments pour les rassembler dans un algorithme concret d'encodage et de décodage.

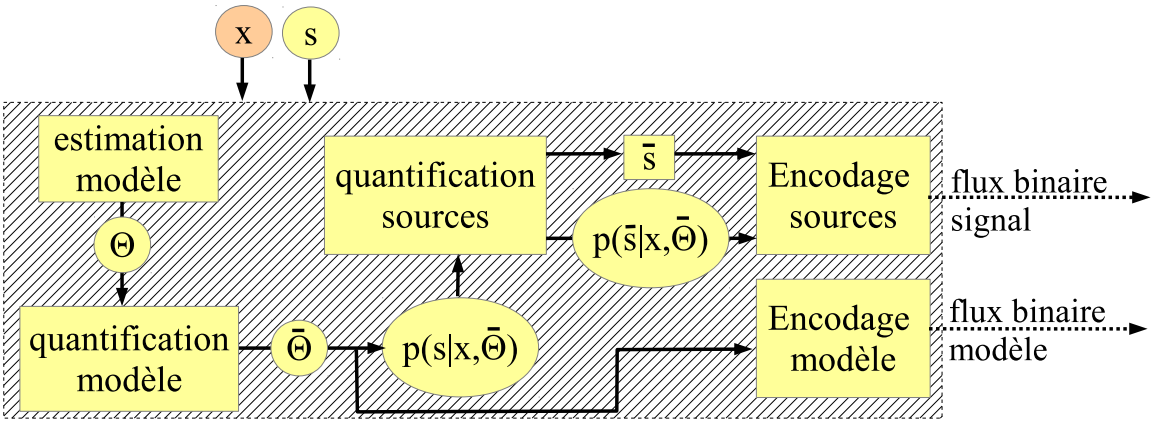

FIGURE 15.1: Codeur CISS. Les différents blocs constituant ce codeur incluent l'estimation et la quantification du modèle, ainsi que la quantification et l'encodage des sources en utilisant leur distribution *a posteriori*.

En figure 15.1, j'ai représenté la structure générale du codeur CISS. Ce codeur dispose de la connaissance des signaux sources $\boldsymbol{s}$ à encoder ainsi que des signaux mélanges $\boldsymbol{x}$.

Ces deux tenseurs, de dimensions respectives $F \times N \times J$ et $F \times N \times I$ sont les TFCT des formes d'onde $\tilde{\boldsymbol{s}}$ et $\tilde{\boldsymbol{x}}$ correspondantes, de dimension $L \times J$ et $L \times I$. Alternativement, on peut considérer l'utilisation d'une transformée fréquentielle différente de la TFCT telle que la transformée en cosinus discrets (MDCT) ou bien en toute généralité la sortie d'un banc de filtres composé de F bandes. Ces alternatives peuvent permettre de produire moins de points (f, n) à encoder que la TFCT et une représentation réelle au lieu de valeurs complexes, conduisant à des débits plus compétitifs. Cependant, il faut prendre garde au fait que l'approximation 6.1.15 page 88 :

$$\forall\,(f,n)\,,\ \boldsymbol{x}\,(f,n,\cdot) \approx A\,(f)\,\boldsymbol{s}\,(f,n,\cdot)$$

sur laquelle repose la séparation de mélanges convolutifs de PGLS en section 6.1.3 ne tient pas nécessairement pour ces transformées, sauf dans le cas d'un mélange linéaire instantané. Au besoin,

le modèle de séparation peut être revu [1], mais je supposerai ici simplement que les résultats de la section 6.1.3 sont valables, ce qui suppose soit un mélange instantané, soit un mélange convolutif avec l'utilisation d'une TFCT pour les signaux.

Muni des TFCT des signaux, la première tâche que doit effectuer le codeur CISS est l'estimation du modèle de source *a posteriori*, c'est-à-dire du modèle Θ qui définit la distribution $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$. J'aborde cette question en section 15.1. Une fois le modèle Θ estimé, la question se pose du débit $R_{\bar{\Theta}}$ qu'on doit allouer à son transfert par rapport au débit signal R_s et donc de la manière de quantifier Θ pour produire $\bar{\Theta}$. Ce point sera traité en section 15.2. Une fois le modèle quantifié disponible, je montrerai en section 15.3 comment les résultats de la section 13.3.2 page 179 peuvent être utilisés pour procéder à l'encodage des sources. Enfin, l'opération de décodage sera précisée en section 15.4.

15.1 Estimation du modèle Θ

Comme on peut le constater à l'examen de la figure 15.1, ce qu'on appelle *modèle* dans le cadre de CISS correspond à ce qui était appelé *information annexe* pour une technique de séparation informée paramétrique en partie III. En effet, dans le cas paramétrique, le débit signal R_s est nul, ce qui a pour conséquence le fait que l'information transmise du codeur vers le décodeur se confond avec les paramètres Θ du modèle. C'est par le biais de ce modèle Θ qu'est disponible la distribution $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$, commune aux deux approches. Dans le cas de CISS, le débit signal R_s se rajoute au débit modèle R_Θ pour former le flux annexe total, de débit R_{tot}. Le modèle considéré par CISS est ici constitué du modèle θ des sources, ainsi que des paramètres $\{A(f)\}_f$ de mélange :

$$\Theta = \left\{\theta, \{A(f)\}_f\right\}.$$

Sous hypothèse haute-résolution à entropie contrainte, le débit total $R_s(D \mid \Theta)$ nécessaire pour le codage *a posteriori* des sources est la somme des débits requis pour le transfert de la séquence des JFN vecteurs $\boldsymbol{s}(f, n, \cdot)$ en utilisant leur distribution $p(\boldsymbol{s}(f, n, \cdot) \mid \boldsymbol{x}(f, n, \cdot), \Theta)$. Ce débit est donné par la fonction de débit-distorsion opérationnelle [2] [122, 161] donnée par 13.3.8 page 181 :

$$R_s(D \mid \Theta) = -\log_2 p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta) - \frac{JFN}{2} \log_2 12D. \tag{15.1.1}$$

L'estimation du modèle Θ par un codeur CISS est identique à celle d'un codeur paramétrique présentée en chapitres 10 et 11. À distorsion fixée, cette estimation se justifie cependant pour CISS selon un critère de minimisation du débit.

Par conséquent, de manière à minimiser le débit nécessaire à l'encodage des sources en utilisant leur distribution *a posteriori*, l'estimation du modèle Θ doit se faire de manière à maximiser $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$, indépendamment de la distorsion. Comme on le voit, la problématique relative à l'estimation du modèle est exactement la même dans le cas de CISS que dans le cas paramétrique. J'ai en effet déjà consacré les chapitres 10 et 11 à la question de l'estimation au maximum de vraisemblance des paramètres du modèle Θ, appelés alors *information annexe*. Dans le cas de CISS, on peut donc appliquer exactement les mêmes procédures. J'ai remarqué expérimentalement que dans le cas où le mélange est convolutif, il est important d'utiliser de préférence l'algorithme 11.3 pour l'estimation de Θ. Dans le cas d'un mélange instantané, utiliser l'algorithme présenté au chapitre 10 est suffisant.

Quoi qu'il en soit, je note que les implémentations que j'ai considérées ne procèdent pas réellement à la maximisation de $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$, mais plutôt à celle de $p(\boldsymbol{s}, \boldsymbol{x} \mid \Theta)$. J'ai justifié ce choix en section 10.1 lorsque j'ai montré qu'il correspond à un point de vue *génératif* d'apprentissage, plus simple, et non pas à un apprentissage *discriminant*, optimal mais difficile dans notre contexte.

1. Pour une MDCT, on peut envisager d'utiliser un modèle HR-NMF tel que présenté dans [11] qui permet de rendre compte des dépendances induites par un filtrage entre les valeurs adjacentes d'une MDCT.

2. Il faudrait remplacer $2JFN$ par JFN en cas de transformée fréquentielle réelle telle que la MDCT. Il faut faire attention ici au fait que pour que F et N soient identiques entre une MDCT et une TFCT, il faut que le recouvrement de la TFCT soit de 0.5.

Comme je l'ai déjà évoqué alors, c'est une piste intéressante de recherche que de paramétrer différemment qu'en 6.1.19 page 89 la distribution $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$, de manière à permettre l'apprentissage optimal de Θ. Une telle approche discriminante pourrait très naturellement s'insérer dans le cadre de CISS.

15.2 Compromis entre débit signal et débit modèle [3]

Une fois les paramètres du modèle estimés, il est important de déterminer la manière de les encoder qui permet à la fois de les transmettre à débit $R_{\bar{\Theta}}$ réduit, mais qui autorise malgré tout un encodage des sources efficace en utilisant la distribution $p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right)$ plutôt que $p(\boldsymbol{s} \mid \boldsymbol{x}, \Theta)$. De plus, il est important de déterminer une stratégie efficace permettant de choisir les paramètres libres du modèle, tels que le nombre de composantes pour le modèle NTF ou la qualité de compression pour le modèle CI.

Sous hypothèse de haute résolution, le débit optimal $R^{\star}_{\Theta}$ à allouer au modèle est indépendant de la distorsion.[122]. Il reste à déterminer comment quantifier le modèle à débit donné et ensuite la valeur précise de ce débit optimal.

Des travaux récents dans le domaine du codage de source [122, 157, 158, 159] se sont précisément penchés sur cette question délicate de la manière optimale d'estimer et de quantifier les paramètres d'un modèle de source en vue du codage. Ce qui suit est un condensé de ce qui apparait aujourd'hui comme la stratégie optimale en la matière.

Un des résultats forts de la théorie haute-résolution dans ce contexte et qui a une importance déterminante pour la suite de cette section est qu'on montre que le débit optimal $R^{\star}_{\Theta}$ à allouer au modèle est indépendant de la distorsion [122]. Ce résultat étant donné, il reste à déterminer de quelle manière il faut quantifier le modèle pour R_{Θ} fixé et quelconque, et ensuite comment on peut déterminer le débit optimal $R^{\star}_{\Theta}$.

Soit donc R_{Θ} quelconque. Le débit total nécessaire au codage des sources se décompose comme la somme de R_{Θ} et du débit nécessaire au codage des sources en utilisant des paramètres quantifiés $\bar{\Theta}$:

$$R_{\text{tot}} = R_{\Theta} + R_s\left(D \mid \bar{\Theta}\right).$$

On montre [122] que le coût en débit $\Psi\left(\bar{\Theta}, \Theta, \boldsymbol{s}, \boldsymbol{x}\right)$ induit par l'utilisation des paramètres quantifiés $\bar{\Theta}$ plutôt que des paramètres Θ dans l'opération de codage *a posteriori* des source $\boldsymbol{s}$ est donné par :

$$\begin{aligned} \Psi\left(\bar{\Theta}, \Theta, \boldsymbol{s}, \boldsymbol{x}\right) &= R_s\left(D \mid \bar{\Theta}\right) - R_s\left(D \mid \Theta\right) \\ &= \log_2 \frac{p\left(\boldsymbol{s} \mid \boldsymbol{x}, \Theta\right)}{p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right)} \end{aligned} \tag{15.2.1}$$

Pour un débit modèle R_{Θ} fixé [4], l'objectif de la quantification devient donc de minimiser le critère 15.2.1, de manière à réduire R_{tot}. Malheureusement, compte tenu de l'expression complexe 6.1.19 page 89 de la distribution *a posteriori*, cette optimisation est difficile pour la mise au point de procédures simples de quantification du modèle.

La stratégie adoptée dans [161] a consisté une fois encore à remplacer les distributions *a posteriori* dans 15.2.1 par les distributions jointes $p(\boldsymbol{s}, \boldsymbol{x} \mid \Theta)$ et $p\left(\boldsymbol{s}, \boldsymbol{x} \mid \bar{\Theta}\right)$ correspondantes. Ce choix se justifie une fois encore en invoquant un point de vue génératif pour l'apprentissage plutôt qu'un

3. Cette discussion sur l'encodage du modèle est issue de l'article [161] d'Ozerov, Liutkus *et al.*
4. En particulier pour le débit modèle optimal $R^{\star}_{\Theta}$ encore inconnu à ce stade.

point de vue discriminant. On aboutit alors à un nouveau critère à minimiser :

$$\begin{aligned}\Psi\left(\bar{\Theta},\Theta,\boldsymbol{s},\boldsymbol{x}\right) &\approx \log_2\frac{p\left(\boldsymbol{s},\boldsymbol{x}\mid\Theta\right)}{p\left(\boldsymbol{s},\boldsymbol{x}\mid\bar{\Theta}\right)}\\ &= \log_2\frac{p\left(\boldsymbol{s}\mid\theta\right)p\left(\boldsymbol{x}\mid\boldsymbol{s},A\right)}{p\left(\boldsymbol{s}\mid\bar{\theta}\right)p\left(\boldsymbol{x}\mid\boldsymbol{s},\bar{A}\right)}\\ &= \log_2\frac{p\left(\boldsymbol{s}\mid\theta\right)}{p\left(\boldsymbol{s}\mid\bar{\theta}\right)}+\log_2\frac{p\left(\boldsymbol{x}\mid\boldsymbol{s},A\right)}{p\left(\boldsymbol{x}\mid\boldsymbol{s},\bar{A}\right)}\end{aligned}\tag{15.2.2}$$

où on reconnaît un premier terme qui ne dépend que du modèle de source θ et un deuxième qui ne dépend que des paramètres de mixage A. Lorsque j'ai évoqué l'encodage de l'information annexe en section 10.3, j'ai souligné le fait que la transmission des paramètres de mélange A ne provoque pas un débit très important à l'échelle d'un morceau, puisque leur nombre de dépend pas de la longueur des signaux considérés. Par conséquent, j'ai effectué une simple quantification à haute résolution de A sur 32 bits. Dans la suite, je peux donc considérer comme négligeable le deuxième terme de 15.2.2, avec $\frac{p(\boldsymbol{x}|\boldsymbol{s},A)}{p\left(\boldsymbol{x}|\boldsymbol{s},\bar{A}\right)}\approx 1$. Ψ devient alors indépendant des mélanges $\boldsymbol{x}$ et s'écrit :

$$\begin{aligned}\Psi\left(\bar{\Theta},\Theta,\boldsymbol{s}\right) &\approx \log_2\frac{p\left(\boldsymbol{s}\mid\theta\right)}{p\left(\boldsymbol{s}\mid\bar{\theta}\right)}\\ &= \frac{1}{\log 2}\sum_{j,f,n}\left(\frac{v\left(f,n,j\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}-\frac{v\left(f,n,j\right)}{\mathcal{P}\left(f,n,j\mid\theta\right)}-\log\frac{\mathcal{P}\left(f,n,j\mid\theta\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}\right)\\ &= \frac{1}{\log 2}\sum_{j,f,n}\left(\frac{\mathcal{P}\left(f,n,j\mid\theta\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}-\log\frac{\mathcal{P}\left(f,n,j\mid\theta\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}-1\right)+\\ &\quad\frac{1}{\log 2}\sum_{j,f,n}\left(\frac{v\left(j,f,n\right)-\mathcal{P}\left(f,n,j\mid\theta\right)}{\mathcal{P}\left(f,n,j\mid\theta\right)}\right.\\ &\quad\left.\times\frac{\mathcal{P}\left(f,n,j\mid\theta\right)-\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}\right)\end{aligned}\tag{15.2.3}$$

$$\approx \frac{1}{\log 2}\sum_{j,f,n}\left(\frac{\mathcal{P}\left(f,n,j\mid\theta\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}-\log\frac{\mathcal{P}\left(f,n,j\mid\theta\right)}{\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)}-1\right).\tag{15.2.4}$$

L'approximation menant à la ligne 15.2.4 provient de l'hypothèse raisonnable que les deux erreurs relatives de modélisation et de quantification formant le deuxième terme de la ligne 15.2.3 sont décorrélées et qu'au moins une des deux est de moyenne nulle [122, 161]. Dans ces conditions, on voit que Ψ ne dépend plus de $\boldsymbol{s}$ mais uniquement de θ et de $\bar{\theta}$. On peut en outre reconnaître dans l'équation 15.2.4 la divergence d'Itakura-Saito, et le problème de la quantification de θ se ramène donc au problème de la minimisation de

On montre après quelques hypothèses raisonnables que la quantification optimale du modèle pour CISS obéit aux mêmes contraintes que dans le cas paramétrique. On peut donc appliquer les procédures de la section 10.3.

$$\Psi\left(\bar{\theta},\theta\right)=\sum_{j,f,n}d_0\left(\mathcal{P}\left(f,n,j\mid\theta\right)\mid\mathcal{P}\left(f,n,j\mid\bar{\theta}\right)\right),$$

où on reconnaît le même critère que celui défini en 10.3.2 page 134 dans le cas paramétrique, à la différence qu'il a cette fois été justifié sur des bases plus solides. Dans tous les cas, les stratégies discutées en section 10.3.1 pour l'encodage des paramètres des modèles CI et NTF, respectivement, restent valables dans le cas de CISS.

Je viens de montrer qu'on peut établir la meilleure manière de quantifier les paramètres du modèle, à débit R_Θ fixé. Il reste à présent à déterminer quelle est la valeur $R_\Theta^\star$ optimale de ce débit,

dont la théorie haute-résolution démontre l'existence et l'indépendance vis-à-vis de la distorsion. En d'autres termes,on cherche maintenant une stratégie permettant de choisir les paramètres librès du modèle comme le nombre de composantes pour NTF ou la qualité de compression d'image pour CI.

Toujours sous hypothèse haute-résolution et muni à présent de la manière optimale de quantifier le modèle, le coût total $R_{\text{tot}}(D)$ requis pour l'encodage des sources devient, en utilisant 15.1.1 :

$$\begin{aligned} R_{\text{tot}}(D) &= R_\Theta + R_s\left(D \mid \bar{\Theta}\right) \\ &= R_\Theta - \log_2 p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right) - \frac{JFN}{2}\log_2 12D. \end{aligned} \tag{15.2.5}$$

Puisque la théorie débit distorsion prédit que le débit modèle optimal R_Θ est indépendant de la distorsion D, il est celui qui permettra de minimiser $R_{\text{tot}}(D)$ pour tout D fixé et donc celui qui minimise :

$$\eta\left(\boldsymbol{s}, \boldsymbol{x}, \bar{\Theta}\right) = R_\Theta - \log_2 p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right). \tag{15.2.6}$$

L'encodage optimal à haute résolution du modèle est celui qui mène à la minimisation de 15.2.6. En pratique, on peut aussi choisir la quantification $\bar{\Theta}$ qui conduit aux meilleures distorsions pour des débits ciblés dans les applications.

La stratégie adoptée pour l'optimisation du modèle consiste ainsi à calculer 15.2.6 pour plusieurs paramètres de quantification du modèle, et à choisir ceux qui minimisent cette valeur. C'est en effet eux qui conduiront aux meilleurs débits sous hypothèse haute-résolution.

On peut remarquer que pour un débit modèle R_Θ fixé, la fonction 15.2.5 de débit-distorsion opérationnelle sous hypothèse de haute-résolution est simplement une droite donnant le débit en fonction de $\log_2 12D$. Minimiser $\eta\left(\boldsymbol{s}, \boldsymbol{x}, \bar{\Theta}\right)$ revient ainsi à trouver la configuration qui produit la plus basse de ces courbes.

Si les débits d'utilisation ciblés ne correspondent pas nécessairement à ceux pour lesquels l'analyse haute-résolution ci-dessus est pleinement opérationnelle, une stratégie alternative est de tout simplement choisir les paramètres quantifiés qui mènent expérimentalement à la meilleure distorsion pour les débits visés par l'application. Par ailleurs, cette optimisation peut soit être faite pour chaque fichier à encoder, soit une fois pour toutes sur une base de données d'apprentissage.

15.3 Algorithme de codage de source

Une fois les paramètres du modèle $\bar{\Theta}$ estimés et quantifiés, on peut calculer la distribution *a posteriori* $p\left(\boldsymbol{s}(f,n,\cdot) \mid \boldsymbol{x}(f,n,\cdot), \bar{\Theta}\right)$ du vecteur de sources $\boldsymbol{s}(f,n,\cdot)$, de dimension $J \times 1$, pour chaque point (f,n). Compte tenu du modèle PGLS choisi, cette distribution est gaussienne et donnée par l'expression 6.1.19 page 89. De plus, tous ces vecteurs sont supposés indépendants.

Par conséquent, on dispose à ce stade d'une séquence de NF vecteurs complexes indépendants, dont on connait les distributions : une gaussienne complexe circulaire de moyenne $\boldsymbol{\mu}_{\text{post}}(f,n,\cdot)$ définie en 6.1.20 et de covariance $K_{\text{post}}(f,n,\cdot,\cdot)$ définie en 6.1.21 page 89. Dans ces conditions, leur partie imaginaire $\text{Im}\left[\boldsymbol{s}(f,n,\cdot)\right]$ et leur partie réelle $\text{Re}\left[\boldsymbol{s}(f,n,\cdot)\right]$ sont indépendantes, de même covariance $K_{\text{post}}(f,n,\cdot,\cdot)$ et de moyennes $\text{Im}\left[\boldsymbol{\mu}_{\text{post}}(f,n,\cdot)\right]$ et $\text{Re}\left[\boldsymbol{\mu}_{\text{post}}(f,n,\cdot)\right]$, respectivement [5].

On peut donc considérer le signal à encoder comme une séquence de $2NF$ vecteurs gaussiens réels dont on connait moyenne et covariance. Tout indique alors l'utilisation de l'algorithme 13.1 de codage d'une séquence de vecteurs gaussiens, optimal sous hypothèse haute-résolution.

On peut résumer l'ensemble des opérations menées par le codeur CISS dans l'algorithme 15.1.

15.4 Algorithme de décodage

Les opérations menées lors du décodage d'un flux de donnée CISS sont beaucoup plus simples qu'à l'encodage. Elles consistent à recevoir le flux, décoder le modèle $\bar{\Theta}$ et utiliser les distribu-

5. Dans le cas où la représentation fréquentielle utilisée est réelle, on n'a pas à concaténer comme ici les deux séquences des parties réelles et imaginaires pour l'encodage.

Algorithme 15.1 Opérations effectuées par un encodeur CISS.

Entrées
- Signaux source $\tilde{\boldsymbol{s}}$ et mélanges $\tilde{\boldsymbol{x}}$, de dimensions $L \times J$ et $L \times I$.
- Paramètre de qualité du modèle : K et Δ_θ pour NTF, qualité JPEG pour CI.
- Distorsion moyenne maximale autorisée D.

Initialisation
- Calculer les TFCT $\boldsymbol{s}$ et $\boldsymbol{x}$ de $\tilde{\boldsymbol{s}}$ et $\tilde{\boldsymbol{x}}$, de dimensions $F \times N \times J$ et $F \times N \times I$.

Apprentissage et codage du modèle de sources
- Pour le modèle NTF, apprendre le modèle de source θ en utilisant l'algorithme 4.1. L'encoder en utilisant une des techniques vues en section 10.3.3. Par exemple par codage de Huffman d'une quantification uniforme des $\log W$, $\log H$, $\log Q$.
- Pour le modèle CI, effectuer une compression de chaque log-spectrogramme $\log v\left(\cdot,\cdot,j\right)$ des sources en utilisant l'algorithme JPEG et le paramètre de qualité choisi. Idéalement, éviter d'utiliser la quantification perceptuelle de JPEG mais une quantification uniforme.
- Dans tous les cas, construire les DSP estimées $\mathcal{P}\left(\cdot \mid \bar{\Theta}\right)$ des sources.

Apprentissage et codage du modèle de mixage
- Apprendre les filtres de mélange en utilisant les techniques proposées dans 10.2.2 ou 11.4.
- Les quantifier uniformément sur 32 bits et les encoder en utilisant un codage de Huffman.

Calcul des distributions *a posteriori* des sources et codage arithmétique
- Pour chaque point (f,n), calculer la distribution *a posteriori* $p\left(\boldsymbol{s}\left(f,n,\cdot\right) \mid \boldsymbol{x}\left(f,n,\cdot\right),\bar{\Theta}\right)$ selon 6.1.19.
- Utiliser ces distributions pour appliquer l'algorithme de codage 13.1 à la séquence des parties réelles et imaginaires de $\boldsymbol{s}\left(f,n,\cdot\right)$, en utilisant $\frac{D}{2}$ comme seuil de distorsion.

Sortie
- Flux binaire correspondant au modèle quantifié $\bar{\Theta}$ ainsi qu'aux sources quantifiées $\bar{\boldsymbol{s}}$.

tions $p\left(\boldsymbol{s} \mid \boldsymbol{x},\bar{\Theta}\right)$ alors disponibles pour décoder le flux signal, comme cela est résumé dans l'algorithme 15.2. Les algorithmes 15.1 et 15.2 seront testés et évalués au chapitre 16.

Pour finir ce chapitre sur le détail des traitements effectués par le codeur et le décodeur CISS, je mentionne qu'en cas de singularité des matrices de covariance *a posteriori* des sources obtenues selon 6.1.21, il peut y avoir des problèmes en pratique. En effet, une singularité de la matrice de covariance peut conduire à une probabilité nulle pour certaines observations et donc à des soucis numériques lors du codage arithmétique de la séquence.
Une telle situation peut tout d'abord survenir si la distribution *a posteriori* des sources décrit mal les observations. Cela peut arriver si le modèle de source n'est pas suffisamment précis ou si le modèle de mélange convient mal au mixage réellement effectué. Il est par ailleurs possible que la matrice de covariance *a posteriori* soit singulière. Par exemple, si le mélange est la simple somme des sources, il n'est nécessaire que de coder $J-1$ sources, puisque la dernière est donnée par soustraction des autres du mélange. Cela se traduit par une matrice de covariance *a posteriori* singulière.
Pour prendre en compte cette possibilité, une stratégie simple que j'ai utilisée a été de toujours considérer :

$$\hat{\lambda}_{fn,j} = \max\left(\epsilon, \lambda_{fn,j}\right), \tag{15.4.1}$$

où ϵ est un réel positif, fixé à une faible valeur comme $\epsilon = 10^{-5}$.

Algorithme 15.2 Opérations effectuées par un décodeur CISS.

Entrées

– Signaux de mélanges $\tilde{\boldsymbol{x}}$, de dimensions $L \times I$.
– flux binaire signal et modèle

Initialisation et décodage du modèle

– Calculer la TFCT $\boldsymbol{x}$ de $\tilde{\boldsymbol{x}}$, de dimension $F \times N \times I$
– Décoder le flux binaire modèle de manière à reconstruire le modèle quantifié $\bar{\Theta}$

Décodage des sources

– Pour chaque point (f, n), construire les distributions $p\left(\boldsymbol{s}\left(f,n,\cdot\right) \mid \boldsymbol{x}\left(f,n,\cdot\right), \bar{\Theta}\right)$, comme des gaussiennes de moyenne $\bar{\boldsymbol{\mu}}_{\text{post}}\left(f,n,\cdot\right)$ et de covariance $\bar{K}_{\text{post}}\left(f,n,\cdot,\cdot\right)$ définies en 6.1.20 et 6.1.21 page 89 respectivement, dont les dimensions sont $J \times 1$ et $J \times J$.
– Calculer la décomposition en valeurs propres de $\bar{K}_{\text{post}}\left(f,n,\cdot,\cdot\right)$:

$$\bar{K}_{\text{post}}\left(f,n,\cdot,\cdot\right) = U_{fn}\text{diag}\left[\lambda_{fn,1},\ldots,\lambda_{fn,J}\right]U_{fn}^{H},$$

où U_{fn} est de dimension $J \times J$
– Décoder le flux binaire source comme produit par l'encodage arithmétique d'une séquence de $2NFJ$ scalaires $\text{Re}\left[\bar{\boldsymbol{\xi}}\left(f,n,j\right)\right]$ et $\text{Im}\left[\bar{\boldsymbol{\xi}}\left(f,n,j\right)\right]$, distribués selon des lois gaussiennes centrées de variance $\lambda_{fn,j}$.
– Reconstruire les signaux sources comme :

$$\bar{\boldsymbol{s}}\left(f,n,\cdot\right) = U_{fn}\bar{\boldsymbol{\xi}}\left(f,n,j\right) + \boldsymbol{\mu}\left(f,n,\cdot\right).$$

– Reconstruire les signaux sources estimés $\bar{\tilde{\boldsymbol{s}}}$ par TFCT inverse.

Sortie

– Sources estimées $\bar{\tilde{\boldsymbol{s}}}$

Chapitre 16

Évaluation

16.1 Introduction

Dans ce chapitre, je présente les résultats des différentes campagnes d'évaluation que j'ai menées sur le système CISS et qui ont fait l'objet de plusieurs publications [160, 136, 161]. Alors que l'étude expérimentale de [160] a surtout consisté à montrer la validité de l'approche CISS proposée sous forme de tests préliminaires, celles menées dans [136] et [161] sont plus approfondies et sont basées sur la même méthodologie que celle présentée au chapitre 12 dans le cas paramétrique.

CISS a été évalué sur la même base de donnée que les systèmes paramétriques de la partie III, en utilisant les mêmes métriques.

Les performances de CISS ont été testées sur un ensemble de 7 extraits musicaux de 30 secondes, échantillonnés à 44.1kHz et dont on dispose de toutes les pistes séparées monophoniques. De ces pistes séparées ont été produits des mélanges. Dans un premier cas, les sources ont simplement été sommées pour produire l'unique mélange ($I = 1$). Dans le deuxième cas, les sources ont été mixées dans des mélanges stéréophoniques ($I = 2$) soit de manière instantanée, soit en utilisant les mêmes filtres de mélange HRTF [1] que pour l'évaluation du système paramétrique en section 12.3.2 page 154.

De la même manière que pour le cas paramétrique, les métriques considérées sont δ_{SDR} et δ_{PSM}, présentées en section 12.2 page 151 et qui permettent d'évaluer les performances relatives du système proposé par rapport à une séparation paramétrique dans sa configuration oracle 9.1.7 page 122. Il y a deux intérêts principaux supplémentaires à considérer ces métriques différentielles dans le cas de CISS, en plus de leur propriété de normalisation d'une difficulté de séparation variable. Le premier est qu'elles permettent de voir que CISS a des meilleures performances que le système paramétrique dès lors que ces scores sont positifs. En effet, $\delta_{SDR} > 0$ ou $\delta_{PSM} > 0$ indique une qualité meilleure que la borne supérieure atteignable par le système paramétrique. Le deuxième avantage est qu'utiliser les mêmes métriques sur les mêmes données rend possible une comparaison des résultats de l'évaluation.

Dans le cas d'un seul mélange instantané ($I = 1$), j'ai également procédé à l'évaluation des performances de CISS si la représentation fréquentielle utilisée est la TFCT ou bien la MDCT.

Tous les débits qu'on trouvera dans cette évaluation sont les débits théoriques calculés en utilisant l'expression 15.2.5 page 197 du débit total. En effet, je ne dispose pas d'une implémentation concrète d'un codeur arithmétique tel que celui requis par l'algorithme 13.1 page 181 d'encodage d'une séquence de vecteurs gaussiens. Il est cependant admis dans la littérature que les implémentations de tels codeurs permettent d'effectivement atteindre les performances théoriques [76]. Ceci a été vérifié informellement par Alexey Ozerov qui dispose d'une telle implémentation réservée à un usage privé. Je réserve cependant à l'analyse de vrais flux binaires le soin d'indiquer dans quelle mesure les performances obtenues en pratique sont identiques aux valeurs théoriques données ici.

1. *Head Related Transfer Function* : ce sont des filtres de mélange de longueur impulsionnelle $H = 200$, permettant une spatialisation réaliste des sources dans l'espace stéréophonique. La base de donnée utilisée est HRIR [4].

Enfin, les résultats sont encore présentés ici sous la forme de courbes débit-qualité, obtenues en effectuant un lissage par LOESS [36] du nuage de points (R_{tot}, δ_{SDR}) obtenu sur l'ensemble des fichiers de la base.

16.2 Choix des paramètres du modèle

J'ai procédé aux évaluations de CISS pour le seul modèle de sources NTF. Le système résultant est nommé CISS-NTF et dispose de deux paramètres de modèle que l'utilisateur doit régler et qui sont :

- Le nombre K de composantes du modèle NTF
- Le pas de quantification Δ_θ à utiliser pour la quantification du modèle NTF si on considère la quantification MMG vue en section 10.3.3 page 135.

Le choix d'un couple (K, Δ_θ) produit un lot de paramètres quantifiés $\bar{\Theta}$ auquel correspond un débit de modèle R_Θ. Dans l'évaluation de CISS-NTF faite dans [161], nous avons étudié en détail les différentes approches permettant de choisir ces paramètres. Dans cette section, je présente les résultats de cette étude, faite sur des mélanges monophoniques ($I = 1$) obtenus comme la simple somme des sources.

La première manière de choisir le lot (K, Δ_θ) que j'ai envisagée est celle présentée en section 15.2 page 195 qui repose sur la théorie haute-résolution. Pour chaque fichier, on peut calculer le score 15.2.6 page 197 pour toutes les combinaisons (K, Δ_θ) et choisir comme paramètres optimaux ceux qui produisent la plus petite valeur pour ce critère. Ainsi, pour chacun des 7 fichiers considérés, j'ai calculé ce score pour toutes les combinaisons de couples (K, Δ_θ) avec :

En utilisant le critère fourni par l'analyse haute-résolution faite en section 15.2, les paramètres optimaux pour le modèle NTF sont de 5 composantes par source, avec un pas de quantification de $\Delta_\theta = 0.1$ pour l'encodage des $\log W$, $\log H$ et $\log Q$.

- $K/J = [2, 3, 4, 5, 10, 15, 20, 30]$: nombre de composantes NTF par source, soit 8 possibilités.
- $\Delta_\theta = [1.8,\ 0.5,\ 0.13,\ 0.04,\ 0.01]$: pas de quantification du modèle NTF, soit 5 possibilités.

Sur la figure 16.1, j'ai représenté la moyenne sur les 7 fichiers testés des scores ainsi obtenus pour chacune des $8 \times 5 = 40$ combinaisons (K, Δ_θ) considérées. De cette analyse, il ressort qu'en moyenne, sur l'ensemble de la base, les paramètres optimaux selon la théorie haute-résolution sont $K/J \approx 5$ composantes NTF par sources pour un pas de quantification $\Delta_\theta \approx 0.1$. On peut mettre ce résultat en perspective avec celui donné dans [155], où c'est la valeur $K/J = 4$ qui avait été trouvée optimale pour une séparation semi-informée [2] utilisant le modèle NTF. Il semblerait donc que ce sont des valeurs similaires des paramètres qui conviennent au codage ou à la séparation des sources.

En parallèle de cette optimisation théorique des paramètres du modèle, j'ai aussi mené une optimisation expérimentale. Pour chaque fichier de la base et chacune de ces 8×5 configurations (K, Δ_θ) pour le modèle (et donc pour les $\bar{\Theta}$ et R_Θ correspondants), j'ai calculé le débit requis par CISS pour 11 valeurs différentes de distorsion logarithmiquement espacées entre $D = 0.7$ et $D = 310$, conduisant pour chaque fichier à la valeur du débit total $R_{\text{tot}}(K, \Delta_\theta, D)$ pour chacune des $8 \times 5 \times 11$ configurations de CISS-NTF correspondantes [3].

À partir de ces observations, j'ai considéré 5 différentes stratégies pour le choix des paramètres (K, Δ_θ) :

2. Au sens précisé au chapitre 7 d'une méthode qui fait des hypothèses fortes sur la structure des DSP des sources, mais qui en apprend les paramètres sur les seuls mélanges.

3. Cette étude a mené au calcul des scores des pistes séparées des 7 morceaux pour $8 \times 5 \times 11 = 440$ configurations différentes de CISS, pour chacune des représentations TFCT et MDCT considérées. Cela correspond à près de 10000 combinaisons possibles de CISS, pour lesquelles j'ai calculé tous les scores δ_{PSM} et δ_{SDR}. Ce travail a nécessité plusieurs mois de calculs.

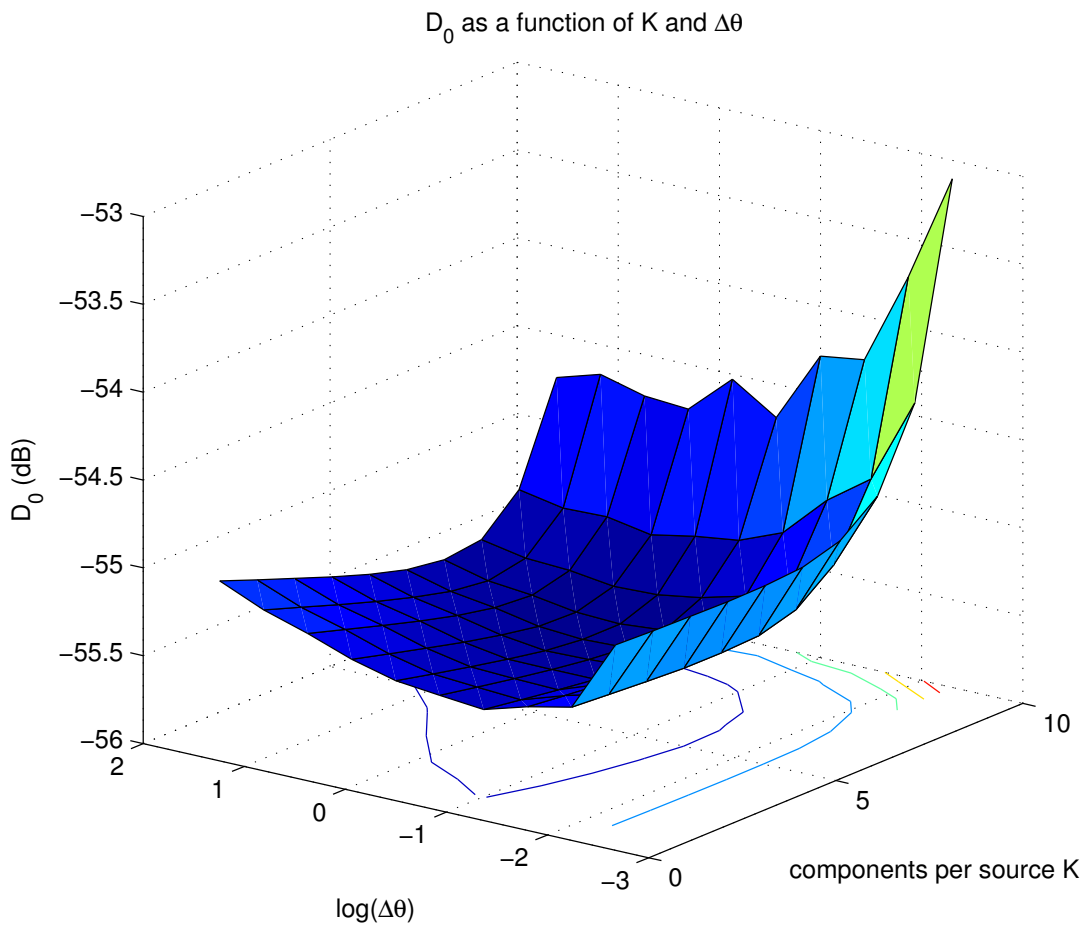

FIGURE 16.1: Scores $\eta\left(\boldsymbol{s}, \boldsymbol{x}, \bar{\Theta}\right)$ 15.2.6 obtenus en moyenne pour l'ensemble des 280 configurations testées, ramenés à une distorsion équivalente à débit total nul D_0 selon $0 = \eta\left(\boldsymbol{s}, \boldsymbol{x}, \bar{\Theta}\right) - JFN \log_2 12D_0$. Comme on le voit, les paramètres optimaux sont dans l'ensemble $K/J \approx 5$ composantes par source et $\Delta_\theta \approx 0.1$.

- **[Opt-HR-avg]** Le lot (K, Δ_θ) qui minimise la distorsion à haut débit sur l'ensemble de la base est déterminé. Une fois cette valeur apprise, on la fixe pour l'utilisation de CISS-NTF. De manière intéressante, les valeurs trouvées par cette étude expérimentale sont similaires à celles trouvées par l'analyse théorique : $K/J \approx 5$ et $\Delta_\theta \approx 0.1$.
- **[Opt-LR-avg]** Identique à [Opt-HR-avg], hormis le fait que c'est le lot (K, Δ_θ) qui minimise la distorsion à bas débit ($R_{\text{tot}} \leq$ 5kbps) sur l'ensemble de la base qui est choisi. Dans ce cas, on constate que moins de composantes NTF sont requises avec $K/J \approx 2$, mais qu'elles doivent être quantifiées avec une précision similaire de $\Delta_\theta \approx 0.1$. Ce comportement signifie qu'à très bas débit, il vaut mieux utiliser un modèle des sources un peu moins bon mais pouvoir dépenser plus de débit sur l'encodage du signal.
- **[Opt-HR-mix]** Pour chaque fichier, on identifie le lot (K, Δ_θ) qui minimise la distorsion à haut débit. Une fois cette valeur apprise, on la fixe pour ce fichier.
- **[Opt-LR-mix]** Identique à [Opt-HR-mix], sauf que l'optimisation se fait à bas débit.
- **[Opt-System]** Pour chaque débit et chaque fichier, on garde le lot de paramètres (K, Δ_θ) qui donne la distorsion la plus faible. Cette stratégie est optimale, mais elle conduit à la nécessité d'effectuer énormément de calculs pour l'encodage d'un seul fichier. Je l'utilise surtout comme une borne en fonction de laquelle comparer les performances des autres stratégies.

On trouvera les courbes débit-qualité obtenues par chacune de ces 5 stratégies de choix des paramètres, lissées sur l'ensemble de la base, en figure 16.2 page suivante.

Je reviendrai sur l'allure générale de ces courbes en section suivante. Pour l'heure, je me contente de discuter des performances relatives constatées pour les différentes stratégies de choix des paramètres. En observant la figure 16.2, on constate pour commencer que le gain en performance engendré par l'optimisation des paramètres fichier par fichier est faible (stratégies [Opt-(H/L)R-mix]) par rapport à une optimisation globale (stratégies [Opt-(H/L)R-avg]). De plus, toutes ces courbes sont assez proches des performances du système optimal [Opt-System]. Il s'agit d'une bonne nouvelle dans la mesure où une stratégie d'optimisation globale permet en pratique de fixer une fois pour toutes les paramètres K/J et Δ_θ sans procéder à de nouvelles optimisations pour le traitement d'un fichier inconnu.

Au vu des résultats de la figure 16.2, il paraît clair par ailleurs que la question d'une optimisation à bas débit ou à haut débit dépend de l'application visée. Comme on peut le constater, les optimisations à haut débit ne peuvent pas fonctionner à des débits inférieurs à 2kbps/source, qui correspond à la portion irréductible R_Θ allouée au modèle. Les optimisations à bas débit, au contraire, se contentent de modèles plus petits et peuvent descendre jusqu'à 500 bits/s/source. En termes de performance, on constate que jusqu'à 7kbps/source environ, les performances des configurations optimisées à bas débit sont légèrement meilleures que les autres pour δ_{SDR}. Par contre, les performances de la stratégie [Opt-HR-avg] s'avèrent toujours supérieures pour δ_{PSM}, suggérant une meilleure qualité perceptive. La différence est cependant faible.

Enfin, la figure 16.2 permet de comparer l'influence de la technique d'encodage des paramètres NTF. Dans le système Wiener-NTF-2011, les paramètres NTF sont encodés de la même manière que dans [130], c'est-à-dire par une quantification uniforme des paramètres W, H et Q, puis par un encodage de Huffman. J'ai montré en section 10.3.3 page 135 que cette stratégie n'est pas optimale puisque sur une base théorique, on montre que ce sont plutôt les $\log W$, $\log H$ et $\log Q$ qui doivent être quantifiés uniformément. Le système Wiener-NTF-log-Q implémente une telle stratégie, suivie d'un encodage par modèle de mélange de gaussiennes d'un pas $\Delta_\theta = 0.1$. On constate que les performances de cette méthode sont bien meilleures et permettent de gagner presque 15kbps/source en débit pour la même distorsion.

En conséquence, de bons choix de paramètres à utiliser dans le cas du modèle NTF pour CISS sont :
- Pour une utilisation à bas débit ($R_{\text{tot}} < 2$kbps/source) : paramètres fixés à $K/J = 2$ et encodage par MMG avec $\Delta_\theta = 0.1$.
- Pour une utilisation à haut débit ($R_{\text{tot}} > 2$kbps/source) : paramètres fixés à $K/J = 5$ et encodage par MMG avec $\Delta_\theta = 0.1$, qui sont ceux obtenus également par une analyse haute-résolution.

16.3 Résultats

Ayant montré qu'il est possible d'optimiser une fois pour toutes les paramètres à haut ou bas débit selon les applications et qu'il faut les quantifier après une compression logarithmique, je peux aborder la question des performances relatives de CISS-NTF par rapport aux techniques paramétriques proposées en partie III et désignées par Wiener-NTF-2011 ou Wiener-JPG-2011 en fonction du modèle de sources NTF ou CI choisi. Une fois encore, Wiener-NTF-2011 correspond ici à la technique d'encodage de [130], où le modèle est quantifié sans compression logarithmique.

Cette comparaison a fait l'objet de deux études distinctes, présentées dans [161] et [136]. Dans la première, nous nous sommes concentrés sur le cas d'un seul mélange ($I = 1$), obtenu comme une simple somme des sources. Les distributions *a posteriori* utilisées sont donc celles de la section 5.3 page 79. Nous nous sommes aussi intéressés dans ce cas à l'influence du choix de la représentation fréquentielle utilisée : TFCT ou MDCT. Les résultats de cette étude sont donnés en figure 16.3 page 206.

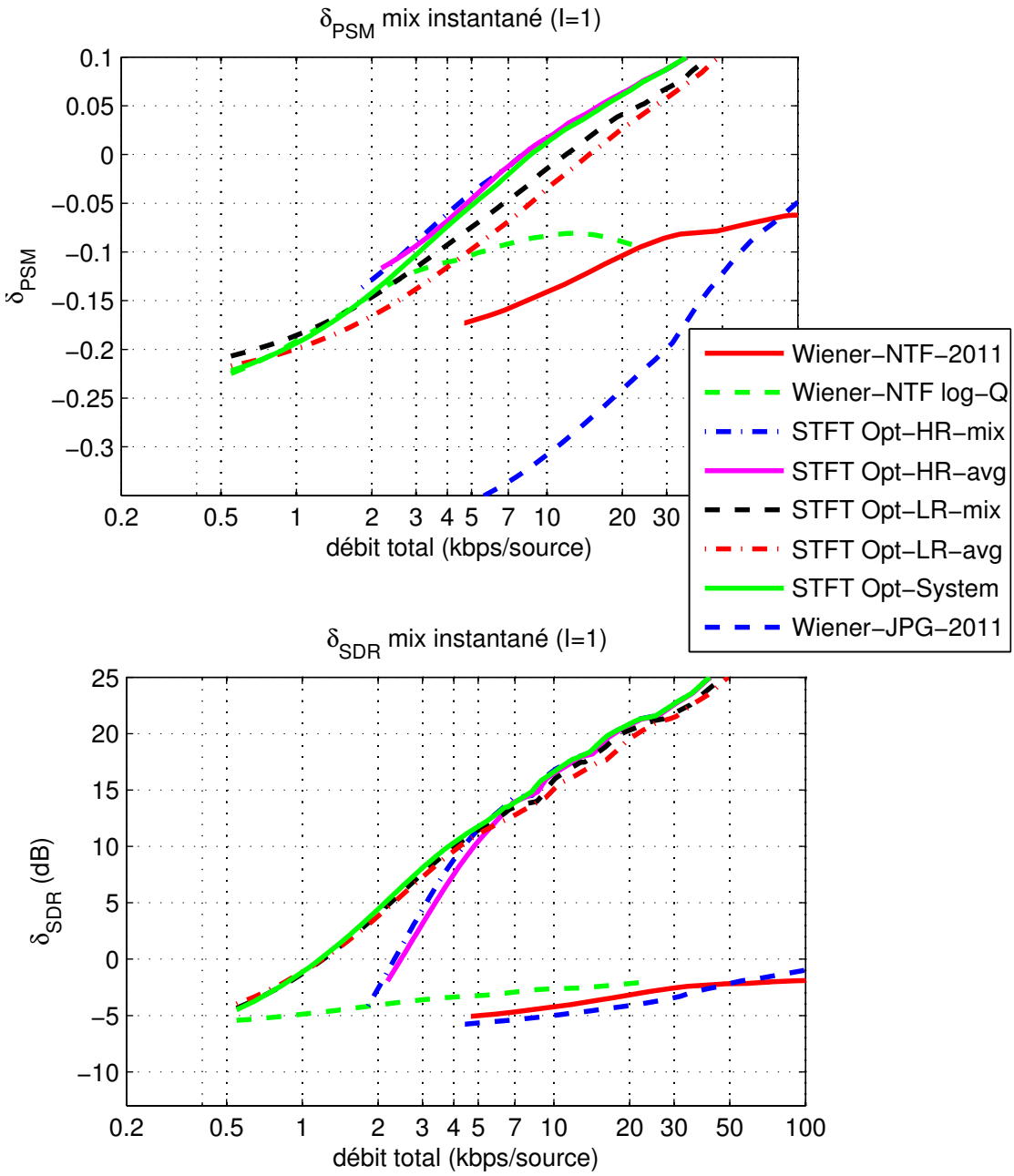

FIGURE 16.2: Courbes de débit-qualité obtenues par les différentes stratégies de choix de paramètres pour CISS. On voit qu'en pratique, l'optimisation à bas débit **[Opt-LR-avg]** fournit un bon compromis entre la performance et la complexité d'utilisation : il suffit de fixer $K/J = 2$ et $\Delta_\theta = 0.1$. Wiener-NTF-2011 et Wiener-JPG-2011 désignent la technique paramétrique proposée en partie III avec les modèles de source NTF et CI, respectivement. Pour Wiener-NTF-2011, les paramètres NTF W, H et Q sont quantifiés uniformément comme dans [130]. Pour le système Wiener-NTF-log-Q, ce sont les $\log W$, $\log H$ et $\log Q$ qui sont quantifiés uniformément comme le suggère l'analyse théorique en section 10.3.3 (d'après [161]).

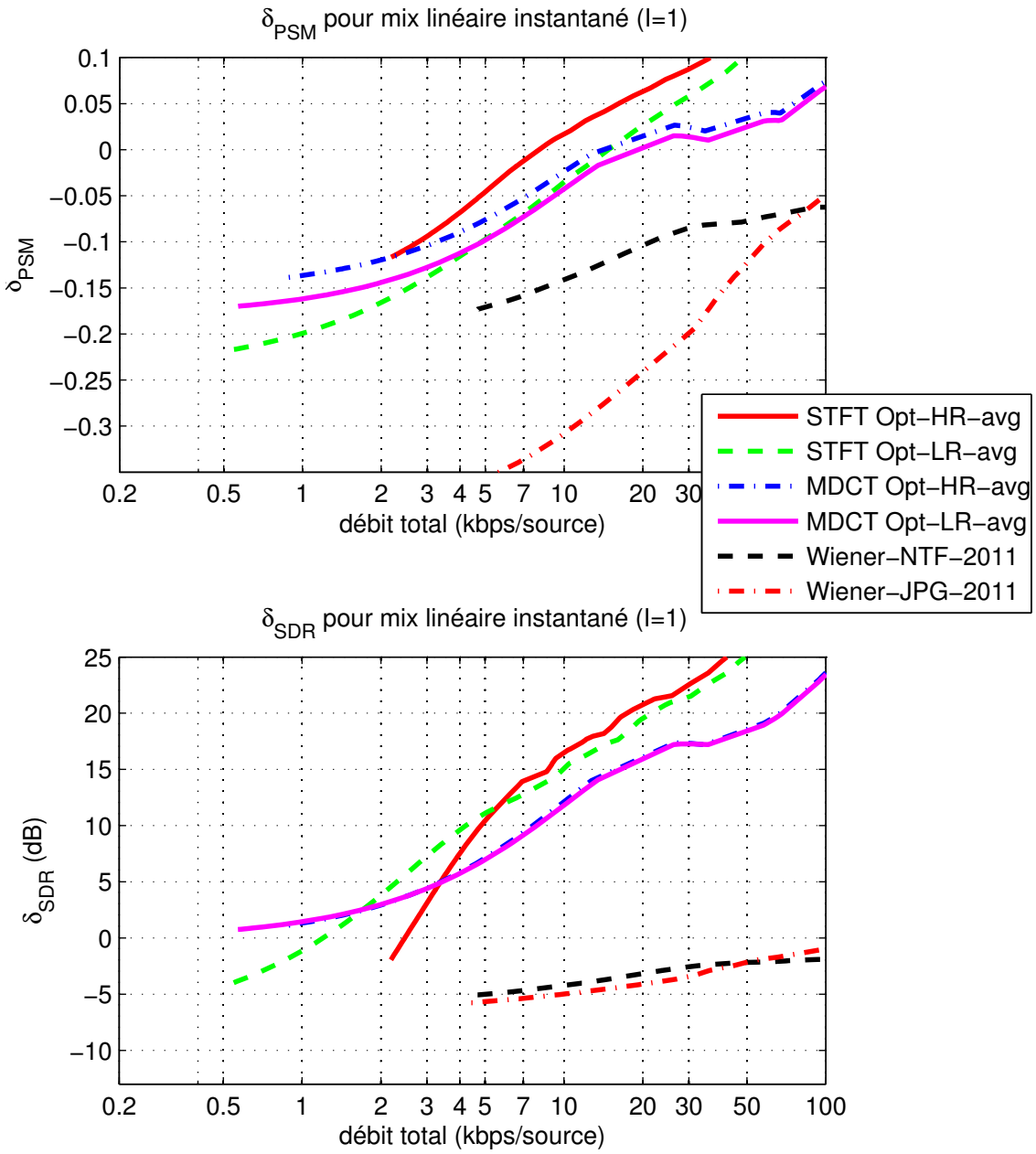

FIGURE 16.3: Courbes de débit-qualité obtenues sur un mélange monocanal ($I = 1$) linéaire instantané, par les deux stratégies d'optimisation globale des paramètres [Opt-HR-avg] et [Opt-LR-avg] de CISS-NTF en utilisant une représentation TFCT ou MDCT. La valeur de référence 0 est le score obtenu par l'oracle paramétrique en TFCT. Les scores des techniques paramétriques Wiener-NTF-2011 et Wiener-JPG-2011 sont aussi représentés. Chaque courbe correspond au lissage par LOESS [36] d'un nuage de points. Les gains en performances produits par CISS sont très importants, à tous débits (d'après [161]).

Dans la deuxième étude [136], j'ai considéré le cas de mélanges stéréophoniques ($I = 2$) obtenus soit par mixage instantané, soit par mixage convolutif de la manière précisée en section 12.3.2 page 154. Pour CISS-NTF, j'ai choisi $K/J = 3$ composantes par source et une quantification uniforme sur 32 bits des paramètres compressés logarithmiquement, suivie d'un encodage de Huffman. Les résultats correspondants sont donnés en figure 16.4.

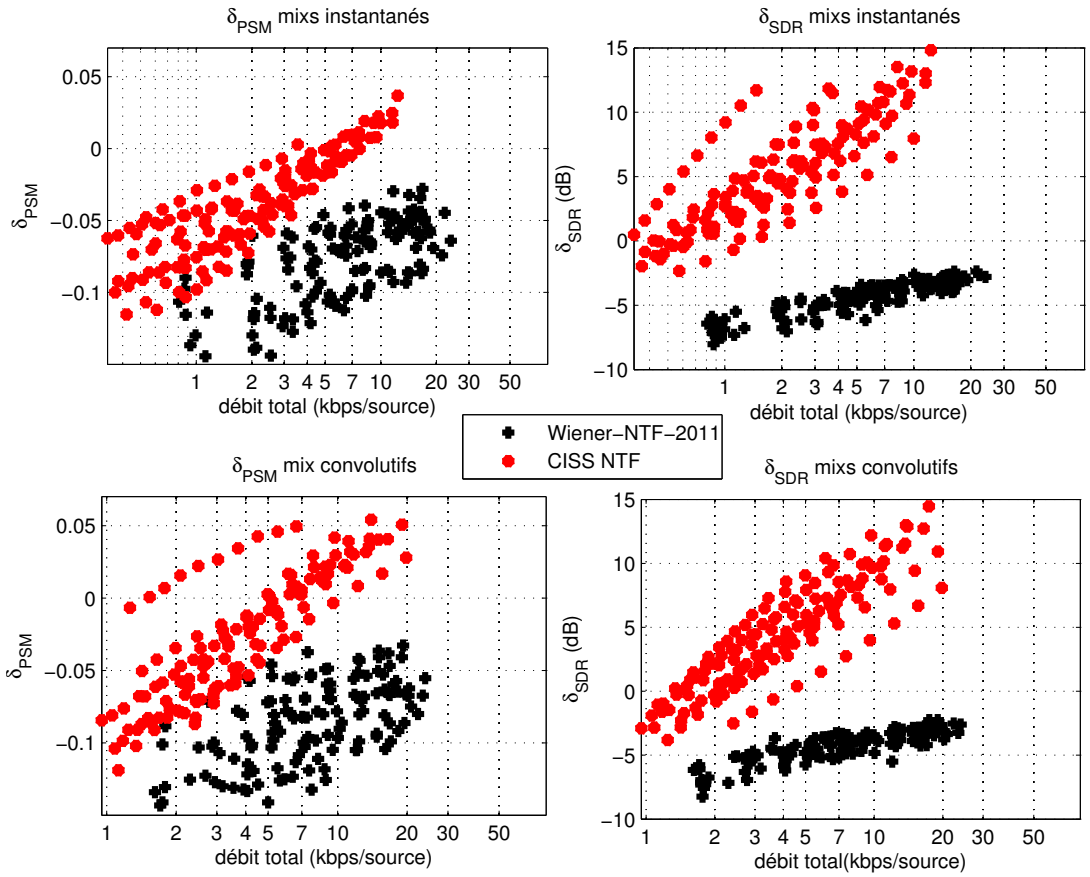

FIGURE 16.4: Courbes de débit-qualité obtenues sur des mélanges multicanaux ($I > 1$) mixés de manière linéaire instantanée ou convolutive. Les scores de Wiener-NTF-2011 et CISS-NTF sont représentés. Chaque lot de points indique les résultats obtenus pour un morceau donné. Les gains en performances produits par CISS sont encore très importants, à tous débits (d'après [136]).

16.4 Discussion

Des résultats des figures 16.3 et 16.4, plusieurs points forts émergent.

Tout d'abord, les performances de CISS ne sont pas bornées. On constate en effet qu'un gain en débit provoque nécessairement une augmentation de la qualité des résultats. Cela est naturel en vue de la fonction de débit-distorsion opérationnelle 15.2.5 page 197 donnée par la théorie haute-résolution :

$$R_{\text{tot}} = R_\Theta - \log_2 p\left(\boldsymbol{s} \mid \boldsymbol{x}, \bar{\Theta}\right) - JFN \log_2 12D,$$

qui prévoit une dépendance linéaire entre le débit et la distorsion en dB pour des hauts débits. Ce comportement est clairement visible sur l'ensemble des courbes débit-qualité de CISS. Un des avantages majeurs de CISS est donc de pouvoir garantir une distorsion quelconque, pour peu que le débit disponible soit suffisant.

Ensuite, les performances de CISS dépassent très rapidement celles de l'oracle paramétrique ($\delta_{SDR} > 0$ ou $\delta_{PSM} > 0$). Selon le critère SDR, ce dépassement s'opère dès 2kbps/source, tandis qu'il se produit plus tard pour δ_{PSM}, vers 10kbps/source [4].

Quoiqu'il en soit, pour un débit d'environ 5kbps/source, les performances de CISS dépassent celles de Wiener-NTF-2011 et de Wiener-JPG-2011 d'au moins 10dB de SDR et d'au moins 0.1 en PSM, ce qui est considérable. Cet écart ne fait qu'augmenter avec le débit et on constate que le gain minimal en distorsion à débit équivalent est de 5dB de SDR à l'avantage de CISS, pour des débits inférieurs à 2kbps/source. Quand on met en perspective ces résultats par rapport à ceux du chapitre 12 où les systèmes paramétrique Wiener-NTF et Wiener-JPG sont apparus comme déjà compétitifs, on mesure l'intérêt qu'il y a à considérer de préférence CISS pour la séparation informée.

CISS apparait comme largement plus performant que les système paramétriques évalués au chapitre 12. Il faut cependant tempérer ce constat par le fait que je ne dispose pas encore d'un codeur CISS effectif, puisque tous les débits calculés sont théoriques et non pas obtenus par une véritable implémentation de l'algorithme 13.1 d'encodage arithmétique. Cependant, de tels codeurs ont été implémentés par le passé [76], qui atteignent leurs performances théoriques.

Comme on peut le constater sur la figure 16.3, les avantages de CISS se confirment pour des mélanges stéréophoniques, qu'ils soient instantanés ou convolutifs. Dans le cas convolutif, c'est l'algorithme 11.2 page 146 d'estimation des paramètres de mixage qui a été utilisé. Pour le cas instantané, on constate par ailleurs sur la figure 16.3 qu'à bas débit, on obtient de meilleures performances si on utilise la MDCT comme représentation temps-fréquence [5]. Cela se comprend par le fait que la MDCT produit une représentation qui est réelle d'une part et d'autre part qu'elle est à décimation critique, contrairement à la TFCT qui présente des redondances. Cependant, le cas des mélanges convolutifs ne peut pas se traiter avec la MDCT selon les techniques présentées en section 6.1.3 page 87, puisque l'approximation $\boldsymbol{x}(f,n,\cdot) \approx A(f)\,\boldsymbol{s}(f,n,\cdot)$ ne tient pas dans ce cas. Rien n'empêche cependant de considérer d'autres modèles plus puissants qui permettent de rendre compte d'un mixage convolutif avec la MDCT [6].

En conclusion, cette évaluation permet d'énoncer les différents avantages et inconvénients de CISS par rapport à une approche paramétrique [7], regroupés dans le tableau 16.1.

4. Dans la mesure où le SDR s'apparente à un critère quadratique, il s'agit d'une métrique très proche de celle optimisée par CISS. Cela explique ce retard de PSM. La prise en compte d'une distorsion perceptive peut se faire de manière naturelle avec CISS et fait l'objet de recherches en cours.

5. Il serait même attendu d'obtenir aussi de meilleures performances pour la MDCT à haut débit dans le cas d'un mélange instantané. Je pense que ce n'est pas ce que j'ai observé à cause d'une valeur trop élevée de ϵ dans 15.4.1 page 198 choisie pour la MDCT par rapport à celle choisie pour TFCT. Je n'ai pas eu le temps de vérifier cette hypothèse.

6. Je pense en particulier au modèle HR-NMF [11].

7. Du fait de mon enthousiasme, il y a sans doute quelques autres inconvénients à CISS qui ne me viennent pas à l'esprit.

Avantages de CISS

- A tout débit, les performances de CISS sont largement supérieures à celles d'un système paramétrique basé sur le même modèle de source.
- Les performances de CISS ne sont pas bornées et dépassent très rapidement celles des estimateurs paramétriques oracles.
- Le système CISS présenté permet de traiter des mélanges multicanaux convolutifs.
- Il est possible de déterminer pour un débit fixé quelle sera la distorsion moyenne observée et réciproquement de déterminer le débit nécessaire pour garantir une distorsion donnée.
- CISS peut être étendu naturellement pour l'optimisation d'une distorsion perceptive et non plus quadratique.
- Le principe du codage *a posteriori* qu'exploite CISS n'est pas limité au formalisme gaussien présenté aux parties I et II. Tout nouveau modèle probabiliste permettant d'obtenir une distribution *a posteriori* des sources peut être utilisé.

Inconvénients de CISS

- L'utilisation de CISS en pratique nécessite l'implémentation d'un encodeur arithmétique encodant une suite de scalaires gaussiens selon l'algorithme 13.1.
- Telle que présentée, la procédure de codage de CISS fait appel à un apprentissage génératif sous-optimal. Il pourrait être intéressant de considérer un autre modèle permettant un apprentissage discriminant.
- Il ne parait pas évident de générer avec CISS un flux binaire compatible avec le standard SAOC.

TABLE 16.1: Avantages et inconvénients de CISS

Conclusion de la quatrième partie

De manière à outrepasser les limites intrinsèques des systèmes paramétriques pour la séparation informée, j'ai montré qu'un angle d'approche adéquat est celui de la théorie du codage de source. En effet, cette théorie se concentre sur la façon la plus concise de coder un signal de telle manière à ce que sa reconstruction présente avec l'original une distorsion inférieure à un seuil donné.

Le codage arithmétique permet d'encoder des séquences de variables aléatoires dont les lois sont différentes.

Pour commencer, j'ai présenté dans le chapitre 13 les différents résultats de la théorie du codage de source qui m'ont été utiles dans mon travail. Ainsi, j'ai rappelé les définitions générales de l'entropie et de l'information mutuelle pour des variables aléatoires discrètes. J'ai aussi rappelé le théorème du codage de source, qui établit un débit minimal limite nécessaire au codage sans perte d'une variable aléatoire discrète. Ensuite, j'ai présenté deux algorithmes classiques de codage sans perte permettant asymptotiquement d'atteindre cette borne. Si j'ai évoqué très rapidement le codage de Huffman, classique, je me suis attardé plus longtemps sur le codage arithmétique. J'ai insisté en particulier sur le fait que contrairement à l'algorithme de Huffman, le codage arithmétique permet d'encoder de manière asymptotiquement optimale une séquence de réalisations indépendantes de variables aléatoires discrètes de lois *différentes*.

C'est alors que je me suis intéressé à la théorie débit-distorsion, qui est le pendant du théorème du codage de source pour les variables continues, c'est-à-dire à valeurs dans $\mathbb{R}$. Cette théorie établit des bornes absolues de débit en deçà desquelles il est impossible de garantir une distorsion moyenne inférieure à un certain seuil. J'ai commencé par rappeler la définition générale de cette fonction débit-distorsion, avant d'en donner les expressions analytiques pour des variables aléatoires gaussiennes.

La théorie débit-distorsion indique les bornes absolues en débit en deçà desquelles on ne peut pas garantir une distorsion inférieure à un seuil donné. La théorie haute-résolution fournit une manière pratique d'atteindre cette limite lorsque le débit considéré devient suffisamment grand.

Si elle permet d'établir des bornes absolues en performance pour un système de codage de réalisations de variables aléatoires continues, la théorie débit-distorsion permet rarement de déterminer une démarche concrète pour atteindre ces bornes, bien qu'elle démontre que de telles démarches existent. Pour cette raison, j'ai présenté la théorie *haute-résolution*, qui en fournit un contrepoids pratique. Sous hypothèse de distorsion suffisamment faible, on peut déduire de nombreux résultats qui mènent à une manière concrète de procéder à la quantification des signaux à transférer. Pour faire court, si on dispose de suffisamment de débit, la théorie haute-résolution démontre que le débit à allouer aux paramètres de la distribution des sources est constant et qu'il faut quantifier de manière uniforme les signaux à transmettre pour minimiser le débit. Dans le cas gaussien, on établit en outre des bornes de *débit-distorsion opérationnelles*, valables pour ces régimes de fonctionnement.

De ces considérations, j'ai déduit l'algorithme optimal par lequel il est possible d'encoder une séquence de vecteurs indépendants, dont chacun est la réalisation d'une variable aléatoire gaussienne multivariée dont on connait la loi. Cet algorithme procède d'abord à une décorrélation de chaque vecteur par une transformée de Karhunen-Loeve, puis à une quantification uniforme de toutes les composantes ainsi décorrélées. Enfin, un codage arithmétique permet d'encoder optimalement la séquence de ces valeurs quantifiées.

Le codage *a posteriori* est identique au codage de source, à la différence que la distribution utilisée pour encoder les sources est conditionnée à la connaissance au codeur et au décodeur d'une même information *tierce*.

Une fois la théorie du codage de source présentée, j'ai montré dans le chapitre 14 comment la séparation informée peut être vue comme une instance particulière d'un problème de codage, dit codage de Berger-Flynn-Gray, pour laquelle à la fois le codeur et le décodeur disposent de la même connaissance d'une information *tierce*. J'ai appelé codage *a posteriori* ce cas de figure spécial et j'ai montré qu'il est intéressant pour le codage de prendre en compte cette information tierce de manière à réduire le débit nécessaire. J'ai de plus établi que dans le cas gaussien, le gain en débit d'un codage *a posteriori* sera d'autant plus grand que l'information tierce sera corrélée à la source à transmettre.

En pratique, le codage *a posteriori* revient à ne pas utiliser leur distribution $p(\boldsymbol{s})$ pour encoder les sources, mais plutôt leur distribution *a posteriori* $p(\boldsymbol{s} \mid \boldsymbol{x})$, où $\boldsymbol{x}$ est l'information tierce. Dans le cas de la séparation informée, l'information connue à la fois au codeur et au décodeur est constituée des mélanges des sources. Dans ces conditions, on peut utiliser n'importe laquelle des distributions *a posteriori* données par l'analyse du problème de séparation dans le but d'encoder les sources. Le système correspondant a été nommée séparation informée par codage (CISS).

J'ai montré que cette idée d'encoder les sources en utilisant un algorithme classique de codage et leur distribution *a posteriori* présente de nombreux avantages. Tout d'abord, elle généralise la configuration paramétrique qui se comprend comme le simple choix de la moyenne *a posteriori* pour encoder les sources, ce qui est optimal si la distorsion tolérée est très grande. Ensuite, elle présente l'avantage de disposer de performances non bornées, puisqu'il suffit de rajouter un peu de débit pour garantir une meilleure distorsion. En outre, elle procède à une distribution optimale du débit entre modèle et signal et indique des bornes de fonctionnement qu'un utilisateur peut exploiter pour définir un débit à utiliser. Enfin, elle permet d'utiliser les accomplissements récents du domaine du codage de source, en prenant par exemple en compte un critère perceptif de distorsion.

Si les idées derrière CISS sont séduisantes, il a fallu considérer les opérations concrètes que le codeur et le décodeur correspondants doivent effectuer. C'est ainsi que j'ai montré dans le chapitre 15 que moyennant certaines hypothèses raisonnables, l'estimation d'un modèle pour CISS peut se faire de la même manière que dans le cas paramétrique, ce qui m'a permis d'utiliser dans ce but les mêmes algorithmes.

Ensuite, j'ai abordé le problème de la répartition optimale à choisir entre le débit permettant de transmettre les paramètres des distributions *a posteriori* et le débit requis pour transmettre les source encodées. En utilisant des résultats récents de la théorie haute-résolution, j'ai montré que cette optimisation peut se faire automatiquement et j'ai indiqué la manière opérationnelle de la mener. J'ai conclu sur le détail des algorithmes de codage et de décodage CISS.

Pour finir, j'ai procédé dans le chapitre 16 à une large évaluation des performances de CISS sur les mêmes données et en utilisant les mêmes métriques que pour l'évaluation des systèmes paramétriques. C'est ainsi que j'ai procédé à des tests poussés des performances théoriques de CISS dans le cas de mélanges monophoniques, stéréophoniques, instantanés ou convolutifs.

Sur une large évaluation, CISS permet d'obtenir des excellentes estimées des sources dès 0.5kbps/source et dépasse les bornes théoriques des systèmes paramétriques proposés dès 2kbps/source.

Les résultats de ces évaluations sont extrêmement encourageants, dans la mesure où CISS tient ses promesses d'offrir des performances non bornées qui sont au pire les mêmes à débit signal nul que celles des systèmes paramétriques correspondants. Ainsi, CISS dépasse les bornes théoriques des systèmes paramétriques considérés dès 2kbps/source en termes de distorsion quadratique et dès 7kbps/source en termes de distorsion perceptive, alors même que ce n'est pas ce second critère qui est optimisé par la version actuelle.

Lorsqu'on considère que les scores obtenus par ces bornes théoriques des systèmes paramétriques sont déjà excellentes, on mesure tout l'intérêt qu'il y a à envisager CISS comme une méthode de choix pour aborder un problème de séparation informée.

Cinquième partie

Conclusion

Chapitre 17

Résumé des contributions

17.1 Processus gaussiens

La première partie de cet exposé est consacrée à la présentation du formalisme des processus gaussiens, que j'ai mis en œuvre tout au long de mon doctorat.

Une première contribution mineure de mon travail dans ce domaine est la présentation faite de ces modèles en partie I. S'il existe d'excellents livres ou articles introductifs sur le sujet en anglais, je n'ai pas trouvé d'exposé détaillé et en français des processus gaussiens dans un cadre aussi général que celui que j'ai considéré ici.

Mes contributions portant sur les processus gaussiens portent sur l'hypothèse de trames indépendantes et la généralisation de processus gaussiens localement stationnaires à une dimension quelconque du domaine de définition des sources.

Ensuite, s'ils reposent sur des théorèmes classiques du traitement du signal tels que le théorème de Bochner et de Wiener-Khinchin, il est peu courant de trouver énoncées les propriétés blanchissantes de la transformée de Fourier sur la représentation spectrale d'un processus gaussien en dimension quelconque.

Par ailleurs, le tramage et le modèle des Processus gaussiens Localement Stationnaires (PGLS), omniprésents dans les applications considérées en parties III et IV, ont fait l'objet d'une introduction originale comme des méthodes d'approximation pour la régression par processus gaussiens. De ce point de vue, je les ai mises en perspective par rapport aux autres approches proposées dans la littérature. L'extension naturelle du modèle PGLS en dimension quelconque a permis d'envisager des applications de séparation et de synthèse de signaux aléatoires définis sur des espaces à haute dimension.

J'ai introduit le modèle à factorisation non négative pour des PGLS définis en dimension quelconque. Si cette généralisation s'est faite naturellement, elle demeure originale. J'ai présenté l'algorithme d'optimisation correspondant, utilisable pour l'apprentissage efficace des hyperparamètres de PGLS. De plus, j'ai introduit le modèle par compression d'image, qui permet de compresser efficacement la DSP d'un signal audio.

Toutes ces contributions parcellaires portant sur le formalisme gaussien se trouvent soit dans ce texte, soit dans les publications issues de mon travail sur la séparation de processus gaussiens, que je vais aborder à présent.

17.2 Séparation de processus gaussiens

La principale contribution théorique de mon travail dans le domaine de la séparation de sources a été de considérer dans toute leur généralité les processus gaussiens comme un modèle de sources intéressant en vue de la séparation. Si ces modèles ont déjà fait l'objet de recherches très nombreuses dans la littérature bien avant le début de mon travail, ils étaient surtout envisagés dans le cas de PGLS définis sur $\mathbb{Z}$, c'est-à-dire de signaux audio.

Ainsi, j'ai montré tout au long de la partie II comment il est possible de séparer des réalisations de processus gaussiens définis sur des espaces quelconques dont on connaît les hyperparamètres. Ce

travail a fait l'objet d'une publication [131] aux IEEE Transactions on Signal Processing et d'une communication [132] à la conférence internationale IEEE Statistical Signal Processing (SSP 2011). Ces résultats permettent d'utiliser les processus gaussiens pour aborder des configurations inédites du problème de séparation de sources.

Si les modèles de mixages instantanés et convolutifs sont largement établis pour l'audio, j'en ai considéré l'extension naturelle au cas d'un domaine de définition de dimension quelconque pour les sources. La présentation originale du modèle diffus, récemment introduit par Duong *et al.* qui est faite en section 6.2 page 89 modélise l'image d'une source comme la résultante des mixages convolutifs de plusieurs réalisations indépendantes du même processus. Ce modèle a également été étendu en dimension quelconque. Sur ce sujet de la modélisation des procédures de mixage effectuées en studio, je suis par ailleurs co-auteur de l'article [201] de Nicolas Sturmel, présenté à la 132ème conférence de l'Audio Engineering Society (AES).

J'ai présenté dans toute leur généralité les processus gaussiens comme un modèle de source adéquat pour effectuer de la séparation.

Le problème de l'apprentissage des paramètres du modèle gaussien à partir des seuls mélanges en vue de la séparation a fait l'objet de nombreux travaux dans le domaine de la séparation de sources ces dernières années. Je me suis moi-même intéressé à plusieurs problèmes de ce type au cours de ma thèse. Ainsi, j'ai étudié la séparation de rythmiques d'enregistrements musicaux [131], la décomposition de mouvements de danse [133], et la séparation de la voix d'enregistrement musicaux [138, 78]. Pour des raisons de concision de mon exposé, je n'ai présenté dans ce texte que les deux premières de ces problématiques, au chapitre 8.

17.3 Séparation informée paramétrique

Dans le domaine de la séparation informée, présenté en partie III, j'ai proposé l'utilisation du modèle gaussien comme alternative au système d'inversion locale proposé par Parvaix *et al.*

S'il est apparu que SAOC utilise des techniques similaires au système que j'ai proposé, on a vu que l'approche originale que je présente repose sur l'utilisation de modèles de sources plus complexes et sur la prise en compte de procédures de mixage qui dépassent le seul modèle instantané.

Par ailleurs, la formalisation faite du problème de la séparation de sources informée et de ses différentes configurations que j'ai présentée au chapitre 9 englobe tous les cas de figure dont j'ai connaissance et qui ont été considérés auparavant dans la littérature. Il en inclue en plus quelques nouveaux, en particulier par la prise en compte de mixages diffus.

J'ai introduit l'utilisation de modèles de sources complexes pour la séparation paramétrique basée sur hypothèse gaussienne. La formalisation générale du problème est originale.

L'utilisation d'une première technique de séparation de sources informée mettant en œuvre le modèle NTF a fait l'objet d'une communication [130] à la conférence internationale Latent Variable Analysis and Independent Component Analysis (LVA/ICA 2010). J'y ai présenté les principales idées de la séparation informée paramétrique mettant en œuvre le modèle NTF.

L'utilisation du modèle CI dans un contexte de séparation informée a fait l'objet d'un dépôt de brevet [83] dont je suis un des co-auteurs. La présentation d'une première version du système paramétrique complet que je propose a fait l'objet d'une publication [130] à Signal Processing.

Enfin, j'ai organisé une vaste campagne d'évaluation des systèmes de séparation paramétrique existants en date de février 2012. Cette campagne d'évaluation a fait l'objet d'une communication [134] à la conférence internationale European Signal Processing Conference (EUSIPCO 2012)

17.4 Séparation informée par codage

Faire le lien entre séparation informée et codage de sources est une idée originale d'Alexey Ozerov sur laquelle nous avons beaucoup travaillé ensemble durant mon doctorat. Le système

CISS [1] qui résulte de cette collaboration est basé sur l'utilisation de techniques de codage de sources qui mettent en œuvre les distributions *a posteriori* fournies par la séparation. Il offre un cadre théorique puissant dans lequel la qualité des sources reconstruites peut être garantie et le débit total minimisé.

CISS est une idée d'Alexey Ozerov, que nous avons développée ensemble.

Notre travail sur CISS nous a conduit à présenter une première étude [160] sur le sujet à la conférence internationale IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA 2011). Ce travail préliminaire a utilisé un modèle de sources très basique et a surtout servi à valider l'intérêt de l'approche.

L'utilisation de modèles de sources plus poussés et de mixages complexes a fait l'objet de deux études plus approfondies.

La première [136] a porté sur l'utilisation de CISS dans le cas de mélanges multicanaux convolutifs avec le modèle de source NTF. Elle a été présentée à la conférence internationale European Signal Processing Conference (EUSIPCO 2012).

La deuxième est un article de revue [161] soumis aux IEEE Transactions on Audio Speech and Language Processing, qui porte sur une analyse théorique poussée de CISS et sur une évaluation approfondie de ses performances dans le cas d'un mixage monophonique instantané et du modèle NTF pour les sources.

17.5 Publications

Durant les trois années de mon doctorat et en comptant les articles dont je suis deuxième auteur, mes publications incluent trois articles de revue, dix articles de conférence et un brevet.

Par ailleurs, j'ai organisé avec Gaël Richard, Manuel Moussallam et Benoit Fuentes une session spéciale sur la séparation informée à la conférence internationale European Signal Processing Conference (EUSIPCO 2012), dont j'ai été membre du comité technique.

17.5.1 Articles de revue

- A. Liutkus, R. Badeau, and G. Richard, *Gaussian processes for underdetermined source separation*, IEEE Transactions on Signal Processing **59** (2011), no. 7, 3155 –3167
- A. Liutkus, J. Pinel, R. Badeau, L. Girin, and G. Richard, *Informed source separation through spectrogram coding and data embedding*, Signal Processing **92** (2012), no. 8, 1937 – 1949
- A. Ozerov, A. Liutkus, R. Badeau, and G. Richard, *Coding-based informed source separation: Nonnegative tensor factorization approach*, IEEE Trans. on Audio, Speech and Language Processing (2012)

17.5.2 Articles de conférences

- A. Liutkus and P. Leveau, *Separation of music+effects sound track from several international versions of the same movie*, Audio Engineering Society Convention 128, May 2010
- A. Liutkus, R. Badeau, and G. Richard, *Informed source separation using latent components*, 9th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA'10) (St Malo, France), 2010
- A. Liutkus, R. Badeau, and G. Richard, *Multi-dimensional signal separation with gaussian processes*, Proc. of IEEE Conf. on Statistical Signal Processing (SSP2011) (Nice, France), 2011
- A. Liutkus, Z. Rafii, R. Badeau, B. Pardo, and G. Richard, *Adaptive filtering for music/voice separation exploiting the repeating musical structure*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), mars 2012, pp. 1 –4

1. *Coding-Based Informed Source Separation.*

- N. Sturmel, A. Liutkus, J. Pinel, L. Girin, S. Marchand, G. Richard, R. Badeau, and L. Daudet, *Linear mixing models for active listening of music productions in realistic studio conditions*, 132th AES convention, Budapest, in press, 2012
- A. Liutkus, S. Gorlow, N. Sturmel, S. Zhang, L. Girin, R. Badeau, L. Daudet, S. Marchang, and G. Richard, *Informed source separation : a comparative study*, Proceedings European Signal Processing Conference (EUSIPCO 2012), August 2012
- A. Ozerov, A. Liutkus, R. Badeau, and G. Richard, *Informed source separation: source coding meets source separation*, IEEE Workshop Applications of Signal Processing to Audio and Acoustics (WASPAA) (New Paltz, New York, USA), October 2011
- B. Fuentes, A. Liutkus, R. Badeau, and G. Richard, *Probabilistic model for main melody extraction using constant-Q transform*, Proceedings IEEE Int. Conference on Acoustics, Speech and Signal Processing (ICASSP'2012), 2012
- A. Liutkus, A. Dremeau, D. Alexiadis, S. Essid, and P. Daras, *Analysis of dance movements using Gaussian processes*, ACM Multimedia Grand Challenge, September 2012
- A. Liutkus, A. Ozerov, R. Badeau, and G. Richard, *Spatial coding-based informed source separation*, Proceedings European Signal Processing Conference (EUSIPCO 2012), August 2012

17.5.3 Brevet

- L. Girin, A. Liutkus, G. Richard, and R. Badeau, *Procédé et dispositif de formation d'un signal mixé numérique audio, procédé et dispositif de séparation de signaux, et signal correspondant*, Demande de brevet no. B10/3035FR / GBO, October 2010, Institut Polytechnique de Grenoble et Institut Télécom, Télécom ParisTech

Chapitre 18

Perspectives

Mon travail a permis de fournir certaines approches alternatives pour aborder le problème délicat de la séparation de sources. Il s'inscrit cependant dans l'effort de recherche global d'une large communauté. Ce n'est en effet que grâce aux études menées par mes collègues actuels et passés dans le domaine que me sont venues les idées que j'ai exposées ici. J'espère qu'à son tour, mon travail pourra ouvrir la voie à de nouvelles études intéressantes.

Dans ce court chapitre, j'indique certaines pistes de recherche dans le prolongement naturel de mon travail.

18.1 Modèles non paramétriques de DSP

Dans le cas important où les signaux sources peuvent être modélisés comme des processus gaussiens localement stationnaires (PGLS), on a vu que le problème de la séparation de formes d'ondes se ramène souvent à une séparation de spectrogrammes. Il s'agit en effet de manière schématique d'expliquer les spectrogrammes des mélanges comme la somme des DSP des sources.

Dans tout ce travail, j'ai fait l'hypothèse qu'il est possible d'approximer les DSP des sources selon une forme paramétrique donnée. En particulier, le modèle linéaire NTF considère que ces DSP sont correctement assimilées au produit d'un nombre limité de facteurs. Si cette approche s'est avérée efficace dans de nombreux cas et en particulier pour la séparation informée, il reste clair que si on considère le spectrogramme d'une véritable source audio, il est en fait assez rare qu'on puisse réellement le comprendre comme la somme de plusieurs matrices de faible rang. Évidemment, cette hypothèse est valable en première approximation, mais si on cherche à procéder à une séparation, il est nécessaire d'introduire plus de souplesse dans le modèle.

Dans ce but, on a vu que certaines études se sont concentrées sur des modèles paramétriques plus complexes [53, 163]. Malgré tout, ces modèles restent limités dans le sens où il est toujours possible, voire probable, qu'en pratique les DSP sortent des carcans rigides que leur imposent ces modèles paramétriques.

Modéliser la DSP d'un PGLS comme la réalisation d'un processus aléatoire permettrait d'introduire une plus grande souplesse dans les modèles.

Or, c'est précisément le sens d'un modèle non paramétrique d'éviter de forcer les données à obéir à un modèle rigide fixé. C'est d'ailleurs ainsi que j'ai introduit les processus gaussiens en partie I comme un puissant modèle non-paramétrique de formes d'ondes.

En conséquence, je pense qu'une piste intéressante de recherche dans le formalisme que j'ai présenté pour la séparation est de considérer la DSP $P(\cdot, \cdot, j)$ d'une source comme la réalisation d'un *champ aléatoire* $\mathcal{P}(\cdot, \cdot, j \mid \theta)$, paramétré par un lot *d'hyperparamètres* θ. Cette formulation permettrait par exemple de considérer des modèles de DSP localement stationnaires, paramétrés par leurs *variogrammes* [32, 44].

Certaines études pionnières dans ce domaine ont déjà été proposées, qui modélisent la DSP des signaux sources comme la réalisation de champs Gamma [45] ou comme une somme infinie de bases spectrales [149].

18.2 Abandon de l'hypothèse locale

L'hypothèse locale, que j'ai présentée en section 3.2.4, suppose que les différentes trames d'un signal sont indépendantes. Or, il est manifeste que cela est faux en toute rigueur, puisque les trames présentent entre elles un certain recouvrement. De plus, dans le cas où le signal contient des composantes sinusoïdales stables, une forte dépendance entre les trames successives peut apparaître.

Récemment, certaines études se sont intéressées à ce point dans le but d'améliorer les performances des systèmes de séparation [127, 11]. Dans ce domaine, la prise en compte des dépendances entre trames successives permet en effet de réduire le champ des possibles et donc donne toutes les chances à une séparation d'obtenir des meilleurs résultats.

L'hypothèse locale suppose une indépendance entre les trames. Elle est fausse à cause du recouvrement des trames adjacentes et des composantes sinusoïdales qui sont stables d'une trame sur l'autre.

Par ailleurs, il me semble clair que dans une perspective de codage, l'hypothèse locale est sous-optimale. Elle revient en effet à encoder de nouveau pour chaque point temps-fréquence une information qui est déjà présente en partie dans les trames adjacentes. Je pense qu'une direction de recherche intéressante est donc la prise en compte de ces dépendances dans un système de codage de source, qu'il soit *a priori* ou *a posteriori* comme CISS.

En pratique, l'abandon de l'hypothèse locale se traduit par la nécessite de modéliser les dépendances entre les valeurs adjacentes de la TFCT des signaux. Le principal défi à relever dans ce contexte est la complexité encore élevée des algorithmes correspondants.

18.3 Mélanges modifiés ou compressés

Dans l'ensemble de ma discussion sur la séparation informée aux parties III et IV, j'ai toujours considéré que c'est exactement le même mélange qui est disponible au codeur et au décodeur. En particulier, j'ai toujours supposé que le mélange observé au décodeur est la somme des images des sources :

$$\forall\,(t,i)\,,\ \tilde{\boldsymbol{x}}\,(t,i) = \sum_{j=1}^{J} \tilde{\boldsymbol{y}}\,(t,i,j)\,. \tag{18.3.1}$$

Cette hypothèse permet de fidèlement rendre compte de la plupart des cas de figures où le mélange $\boldsymbol{x}$ est obtenu comme un mixage des sources. En effet, ce n'est en général pas à l'équation 18.3.1 qu'il faut attribuer d'éventuelles faiblesses, mais plutôt à la manière dont est modélisé le lien entre les processus sources $\tilde{s}\,(\cdot,j)$ et leurs images $\tilde{\boldsymbol{y}}\,(\cdot,\cdot,j)$. Si j'ai présenté plusieurs manières de formaliser un tel lien aux chapitres 5 et 6, il est entendu que ces modèles ne sont que des *approximations* de la complexe réalité des procédures de mixage telles qu'effectuées en studio. De nouvelles études viendront sûrement proposer à l'avenir de nouveaux moyens de rendre compte de ce lien, sans fondamentalement remettre en cause l'hypothèse qu'un mélange est la somme des images des sources.

La situation devient différente si ce n'est plus le mélange original $\tilde{\boldsymbol{x}}$ qu'on observe, mais plutôt une version *modifiée*. Dans ce cas en effet, il n'est plus garanti que l'expression 18.3.1 reste valide. Par exemple, on peut considérer une modification classique où le mélange est manipulé après mixage par une opération dite de *post-production,* qui consiste à le faire passer par une chaine de traitement plus ou moins complexe et qui peut faire intervenir des effets fortement non linéaires. Comme Nicolas Stürmel l'a montré dans notre article commun [201], il est souvent possible de répercuter ces post-traitements sur les images des sources, de manière à se ramener à 18.3.1. Dans ce cas, il est nécessaire de pouvoir prendre en compte ces répercussions dans le modèle de mixage.

Un autre cas de figure particulièrement courant de modification du mélange après mixage est celui de sa *compression* par un codeur audio tel que MPEG-1-layerIII (MP3 [146]) ou MPEG-2-LayerIII (AAC [147]). En effet, j'ai supposé partout qu'au décodeur, c'est le mélange original qui est observé. Or, il serait intéressant de pouvoir considérer qu'au lieu du mélange original, c'est une version compressée dont dispose le décodeur. Cette configuration est en effet importante dans les applications compte tenu de l'importance grandissante que prennent les formats numériques dans la diffusion de la musique.

C'est l'objet de travaux en cours que de tâcher de prendre en compte la dégradation infligée aux mélanges dans la procédure de codage informé. Plusieurs pistes de recherche me semblent prometteuses dans ce but.

Tout d'abord, le codage avec perte des mélanges pourrait être intégré au modèle comme l'ajout d'une opération non linéaire appliquée à chaque élément TF de $x\left(f,n,\cdot\right)$ après mixage. Ce faisant, le modèle deviendrait équivalent à un modèle non-linéaire *a posteriori* (Post Non-Linear mixing [202, 2, 38]).

Par ailleurs, on peut envisager d'adapter CISS pour le cas de mélanges compressés. En cas d'une forte dégradation du mélange en un point TF, le modèle probabiliste utilisé pour encoder les sources pourrait être le modèle *a priori* $p\left(\boldsymbol{s}\left(f,n,\cdot\right)\mid\Theta\right)$. Dans le cas contraire, on pourrait utiliser la distribution *a posteriori* $p\left(\boldsymbol{s}\left(f,n,\cdot\right)\mid\boldsymbol{x}\left(f,n,\cdot\right),\Theta\right)$. J'envisage de procéder à la décision entre les deux sur la base de l'amplitude du mélange en chaque point TF. De cette manière, on évite la transmission d'une variable indicatrice supplémentaire.

18.4 Apprentissage discriminant du modèle

Un défaut des systèmes proposés pour la séparation de sources informée que j'ai déjà souligné en section 10.1 page 129 est que l'apprentissage du modèle au codeur ne se fait pas en optimisant la vraisemblance *a posteriori* des sources dans la distribution estimée, mais plutôt la distribution jointe des sources et des mélanges, ce qui correspond à une approche d'apprentissage générative, par opposition à une approche discriminante.

Plutôt que de modéliser conceptuellement les mélanges comme produits à partir des sources, l'approche discriminante modéliserait les sources comme produites à partir des mélanges. Si ce modèle ne correspond pas au processus génératif conduisant aux données observées, il correspond au problème qu'on veut véritablement résoudre.

De manière à pouvoir mettre en œuvre un apprentissage discriminant, je pense qu'il faudrait établir une nouvelle paramétrisation de la distribution *a posteriori* $p\left(\boldsymbol{s}\mid\boldsymbol{x},\Theta\right)$ des sources étant donnés les mélanges qui s'affranchisse de la compréhension des mélanges comme produits à partir des sources. Au lieu de cela, il faudrait plutôt modéliser les sources comme produites à partir des mélanges, de manière à pouvoir optimiser plus facilement les paramètres.

Un tel modèle probabiliste discriminant pourrait être utilisé en séparation informée paramétrique ou par CISS. Il offrirait l'avantage fort de permettre un apprentissage optimal des paramètres de séparation, à défaut de correspondre à un modèle de la production réelle des données.

18.5 CISS perceptif

Pour finir, il est certain que le système CISS de séparation informée par codage que j'ai présenté en partie IV présente des propriétés très attractives qui méritent de plus amples développements. En l'état, deux principaux points me semblent importants à aborder à court terme concernant CISS.

Même si elle sont déjà très bonnes, les performances de CISS pourraient être améliorées par l'utilisation d'une distorsion perceptive.

Tout d'abord, il est nécessaire d'en réaliser une implémentation concrète, de manière à vérifier que les performances théoriques présentées en partie IV sont atteintes en pratique. Il est établi dans la littérature [76] que c'est bien le cas, mais je voudrais m'en assurer.

Ensuite, la fonction de distorsion que j'ai considérée est l'erreur quadratique. Il est bien établi que dans le but d'effectuer un codage de signaux audio, il vaut mieux considérer d'autres distorsions pondérées perceptivement [205, 174, 129]. Fort heureusement, de telles extensions de CISS ne posent aucun problème conceptuel, puisque les algorithmes de codage correspondants existent déjà dans la littérature et qu'il suffit de les appliquer au cas d'un codage *a posteriori*.

Cependant, je suis curieux de mesurer l'impact de la prise en compte d'une distorsion perceptive dans les scores PSM obtenus par CISS.

18.6 Séparation informée par des reprises

Une piste de recherche très intéressante déjà abordée dans [81] et que je n'ai pas détaillée dans ce texte est la possibilité d'informer la séparation par des imitations des sources à séparer. Dans un contexte musical, s'il est souvent difficile d'obtenir les véritables pistes séparées des morceaux, il est en revanche plus aisé d'en obtenir une imitation, réalisée par des musiciens.

Pour avoir déjà abordé informellement cette question, il parait clair que son enjeu est d'arriver à profiter de cette connaissance sans pour autant supposer une structure ou un timbre de l'original exactement identiques à ceux de la reprise. Cette problématique rejoint celle que j'ai considérée dans cet exposé d'une séparation informée par des connaissances annexes, mais la dépasse dans le sens où les signaux observés ne sont pas ceux utilisés pour former les mélanges.

Le problème de la séparation informée par des reprises consiste à chercher à séparer d'un morceau des sources dont on dispose d'une imitation.

Je conjecture qu'une approche intéressante à ce problème serait d'introduire un nouveau type de processus de mixage qui permet de considérer l'image d'une source comme produite à la fois par un filtrage, mais aussi et surtout par un découpage non séquentiel.

Annexe A

Synthèse de processus Gaussiens

A.1 Générer un vecteur gaussien

Dans cette section, je vais présenter rapidement la procédure générale utilisée dans tout ce texte pour la synthèse de processus gaussiens. Comme on l'a vu plus haut, ces modèles reposent sur l'hypothèse qu'un nombre fini quelconque d'échantillons du processus a une distribution jointe gaussienne multivariée. Que l'on considère une distribution *a priori* 2.2.12 page 26 ou une distribution *a posteriori* 2.2.8 page 25 d'un vecteur d'échantillons $\tilde{\boldsymbol{s}}$ de dimension $L \times 1$, on aboutit dans tous les cas à une expression du type :

$$\tilde{\boldsymbol{s}} \sim \mathcal{N}(\boldsymbol{\mu}, K), \tag{A.1.1}$$

où $\boldsymbol{\mu}$ est un vecteur moyenne, de dimension $L \times 1$, tandis que K est une matrice de covariance, de dimension $L \times L$. Dans ces conditions, si on cherche à générer des réalisations de processus gaussiens, il faut être capable de générer des vecteurs gaussiens, puisque tout ensemble fini d'échantillons d'un processus gaussien est un tel vecteur.

Dans ce but, il suffit d'effectuer une décomposition de Cholesky de la matrice K, c'est-à-dire de trouver une matrice M triangulaire inférieure dont les éléments diagonaux sont des réels positifs, telle que :

$$K = MM^H. \tag{A.1.2}$$

Une telle décomposition est en effet nécessairement possible, puisque la définition même des processus gaussiens assure que la matrice K est définie positive. Dans ces conditions, si on note comme en Python ou Matlab

$$\boldsymbol{r} = \text{randn}(L, 1)$$

un vecteur de dimension $L \times 1$ dont toutes les entrées sont indépendantes et distribuées selon une loi gaussienne centrée $\mathcal{N}(0, 1)$, on peut constater que

$$\widetilde{\boldsymbol{s}} = \boldsymbol{\mu} + M\boldsymbol{r} \tag{A.1.3}$$

est bien distribué selon A.1.1. En effet :

- $\mathbb{E}[\widetilde{\boldsymbol{s}}] = \boldsymbol{\mu}$, puisque tous les éléments de $\boldsymbol{r}$ sont de moyenne nulle.
- $\mathbb{E}\left[\widetilde{\boldsymbol{s}}\widetilde{\boldsymbol{s}}^H\right] - \boldsymbol{\mu}\boldsymbol{\mu}^H = M\mathbb{E}\left[\boldsymbol{r}\boldsymbol{r}^H\right]M^H = K$, puisque tous les éléments de $\boldsymbol{r}$ sont indépendants et distribués selon une loi gaussienne de variance 1.

En pratique, il arrive que des problèmes numériques empêchent la décomposition de Cholesky de K. Cela peut arriver lorsque la fonction de covariance produit une matrice de covariance singulière, ce qui est possible parce qu'une fonction de covariance n'a pas à garantir la stricte positivité de K, mais seulement son caractère semi-défini positif. Dans ces conditions, il est possible que l'outil de calcul formel utilisé refuse d'effectuer la décomposition A.1.2. Une manière simple de contourner ce problème est de plutôt considérer une décomposition en valeurs singulières de K (Singular Value Decomposition, SVD, en anglais) :

$$K = U\Lambda V, \tag{A.1.4}$$

où $\Lambda = \text{diag}([\lambda_1, \ldots, \lambda_L])$ est une matrice diagonale. Une telle décomposition ne pose de problème dans aucun cas. Dans la mesure où K est définie positive, on a $U \approx V^H$, et les résultats ci-dessus restent valables pour

$$\widetilde{\boldsymbol{s}} = \boldsymbol{\mu} + U\sqrt{\Lambda}\boldsymbol{r}.$$

C'est en utilisant cette procédure que j'ai généré la plupart des figures de cette partie. Une fois de plus, il est remarquable que cette technique de synthèse ne fasse aucune hypothèse sur le domaine de définition $\mathbb{T}$ du processus considéré. Seuls sont nécessaires le vecteur $\boldsymbol{\mu}$ et la matrice K. Bien entendu, il reste clair que la décomposition de Cholesky A.1.2 ou la SVD A.1.4 sont des opérations dont la complexité en $\mathcal{O}\left(L^3\right)$ peut devenir prohibitive si L est très grand. Dans le cas d'un processus défini sur $\mathbb{T} = \mathbb{R}^2$ comme ceux présentés en figure 2.5 page 32, on peut facilement aboutir à $L = 200 \times 200 = 40000$, ce qui rend la technique présentée ici inutilisable. Une première alternative est de considérer la synthèse d'un tramage du signal recherché suivie d'une addition-recouvrement, comme définis en section 3.2 page 47. Une deuxième est d'exploiter les propriétés des signaux stationnaires, si la fonction de covariance considérée est stationnaire [1].

A.2 Le cas stationnaire

Dans le cas où la fonction de covariance considérée est stationnaire et où on souhaite synthétiser un processus gaussien centré sur une grille de points régulièrement échantillonnés, les résultats de la section 2.5.3 peuvent être mis à profit pour réduire la complexité de la synthèse de $\mathcal{O}\left(L^3\right)$ à $\mathcal{O}\left(L \log L\right)$, dominée par le calcul de transformées de Fourier discrètes. Supposons ainsi que $\mathbb{T} = \mathbb{Z}^D$ et qu'on se donne une grille T de points dans $\mathbb{T}$ comme en 2.5.1 page 35. La fonction de covariance $k\left(t - t'\right)$, stationnaire, est supposée connue ainsi que sa transformée de Fourier discrète $\mathcal{F}_D\left\{k\right\}$. En section 2.5.3 page 38, on a vu que les coefficients de la transformée de Fourier discrète $\boldsymbol{s}$ de $\tilde{\boldsymbol{s}}$ sur T sont indépendants et qu'on a :

$$\forall f,\ \boldsymbol{s}\left(f\right) \sim \mathcal{N}_c\left(0, \mathcal{F}_D\left\{k\right\}\left(f\right)\right).$$

Ainsi, pour générer une réalisation de $\tilde{\boldsymbol{s}}$, il suffit de générer un bruit blanc gaussien $\boldsymbol{r}$ sur T, en calculer la transformée de Fourier discrète, et la multiplier point par point par $\sqrt{\mathcal{F}_D\left\{k\right\}}$. La forme d'onde correspondante s'obtient par transformée de Fourier inverse :

$$\widetilde{\boldsymbol{s}} = \mathcal{F}_D^{-1}\left\{\mathcal{F}_D\left\{k\right\} \cdot \mathcal{F}_D\left(\boldsymbol{r}\right)\right\}.$$

En figure A.1, j'ai représenté un exemple du procédé pour un processus défini sur $\mathbb{T} = \mathbb{Z}^2$, dont la fonction de covariance est l'exponentielle carrée 2.3.2 page 31. Bien qu'on ait $L = 250000$, la synthèse de ces réalisations est effectuée en une fraction de seconde.

On a déjà vu plus haut en section 2.3 page 28 de nombreux autres exemples de réalisations de processus gaussiens.

1. Une troisième possibilité, que je ne présenterai pas ici pour une raison de concision de l'exposé, est de considérer la synthèse séquentielle du signal, pour laquelle la distribution de chaque portion est conditionnée aux échantillons déjà générés, en utilisant les résultats 2.2.8 page 25.

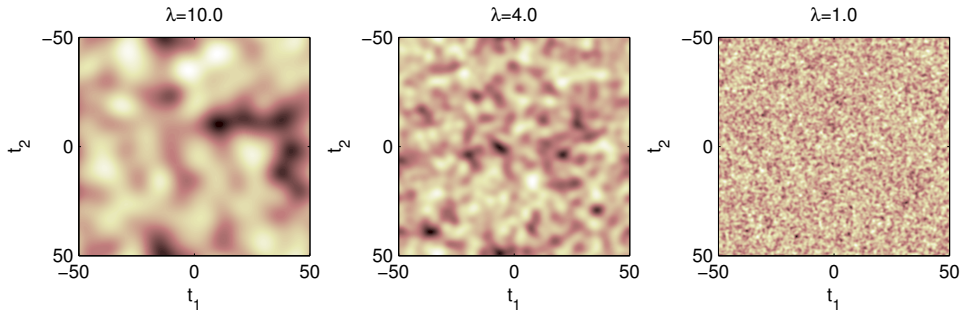

FIGURE A.1: Illustration d'une synthèse de processus gaussien de fonction de covariance EC. Le nombre de points générés est $L = 500 \times 500 = 250000$.

Bibliographie

[1] P. Abrahamsen, *A review of Gaussian random fields and correlation functions*, Tech. Report 878, Norsk Regnesentral, Oslo, Norway, April 1997.

[2] S. Achard and C. Jutten, *Identifiability of post-nonlinear mixtures*, Signal Processing Letters, IEEE **12** (2005), no. 5, 423 – 426.

[3] R.J. Adler and J.E. Taylor, *Random fields and geometry*, Springer Monographs in Mathematics, 2007.

[4] V.R. Algazi, R.O. Duda, D.M. Thompson, and C. Avendano, *The CIPIC HRTF Database*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (New Paltz, New York, USA), October 2001, pp. 99–102.

[5] J. Allen, *Short term spectral analysis, synthesis, and modification by discrete Fourier transform*, Acoustics, Speech and Signal Processing, IEEE Transactions on **25** (1977), no. 3, 235 – 238.

[6] J.B. Allen and L.R. Rabiner, *A unified approach to short-time Fourier analysis and synthesis*, Proceedings of the IEEE **65** (1977), no. 11, 1558 – 1564.

[7] S. Arberet, A. Ozerov, N.Q.K. Duong, E. Vincent, R. Gribonval, F. Bimbot, and P. Vandergheynst, *Nonnegative matrix factorization and spatial covariance model for under-determined reverberant audio source separation*, Information Sciences Signal Processing and their Applications (ISSPA), 2010 10th International Conference on, IEEE, 2010, pp. 1–4.

[8] N. Aronszajn, *La théorie générale des noyaux reproduisants et ses applications, première partie*, Proceedings of the Cambridge Philosophical Society **39** (1944), 133–153.

[9] N. Aronszajn, *Theory of reproducing kernels*, Transactions of the American Mathematical Society **68** (1950), no. 3, 337–404.

[10] C. Avendano, *Frequency-domain source identification and manipulation in stereo mixes for enhancement, suppression and re-panning applications*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), October 2003, pp. 55 – 58.

[11] R. Badeau, *Gaussian modeling of mixtures of non-stationary signals in the time-frequency domain (HR-NMF)*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (New Paltz, NY, USA), October 2011, pp. 253–256.

[12] R. Badeau, N. Bertin, and E. Vincent, *Stability analysis of multiplicative update algorithms for nonnegative matrix factorization*, Proc. IEEE Intl. Conf. Acoust. Speech Signal Processing (ICASSP'11) (Washington, DC, USA), April 2009, pp. 105–108.

[13] ———, *Stability analysis of multiplicative update algorithms and application to non-negative matrix factorization*, IEEE Transactions on Neural Networks **21** (2010), no. 12, 1869–1881.

[14] L. Barrington, A.B. Chan, and G. Lanckriet, *Modeling music as a dynamic texture*, IEEE Transactions on Audio, Speech and Language Processing **18** (2010), 602–612.

[15] S. Beack, T. Lee, M. Kim, and K. Kang, *An efficient time-frequency representation for parametric-based audio object coding*, ETRI Journal **33 (6)** (2011), 945–948.

[16] A. Belouchrani, K. A. Meraim, J. F. Cardoso, and E. Moulines, *A blind source separation technique using second order statistics*, IEEE Transactions on Signal Processing **45** (1997), 434–444.

[17] L. Benaroya, F. Bimbot, and R. Gribonval, *Audio source separation with a single sensor*, IEEE Trans. on Audio, Speech and Language Processing **14** (2006), no. 1, 191–199.

[18] L. Benaroya, L. McDonagh, F. Bimbot, and R. Gribonval, *Non negative sparse representation for Wiener based source separation with a single sensor*, Proc. IEEE Intl. Conf. Acoust. Speech Signal Processing (ICASSP'03) (Hong-Kong), April 2003, pp. 613–616.

[19] N. Bertin, *Les factorisations en matrices non-negatives. approches contraintes et probabilistes, application a la transcription automatique de musique polyphonique.*, Ph.D. thesis, Telecom Paristech, 2009.

[20] A. Bertinet and Thomas C. Agnan, *Reproducing Kernel Hilbert Spaces in Probability and Statistics*, Kluwer Academic Publishers, 2004.

[21] M. Bierlaire, *Introduction à l'optimisation différentiable*, Enseignement des mathématiques, Presses polytechniques et universitaires romandes, 2006.

[22] P. Bofill and M. Zibulevsky, *Underdetermined blind source separation using sparse representations*, IEEE Transactions on Signal Processing **81** (2001), no. 11, 2353–2362.

[23] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, March 2004.

[24] J. Breebaart, J. Herre, C. Faller, J. Röden, F. Myburg, S. Disch, H. Purnhagen, G. Hotho, M. Neusinger, K. Kjörling, and W. Oomen, *MPEG Spatial Audio Coding / MPEG Surround : Overview and current status*, AES 119th convention (New York, USA), October 2005.

[25] J. Breebaart, S. van de Par, and A. Kohlrausch, *High-quality parametric spatial audio coding at low bit rates*, AES 116th convention (Berlin, Germany), May 2004.

[26] A.S. Bregman, *Auditory scene analysis, the perceptual organization of sound*, MIT Press, 1994.

[27] J. Capon, *High-resolution frequency-wavenumber spectrum analysis*, Proceedings of the IEEE **57** (1969), no. 8, 1408 – 1418.

[28] J.-F. Cardoso, *Blind signal separation : statistical principles*, Proceedings of the IEEE **90** (1998), 2009–2026.

[29] A. T. Cemgil, S. J. Godsill, P. H. Peeling, and N. Whiteley, *The Oxford Handbook of Applied Bayesian Analysis*, no. ISBN13 : 978-0-19-954890-3, ch. Bayesian Statistical Methods for Audio and Music Processing, Oxford University Press, 2010.

[30] A.T. Cemgil, P. Peeling, O. Dikmen, and S. Godsill, *Prior structures for Time-Frequency energy distributions*, Proc. of the 2007 IEEE Workshop on. App. of Signal Proc. to Audio and Acoust. (WASPAA) (NY, USA), October 2007, pp. 151–154.

[31] B. Cheng, C. Ritz, and I. Burnett, *Encoding independent sources in spatially squeezed surround audio coding*, 8th Pacific Rim Conference on Multimedia (PCM'07) (Hong Kong, China), December 2007, pp. 804–813.

[32] J.-P. Chilès and P. Delfiner, *Geostatistics : Modeling Spatial Uncertainty*, 2 ed., Wiley-Interscience, April 2012.

[33] W.C. Chu, *Speech coding algorithms : foundation and evolution of standardized coders*, J. Wiley, 2003.

[34] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative matrix and tensor factorizations : Applications to exploratory multi-way data analysis and blind source separation*, Wiley Publishing, September 2009.

[35] A. Cichocki, Liqing Zhang, and T. Rutkowski, *Blind separation and filtering using state space models*, Proc. IEEE Int. Symp. Circuits and Systems ISCAS '99, vol. 5, 1999, pp. 78–81.

[36] W.S. Cleveland and S.J. Devlin, *Locally weighted regression : An approach to regression analysis by local fitting*, Journal of the American Statistical Association **83** (1988), 596–610.

[37] P. Comon, *Independent component analysis, A new concept ?*, Signal Processing **36** (1994), no. 3, 287–314.

[38] P. Comon and C. Jutten (eds.), *Handbook of blind source separation : Independent component analysis and blind deconvolution*, Academic Press, 2010.

[39] ———, *Séparation de sources. Principes et algorithmes*, 2011, Lecture notes.

[40] R. Crochiere, *A weighted overlap-add method of short-time Fourier analysis/synthesis*, Acoustics, Speech and Signal Processing, IEEE Transactions on **28** (1980), no. 1, 99 – 102.

[41] G. Darmois, *Analyse générale des liaisons stochastiques*, Rev. Inst. Internat. Stat. **21** (1953), 2–8.

[42] A.P. Dempster, N.M. Laird, and B.D. Rubin, *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society **39** (1977), 1–38.

[43] L. Deng, H. Leung, N. Gu, and Y. Yang, *Recognizing dance motions with segmental svd*, International Conference on Pattern Recognition, 2010, pp. 1537–1540.

[44] P.J. Diggle and P. J. Ribeiro, *Model-based Geostatistics*, 1 ed., Springer series in statistics, Springer, March 2007.

[45] O. Dikmen and A. T. Cemgil, *Gamma Markov random fields for audio source modelling*, IEEE Transactions on Audio, Speech, and Language Processing **18** (2010), no. 3, 589–601.

[46] O. Dikmen and A.T. Cemgil, *Unsupervised single-channel source separation using Bayesian NMF*, Proc. of the 2009 IEEE Workshop on. App. of Signal Proc. to Audio and Acoust. (WASPAA) (NY, USA), October 2009, pp. 93–96.

[47] N.Q.K Duong, *Modélisation gaussienne de rang plein des mélanges audio convolutifs appliquée à la séparation de sources*, Ph.D. thesis, Université Rennes 1, November 2011.

[48] N.Q.K. Duong, H. Tachibana, E. Vincent, N. Ono, R. Gribonval, and S. Sagayama, *Multichannel harmonic and percussive component separation by joint modeling of spatial and spectral continuity*, Acoustics, Speech and Signal Processing, IEEE Conference on (ICASSP'11) (Prague, Czech Republic), January 2011.

[49] N.Q.K Duong, E. Vincent, and R. Gribonval, *Under-determined convolutive blind source separation using spatial covariance models*, Acoustics, Speech and Signal Processing, IEEE Conference on (ICASSP'10) (Dallas, United States), March 2010, pp. 9–12.

[50] N.Q.K. Duong, E. Vincent, and R. Gribonval, *Under-determined reverberant audio source separation using a full-rank spatial covariance model*, Audio, Speech, and Language Processing, IEEE Transactions on **18** (2010), no. 7, 1830 –1840.

[51] N.Q.K. Duong, E. Vincent, and R. Gribonval, *Under-Determined Reverberant Audio Source Separation Using Local Observed Covariance and Auditory-Motivated Time-Frequency Representation*, Latent Variable Analysis and Signal Separation, 9th International Conference on (Saint-Malo, France), vol. 6365, September 2010, pp. 73–80.

[52] ———, *An acoustically-motivated spatial prior for under-determined reverberant source separation*, Acoustics, Speech and Signal Processing, IEEE Conference on (ICASSP'11) (Prague, Czech Republic), February 2011.

[53] J.-L. Durrieu, B. David, and G. Richard, *A musically motivated mid-level representation for pitch estimation and musical audio source separation*, Selected Topics in Signal Processing, IEEE Journal of **5** (2011), no. 6, 1180 –1191.

[54] J.-L. Durrieu, A. Ozerov., C. Févotte, G. Richard, and B. David, *Main instrument separation from stereophonic audio signals using a source/filter model*, Proc. 17th European Signal Proc. Conf. (EUSIPCO'09) (Glasgow, UK), August 2009, pp. 15–19.

[55] J.-L. Durrieu, G. Richard, and B. David, *Singer melody extraction in polyphonic signals using source separation methods*, Proc. IEEE Intl. Conf. Acoust. Speech Signal Processing (ICASSP'08) (Las Vegas, USA), March 2008.

[56] J.-L. Durrieu, G. Richard, and B. David, *An iterative approach to monaural musical mixture de-soloing*, Proc. IEEE Intl. Conf. Acoust. Speech Signal Processing (ICASSP'09) (Washington, DC, USA), April 2009, pp. 105–108.

[57] J.L. Durrieu and J.P. Thiran, *Musical audio source separation based on user-selected F0 track*, Proc. of International Conference on Latent Variable Analysis and Signal Separation (Tel-Aviv, Israel), March 12-15 2012.

[58] C. Duxbury, J.P. Bello, M. Davies, and M. Sandler, *Complex domain onset detection for musical signals*, In Proc. Digital Audio Effects Workshop (DAFx (London, UK), September 2003.

[59] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, *Subjective and objective quality assessment of audio source separation*, IEEE Transactions on Audio, Speech, and Language Processing (2011).

[60] J. Engdegård, C. Falch, O. Hellmuth, J. Herre, J. Hilpert, A. Hölzer, J. Koppens, H. Mundt, H.O. Oh, H. Purnhagen, B. Resch, L. Terentiev, M.L. Valero, and L. Villemoes, *MPEG spatial audio object coding - the ISO/MPEG standard for efficient coding of interactive audio scenes*, Audio Engineering Society Convention 129, 11 2010.

[61] J. Engdegård, B. Resch, C. Falch, O. Hellmuth, J. Hilpert, A. Hölzer, L. Terentiev, J. Breebaart, J. Koppens, E. Schuijers, and W. Oomen, *Spatial audio object coding (SAOC) - The upcoming MPEG standard on parametric object based audio coding*, 124th Audio Engineering Society Convention (AES 2008) (Amsterdam, Netherlands), May 2008.

[62] S. Essid, Xinyu Lin, M. Gowing, G. Kordelas, A. Aksay, P. Kelly, T. Fillon, Q. Zhang, A. Dielmann, V. Kitanovski, R. Tournemenne, A. Masurelle, E. Izquierdo, N.E. O'Connor, P. Daras, and G. Richard, *A multi-modal dance corpus for reseach into interaction between humans in virtual environments*, Accepted for publication in Journal on Multimodal User Interfaces, Special Issue on Multimodal Corpora, Springer, 2012.

[63] S. Ewert and M. Müller, *Score-informed voice separation for piano recordings*, Proceedings of the International Society for Music Information Retrieval Conference(ISMIR) (Miami, USA), 2011, pp. 245–250.

[64] ———, *Score informed source separation*, Multimodal Music Processing (Masataka Goto Meinard Müller and Markus Schedl, eds.), Dagstuhl Follow-Ups, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2012.

[65] ———, *Using score-informed constraints for NMF-based source separation*, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) (Kyoto, Japan), March 2012.

[66] C. Falch, L. Terentiev, and J. Herre, *Spatial audio object coding with enhanced audio object separation*, 13th International Conference on Digital Audio Effects (DAFx-10) (Graz, Austria), September 2010.

[67] G. Fant, J. Liljencrants, and Q. Lin, *A four-parameter model of glottal flow*, STL-QPSR **4** (1985), no. 1985, 1–13.

[68] M. Feder and E. Weinstein, *Parameter estimation of superimposed signals using the EM algorithm*, IEEE Transactions on Acoustics **36** (1988), 477–489.

[69] C. Févotte, N. Bertin, and J.-L. Durrieu, *Nonnegative matrix factorization with the Itakura-Saito divergence. With application to music analysis*, Neural Computation **21** (2009), no. 3, 793–830.

[70] C. Févotte and J.-F. Cardoso, *Maximum likelihood approach for blind audio source separation using time-frequency gaussian models*, Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (Mohonk, NY, USA), Oct. 2005, pp. 78–81.

[71] C. Févotte and J. Idier, *Algorithms for nonnegative matrix factorization with the beta-divergence*, Neural Computation **23** (2011), no. 9, 2421–2456.

[72] C. Févotte and A. Ozerov, *Notes on nonnegative tensor factorization of the spectrogram for audio source separation : statistical insights and towards self-clustering of the spatial cues*, 7th International Symposium on Computer Music Modeling and Retrieval (CMMR 2010), 2010.

[73] D. Fitzgerald, *Harmonic/percussive separation using median filtering*, Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10) (Graz, Austria), September 2010.

[74] D. Fitzgerald and M. Cranitch, *Sound source separation using shifted non-negative tensor factorisation*, Proceedings on the IEE Conference on Audio and Speech Signal Processing (ICASSP), 2006.

[75] D. FitzGerald, M. Cranitch, and E. Coyle, *On the use of the beta divergence for musical source separation*, Proc. of Irish Sig. and Systems Conf. (ISSC'08), 2008.

[76] FlexCode, *Deliverable D-1.1 : Baseline Source Coder*, Tech. report, European Union, Oct. 2008.

[77] B Fuentes, R Badeau, and G Richard, *Blind Harmonic Adaptive Decomposition Applied to Supervised Source Separation*, Proceedings European Signal Processing Conference (EUSIPCO 2012) (Bucharest, Romania), 2012.

[78] B. Fuentes, A. Liutkus, R. Badeau, and G. Richard, *Probabilistic model for main melody extraction using constant-Q transform*, Proceedings IEEE Int. Conference on Acoustics, Speech and Signal Processing (ICASSP'2012), 2012.

[79] H. Fujisaki and M. Ljungqvist, *Proposal and evaluation of models for the glottal source waveform*, Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86., vol. 11, apr 1986, pp. 1605 – 1608.

[80] J. Ganseman, G. J. Mysore, J.S. Abel, and P. Scheunders, *Source separation by score synthesis*, International Computer Music Conference (ICMC 2010) (New York, USA), June 2010.

[81] T. Gerber, M. Dutasta, L. Girin, and C. Févotte, *Professionally-produced music separation guided by covers*, Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR), 2012, p. to appear.

[82] O. Gillet and G. Richard, *Transcription and separation of drum signals from polyphonic music*, IEEE Trans. on Audio, Speech, and Language Processing, **16** (2008), no. 3, 529 –540.

[83] L. Girin, A. Liutkus, G. Richard, and R. Badeau, *Procédé et dispositif de formation d'un signal mixé numérique audio, procédé et dispositif de séparation de signaux, et signal correspondant*, Demande de brevet no. B10/3035FR / GBO, October 2010, Institut Polytechnique de Grenoble et Institut Télécom, Télécom ParisTech.

[84] P. Smaragdis G.J. Mysore, *Relative pitch estimation of multiple instruments*, Proc. IEEE Intl. Conf. Acoust. Speech Signal Processing (ICASSP'09), 2009, pp. 313–316.

[85] T. Gneiting, *Compactly supported correlation functions*, Tech. report, NRCSE, 2000, NRCSE-TRS No. 045.

[86] Z. Goh, K.-C. Tan, and T.G. Tan, *Postprocessing method for suppressing musical noise generated by spectral subtraction*, IEEE Transactions on Speech and Audio Processing **6** (1998), no. 3, 287 –292.

[87] S. Gorlow and S. Marchand, *Informed source separation : Underdetermined source signal recovery from an instantaneous stereo mixture*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), October 2011, pp. 309 –312.

[88] S. Gorlow and S. Marchand, *Informed audio source separation using linearly constrained spatial filters*, IEEE Transactions on Audio, Speech and Language Processing **20(9)** (2012).

[89] R. M. Gray, *Source coding theory*, Kluwer Academic Press, 1990.

[90] Robert M. Gray, *Toeplitz and circulant matrices : A review*, Tech. report, Deptartment of Electrical Engineering, Stanford University, 2001.

[91] R. Gribonval, *Sparse decomposition of stereo signals with matching pursuit and application to blind separation of more than two sources from a stereo mixture*, Acoustics, Speech and Signal Processing (ICASSP), 2002 IEEE International Conference on, 2002.

[92] D.W. Griffin and J.S. Lim, *Signal estimation from modified short-time Fourier transform*, IEEE Transactions on Acoustics, Speech, and Signal Processing **32(2)** (1984), 236–243.

[93] T. Gustafsson, B.D. Rao, and M. Trivedi, *Source localization in reverberant environments : modeling and statistical analysis*, Speech and Audio Processing, IEEE Transactions on **11** (2003), no. 6, 791 – 803.

[94] I. Guyon and A. Elisseeff, *An introduction to variable and feature selection*, Journal of Machine Learning Research **3** (2003), 1157–1182.

[95] Y. Han and C. Raphael, *Informed source separation of orchestra and soloist*, Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR), 2010, pp. 315–320.

[96] R.A. Harshman, *Foundations of the PARAFAC procedure : Models and conditions for an explanatory multimodal factor analysis*, UCLA Working Papers in Phonetics, 16, 84, 1970.

[97] T. J. Hastie and R. J. Tibshirani, *Generalized additive models*, Statistical Science **1** (1986), 297–310.

[98] ———, *Generalized additive models*, London : Chapman & Hall, 1990.

[99] M. Helén and T. Virtanen, *Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine*, Proc. 13th European Signal Processing Conference (EUSIPCO) (Antalaya, Turkey), 2005.

[100] R. Hennequin, *Décomposition de spectrogrammes musicaux informé par des modèles de synthèse spectrale*, Ph.D. thesis, Telecom ParisTech, 2001.

[101] R. Hennequin, B. David, and R. Badeau, *Score informed audio source separation using a parametric model of non-negative spectrogram*, IEEE International Conference on Acoustics, Speech, and Signal Processing (Prague, Czech Republic), may 2011.

[102] N. Henrich, C. d'Alessandro, and B. Doval, *Spectral correlates of voice open quotient and glottal flow asymmetry : theory, limits and experimental data*, Proceedings of EUROSPEECH (Aalborg, Denmark), September 2001, p. 1.

[103] J. Herre, *From joint stereo to spatial audio coding*, In Proc. Digital Audio Effects Workshop (DAFx (Naples, Italy), October 2004.

[104] J. Herre and S. Disch, *New concepts in parametric coding of spatial audio : From SAC to SAOC*, IEEE International Conference on Multimedia and Expo (ICME 2007) (Beijing, China), July 2007, pp. 1894–1897.

[105] J. Herre, K. Kjörling, J. Breebaart, C. Faller, S. Disch, H. Purnhagen, J. Koppens, J. Hilpert, J. Rödén, W. Oomen, K. Linzmeier, and K.S. Chong, *MPEG Surround-The ISO/MPEG Standard for Efficient and Compatible Multichannel Audio Coding*, Journal of the Audio Engineering Society **56 (11)** (2008), 932–955.

[106] T. Hofmann, *Probabilistic latent semantic analysis*, In Proc. of Uncertainty in Artificial Intelligence, UAI'99, 1999, pp. 289–296.

[107] R. Huber and B. Kollmeier, *PEMO-Q - a new method for objective audio quality assessment using a model of auditory perception*, IEEE Transactions on Audio, Speech, and Language Processing **14** (2006), no. 6, 1902 –1911.

[108] D.A. Huffman, *A method for the construction of minimum-redundancy codes*, Proceedings of IRE **40 (9)** (1952), 1098–1101.

[109] A. Hyvärinen, J. Karhunen, and E. Oja (eds.), *Independent component analysis*, Wiley and Sons, 2001.

[110] E. T. Jaynes and G. L. Bretthorst, *Probability Theory : The Logic of Science*, Cambridge University Press, 2003.

[111] A.G. Journel and C.J. Huijbregts, *Mining geostatistics*, Academic Press, London ; New York, 1978.

[112] T-P. Jung, S. Makeig, C. Humphries, T-W. Lee, M. J. McKeown, V. Iragui, and T. J. Sejnowski, *Removing electroencephalographic artifacts by blind source separation*, Psychophysiology **37** (2000), 163–178.

[113] C. Jutten, *Advances in nonlinear blind source separation*, In Proc. of the 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003, 2003, pp. 245–256.

[114] C. Jutten and J. Herault, *Blind separation of sources, part i : An adaptive algorithm based on neuromimetic architecture*, Signal Processing **24** (1991), no. 1, 1–10.

[115] C. Jutten and A. Taleb, *Source Separation : From Dusk Till Dawn*, Proc. Int. Symp. Independent Component Analysis and Blind Signal Separation (ICA), 2000, pp. 15–26.

[116] T. Kailath, A.H. Sayed, and B. Hassibi, *Linear Estimation (Prentice Hall Information and System Sciences Series)*, 1 ed., Prentice Hall, April 2000.

[117] M. Khadkevich, T. Fillon, G. Richard, and M. Omologo, *A probabilistic approach to simultaneous extraction of beats and downbeats*, Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on, mars 2012, pp. 1 –4.

[118] B.J. King and L. Atlas, *Single-channel source separation using complex matrix factorization*, IEEE Transactions on Audio, Speech, and Language Processing **19** (2011), no. 8, 2591 –2597.

[119] D.E. Kirk, *Optimal Control Theory : An Introduction*, Dover Publications, April 2004.

[120] W. B. Kleijn, *A basis for source coding*, March 2008, lecture notes KTH, Stockholm.

[121] W.B. Kleijn and M.Y. Kim, *Quantization with an adjustable codeword length penalty*, IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'11) (Prague, Czech Republic), May 2011, pp. 4228–4231.

[122] W.B. Kleijn and A. Ozerov, *Rate distribution between model and signal*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) (New Paltz, NY), Oct. 2007, pp. 243–246.

[123] R. Kompass, *A generalized divergence measure for nonnegative matrix factorization*, Neural Comput. **19** (2007), no. 3, 780–791.

[124] R.I. Kondor and J. Lafferty, *Diffusion kernels on graphs and other discrete input spaces*, International Conference on Machine Learning (ICML 2002), 2002, pp. 315–322.

[125] J. Kornycky, Banu Günel, and A. M. Kondoz, *Comparison of subjective and objective evaluation methods for audio source separation*, 2008, pp. 1–10.

[126] D. G. Krige, *A statistical approach to some basic mine valuation problems on the Witwatersrand*, Journal of the Chemical, Metallurgical and Mining Society **52** (1951), 119–139.

[127] J. Le Roux, E. Vincent, Y. Mizuno, H. Kameoka, N. Ono, and S. Sagayama, *Consistent Wiener filtering : Generalized time-frequency masking respecting spectrogram consistency*, Proc. 9th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA 2010) (St. Malo, France), September 2010, pp. 89–96.

[128] D. D. Lee and H. S. Seung, *Algorithms for non-negative matrix factorization*, Advances in Neural Information Processing Systems (NIPS), vol. 13, The MIT Press, April 2001, pp. 556–562.

[129] M. Li, J. Klejsa, and W. B. Kleijn, *Distribution preserving quantization with dithering and transformation*, Signal Processing Letters **17** (2010), no. 12, 1014–1017.

[130] A. Liutkus, R. Badeau, and G. Richard, *Informed source separation using latent components*, 9th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA'10) (St Malo, France), 2010.

[131] A. Liutkus, R. Badeau, and G. Richard, *Gaussian processes for underdetermined source separation*, IEEE Transactions on Signal Processing **59** (2011), no. 7, 3155 –3167.

[132] A. Liutkus, R. Badeau, and G. Richard, *Multi-dimensional signal separation with gaussian processes*, Proc. of IEEE Conf. on Statistical Signal Processing (SSP2011) (Nice, France), 2011.

[133] A. Liutkus, A. Dremeau, D. Alexiadis, S. Essid, and P. Daras, *Analysis of dance movements using Gaussian processes*, ACM Multimedia Grand Challenge, September 2012.

[134] A. Liutkus, S. Gorlow, N. Sturmel, S. Zhang, L. Girin, R. Badeau, L. Daudet, S. Marchang, and G. Richard, *Informed source separation : a comparative study*, Proceedings European Signal Processing Conference (EUSIPCO 2012), August 2012.

[135] A. Liutkus and P. Leveau, *Separation of music+effects sound track from several international versions of the same movie*, Audio Engineering Society Convention 128, May 2010.

[136] A. Liutkus, A. Ozerov, R. Badeau, and G. Richard, *Spatial coding-based informed source separation*, Proceedings European Signal Processing Conference (EUSIPCO 2012), August 2012.

[137] A. Liutkus, J. Pinel, R. Badeau, L. Girin, and G. Richard, *Informed source separation through spectrogram coding and data embedding*, Signal Processing **92** (2012), no. 8, 1937 – 1949.

[138] A. Liutkus, Z. Rafii, R. Badeau, B. Pardo, and G. Richard, *Adaptive filtering for music/voice separation exploiting the repeating musical structure*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), mars 2012, pp. 1 –4.

[139] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, *Text classification using string kernels*, Journal of Machine Learning Research **2** (2002), 419–444.

[140] D. MacKay, *Gaussian processes - a replacement for supervised neural networks ?*, Neural Information Processing Systems (NIPS), MIT Press, 1997.

[141] D.J.C. Mackay, *Introduction to Gaussian processes*, Neural Networks and Machine Learning (C.M. Bishop, ed.), Springer, 1998, pp. 133–165.

[142] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, *Online learning for matrix factorization and sparse coding*, J. Mach. Learn. Res. **11** (2010), 19–60.

[143] T. Masahiro and N. Masahide, *Evaluation method for dance action based on motion capture analysis*, Eizo Joho Media Gakkai Gijutsu Hokoku **29** (2005), 33–36.

[144] G. Matheron, *The intrinsic random functions and their applications*, Advances in Applied Probability **5** (1973), no. 3, 439–468.

[145] A. Melkumyan and F. Ramos, *A sparse covariance function for exact gaussian process inference in large datasets*, IJCAI'09 : Proceedings of the 21st international joint conference on Artifical intelligence (San Francisco, CA, USA), Morgan Kaufmann Publishers Inc., July 2009, pp. 1936–1942.

[146] Layer III MPEG-1 Audio, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 mbit/s – Part 3 : Audio*, ISO/IEC 11172-3 :1993 (1993).

[147] AAC MPEG-2 Advanced Audio Coding, *Information technology – Generic coding of moving pictures and associated audio information – Part 3 : Audio*, ISO/IEC 13818-3 :1998 (1998).

[148] M. Nakano, H. Kameoka, J. Le Roux, Y. Kitano, N. Ono, and S. Sagayama, *Convergence-guaranteed multiplicative algorithms for non-negative matrix factorization with beta-divergence*, Proc. IEEE International Workshop on Machine Learning for Signal In Processing (MLSP 2010), August 2010.

[149] M. Nakano, J. Le Roux, H. Kameoka, N. Ono, and S. Sagayama, *Infinite-state spectrum model for music signal analysis*, Proceedings of the ICASSP 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing, May 2011, pp. 1972–1975.

[150] H. Niessner and K. Reichert, *On computing the inverse of a sparse matrix*, International Journal for Numerical Methods in Engineering **19 (10)** (1983), 1513–1526.

[151] J. Nikunen and T. Virtanen, *Object-based audio coding using non-negative matrix factorization for the spectrogram representation*, 128th Audio Engineering Society Convention (AES 2010) (London, UK), May 2010.

[152] J. Nikunen, T. Virtanen, and M. Vilermo, *Multichannel audio upmixing based on non-negative tensor factorization representation*, IEEE Workshop Applications of Signal Processing to Audio and Acoustics (WASPAA) (New Paltz, New York, USA), October 2011.

[153] H.O. Oh, Y.W. Jung, A. Favrot, and C. Faller, *Enhancing Stereo Audio with Remix Capability*, AES 129th Convention Preprint 8290 (San Francisco, CA, USA), November 2010.

[154] R.K. Olsson and L.K. Hansen, *Linear state-space models for blind source separation*, Journal of Machine Learning Research **7** (2006), 2585–2602.

[155] A. Ozerov and C. Févotte, *Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation*, IEEE Transactions on Audio, Speech and Language Processing **18** (2010), no. 3, 550–563.

[156] A. Ozerov, C. Févotte, R. Blouet, and J.-L. Durrieu, *Multichannel nonnegative tensor factorization with structured constraints for user-guided audio source separation*, IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'11) (Prague), May 2011, pp. 257–260.

[157] A. Ozerov and W.B. Kleijn, *Flexible quantization of audio and speech based on the autoregressive model*, IEEE Asilomar Conference on Signals, Systems, and Computers (Asilomar CSSC'07) (Pacific Grove, CA), Nov. 2007.

[158] ———, *Optimal parameter estimation for model-based quantization*, Proc. IEEE Intl. Conf. Acoust. Speech Signal Processing (ICASSP'09), 2009, pp. 2497–2500.

[159] ———, *Asymptotically optimal model estimation for quantization*, IEEE Transactions on Communications **59** (2011), no. 4, 1031–1042.

[160] A. Ozerov, A. Liutkus, R. Badeau, and G. Richard, *Informed source separation : source coding meets source separation*, IEEE Workshop Applications of Signal Processing to Audio and Acoustics (WASPAA) (New Paltz, New York, USA), October 2011.

[161] A. Ozerov, A. Liutkus, R. Badeau, and G. Richard, *Coding-based informed source separation : Nonnegative tensor factorization approach*, IEEE Trans. on Audio, Speech and Language Processing (2012).

[162] A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot, *One microphone singing voice separation using source-adapted models*, IEEE Workshop Applications of Signal Processing to Audio and Acoustics (WASPAA), 2005, pp. 90–93.

[163] A. Ozerov, E. Vincent, and F. Bimbot, *A general flexible framework for the handling of prior information in audio source separation*, Audio, Speech, and Language Processing, IEEE Transactions on **PP** (2011), no. 99, 1.

[164] L. Parra and C. Spence, *Convolutive blind separation of non-stationary sources*, Speech and Audio Processing, IEEE Transactions on **8** (2000), no. 3, 320 –327.

[165] M. Parvaix, *Séparation de sources audio informée par tatouage pour mélanges linéaires instantanés stationnaires*, Ph.D. thesis, Université de Grenoble, Institut polytechnique de Grenoble, 2010.

[166] M. Parvaix and L. Girin, *Informed source separation of underdetermined instantaneous stereo mixtures using source index embedding*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (Dallas, Texas), 2010.

[167] ———, *Informed source separation of linear instantaneous under-determined audio mixtures by source index embedding*, IEEE Transactions on Audio, Speech, and Language Processing **19** (2011), no. 6, 1721 –1733.

[168] M. Parvaix, L. Girin, and J.-M. Brossier, *A watermarking-based method for single-channel audio source separation*, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (Taipei, Taiwan), 2009, pp. 101–104.

[169] M. Parvaix, L. Girin, and J.-M. Brossier, *A watermarking-based method for informed source separation of audio signals with a single sensor*, IEEE Transactions on Audio, Speech, and Language Processing **18** (2010), no. 6, 1464–1475.

[170] M. Parvaix, L. Girin, L. Daudet, J. Pinel, and C. Baras, *Hybrid coding/indexing strategy for informed source separation of linear instantaneous under-determined audio mixtures*, Proceedings of 20th International Congress on Acoustics (Sydney, Australia), Aug. 2010.

[171] R. Pasco, *Source coding algorithms for data compression*, Ph.D. thesis, Standford University, 1976.

[172] J. Pinel, L. Girin, and C. Baras, *A high-capacity watermarking technique for audio signals based on mdct-domain quantization*, Proceedings of the 20th International Congress on Acoustics (Sydney), 2010.

[173] ———, *Une technique de tatouage haute-capacité pour signaux musicaux au format cd-audio*, Actes du 10ème Congrés Francais d'Acoustique (Lyon), Société Francaise d'Acoustique, 2010.

[174] J.H. Plasberg and W.B. Kleijn, *The sensitivity matrix : Using advanced auditory models in speech and audio processing*, IEEE Transactions on Audio, Speech, and Language Processing **15** (2007), no. 1, 310–319.

[175] M.D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M.E. Davies, *Sparse Representations in Audio and Music : from Coding to Source Separation*, Proceedings of the IEEE. **98** (2010), 995–1005.

[176] J. Quiñonero-Candela, C. E. Rasmussen, and R. Herbrich, *A unifying view of sparse approximate Gaussian process regression*, The Journal of Machine Learning Research **6** (2005), 1939–1959.

[177] Z. Rafii and B. Pardo, *A simple music/voice separation method based on the extraction of the repeating musical structure*, Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, may 2011, pp. 221 –224.

[178] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning (adaptive computation and machine learning)*, The MIT Press, 2005.

[179] R. Renner and U. Maurer, *About the mutual (conditional) information*, Tech. report, Department of Computer Science, Swiss Federal Institute of Technology (ETH Zurich), 2002.

[180] J.J. Rissanen, *Generalized kraft inequality and arithmetic coding*, IBM Journal of Research and Developpement **20** (1976), 198–203.

[181] M. N. Schmidt, *Function factorization using warped Gaussian processes*, (2009).

[182] M. N. Schmidt and H. Laurberg, *Non-negative matrix factorization with Gaussian process priors*, Computational Intelligence and Neuroscience **ID 361705** (2008).

[183] M.N. Schmidt and M. Morup, *Nonnegative matrix factor 2-D deconvolution for blind single channel source separation*, ICA '06 : Proc. of the 8th Int. Conf. on Independent Component Analysis and Signal Separation, 2006.

[184] B. Scholkopf and A.J. Smola, *Learning with kernels : Support vector machines, regularization, optimization, and beyond*, MIT Press, Cambridge, MA, USA, 2001.

[185] M. Seeger, *Gaussian processes for machine learning.*, Int. J. Neural Syst. **14** (2004), no. 2, 69–106.

[186] C.E. Shannon, *A mathematical theory of communication*, Bell System Technical Journal **27** (1948), 379–423.

[187] M. Shashanka, B. Raj, and P. Smaragdis, *Sparse overcomplete latent variable decomposition of counts data*, Neural Information Processing Systems (NIPS), 2007.

[188] ———, *Probabilistic latent variable models as non-negative factorizations*, Computational Intelligence and Neuroscience special issue on Advances in Non-negative Matrix and Tensor Factorization **May** (2008), 12–20.

[189] T. Shiratori and K. Ikeuchi, *Synthesis of dance performance based on analyses of human motion and music*, IPSJ Online Transactions **1** (2008), 80–93.

[190] T. Shiratori, A. Nakazawa, and K. Ikeuchi, *Detecting dance motion structure through music analysis*, IEEE Int'l Conference on Automatic Face and Gesture Recognition (New Paltz, NY, USA), October 2004, pp. 1–6.

[191] U. Simsekli and A.T. Cemgil, *Score guided musical source separation using generalized coupled tensor factorization*, Proceedings European Signal Processing Conference (EUSIPCO 2012), August 2012.

[192] P. Smaragdis, *Non-negative matrix factor deconvolution ; extraction of multiple sound sources from monophonic inputs*, ICA '04 : Proc. of the 8th Int. Conf. on Independent Component Analysis and Signal Separation, 2004.

[193] ———, *Relative-pitch tracking of multiple arbitrary sounds*, Journal of the Ac. Soc. of America **125** (2009), 3406–3413.

[194] P. Smaragdis and J.C. Brown, *Non-negative matrix factorization for polyphonic music transcription*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2003, pp. 177–180.

[195] P. Smaragdis and G. J. Mysore, *Separation by "humming" : User-guided sound extraction from monophonic mixtures*, Proceedings IEEE Workshop Applications of Signal Processing to Audio and Acoustics (WASPAA), 2009, pp. 69–72.

[196] P. Smaragdis, M. Shashanka, B. Raj, and G. J. Mysore, *Probabilistic factorization of non-negative data with entropic co-occurrence constraints*, ICA '09 : Proc. of the 8th Int. Conf. on Independent Component Analysis and Signal Separation, 2009.

[197] E. Snelson, *Local and global sparse gaussian process approximations*, Proceedings of Artificial Intelligence and Statistics (AISTATS) (San Juan, Puerto Rico), vol. 2, March 2007, pp. 524–531.

[198] E. Snelson and Z. Ghahramani, *Sparse Gaussian processes using pseudo-inputs*, Neural Information Processing Systems (NIPS), MIT press, 2006, pp. 1257–1264.

[199] P. Sollich, M. Urry, and C. Coti, *Kernels and learning curves for Gaussian process regression on random graphs*, Advances in Neural Information Processing Systems 22 (Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, eds.), 2009, pp. 1723–1731.

[200] N. Sturmel and L. Daudet, *Informed source separation using iterative reconstruction*, arXiv :1202.2075v1.

[201] N. Sturmel, A. Liutkus, J. Pinel, L. Girin, S. Marchand, G. Richard, R. Badeau, and L. Daudet, *Linear mixing models for active listening of music productions in realistic studio conditions*, 132th AES convention, Budapest, in press, 2012.

[202] A. Taleb and C. Jutten, *Source separation in post-nonlinear mixtures*, IEEE Transactions on Signal Processing **47** (1999), no. 10, 2807–2820.

[203] Y. W. Teh and M. I. Jordan, *Hierarchical Bayesian nonparametric models with applications*, Bayesian Nonparametrics : Principles and Practice (N. Hjort, C. Holmes, P. Müller, and S. Walker, eds.), Cambridge University Press, 2010.

[204] H. Valpola, A. Honkela, and J. Karhunen, *An ensemble learning approach to nonlinear dynamic blind source separation using state-space models*, Proc. Int. Joint Conf. Neural Networks IJCNN '02, vol. 1, 2002, pp. 460–465.

[205] S. van de Par, A. Kohlrausch, R. Heusdens, J. Jensen, and S.H. Jensen, *A perceptual model for sinusoidal audio coding based on spectral integration*, EURASIP Journal on Applied Signal Processing **9** (2005), 1292–1304.

[206] J. Vanhatalo and A. Vehtari, *Modelling local and global phenomena with sparse gaussian processes*, Proc. of 24th Conference on Uncertainty in Artificial Intelligence (UAI) (Helsinky, Finland), AUAI Press, July 2008, pp. 571–578.

[207] J. E. Vila-Forcen, O. Koval, and S. Voloshynovskiy, *Distributed single source coding with side information*, IS&T/SPIE16th annual symposium : electronic imaging 2004, image processing (San Jose, California, USA), January 2004.

[208] E. Vincent, S. Araki, and P. Bofill, *The 2008 signal separation evaluation campaign : A community-based approach to large-scale evaluation*, ICA '09 : Proc. of the 8th Int. Conf. on Independent Component Analysis and Signal Separation (Berlin, Heidelberg), 2009, pp. 734–741.

[209] E. Vincent, S. Araki, F.J. Theis, G. Nolte, P. Bofill, H. Sawada, A. Ozerov, B.V. Gowreesunker, D. Lutter, and N.Q.K. Duong, *The signal separation evaluation campaign (2007–2010) : Achievements and remaining challenges*, Signal Processing **92** (2012), no. 8, 1928–1936.

[210] E. Vincent, S. Arberet, and R. Gribonval, *Underdetermined instantaneous audio source separation via local Gaussian modeling*, Independent Component Analysis and Signal Separation. Lecture Notes in Computer Science. (Paraty, Brésil), vol. 5441/2009, Springer-Verlag Berlin Heidelberg 2009, 2009, pp. pp 775–782.

[211] E. Vincent, N. Bertin, and R. Badeau, *Adaptive harmonic spectral decomposition for multiple pitch estimation*, IEEE trans. on Audio, Speech and Language Proc. (TASLP) **18(3)** (2010), 528–537.

[212] E. Vincent, R. Gribonval, and C. Févotte, *Performance measurement in blind audio source separation*, IEEE Transactions on Audio, Speech, and Language Processing **14** (2006), no. 4, 1462 –1469.

[213] E. Vincent, R. Gribonval, and M. Plumbley, *Oracle estimators for the benchmarking of source separation algorithms*, Signal Processing **87** (2007), no. 8, 1933 – 1950.

[214] E. Vincent, G.M. Jafari, A.S Abdallah, D.M. Plumbley, and E.M. Davies, *Probabilistic modeling paradigms for audio source separation*, Machine Audition : Principles, Algorithms and Systems (W. Wang, ed.), IGI Global, 2010, pp. 162–185.

[215] T. Virtanen, *Algorithm for the separation of harmonic sounds with time-frequency smoothness constraint*, Proc. of the 6th Conf. on Digital Audio Effects (DAFx-03) (London, UK), September 2003, pp. 35–40.

[216] G.K. Wallace, *The JPEG still picture compression standard*, Commun. ACM **34** (1991), 30–44.

[217] N. Wiener, *Extrapolation, interpolation, and smoothing of stationary time series with engineering applications.*, MIT Press, 1949 (English).

[218] C. K. I. Williams, *Prediction with Gaussian processes : From linear regression to linear prediction and beyond*, Learning and Inference in Graphical Models (M. I. Jordan, ed.), Kluwer, 1999.

[219] C.K.I. Williams, *Computation with infinite neural networks*, Neural Computation **10** (1998), no. 5, 1203–1216.

[220] J.W. Woods, *Multidimensional signal, image, and video processing and coding*, Academic Press, Inc., Orlando, FL, USA, 2006.

[221] O. Yilmaz and S. Rickard, *Blind separation of speech mixtures via time-frequency masking*, IEEE Trans. on Signal Processing **52** (2004), no. 7, 1830–1847.

[222] R. Zamir, *The rate loss in the wyner-ziv problem*, IEEE Trans. Inform. Theory **42** (1996), 2073–2084.

[223] S. Zhang, *Informed source separation by local inversion*, Tech. report, Gipsa-lab, 2012, Projet ANR DReaM.

[224] D.Y. Zhao, J. Samuelsson, and M. Nilsson, *On entropy-constrained vector quantization using Gaussian mixture models*, IEEE Transactions on Communications **56** (2008), no. 12, 2094–2104.

Index