2012-ENST-079

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité «Traitement du Signal et des Images »

présentée et soutenue publiquement par

Manuel MOUSSALLAM

le 18 décembre 2012

Représentations Redondantes et Hiérarchiques pour l'Archivage et la Compression de Scènes Sonores

Directeur de thèse : **Gaël RICHARD**
Directeur de thèse : **Laurent DAUDET**

Jury

M. Rémi GRIBONVAL, Directeur de Recherche, INRIA, CNRS	Rapporteur
M. Laurent GIRIN, Professeur, Gipsa-Lab, CNRS - Grenoble-INP	Rapporteur
M. Stéphane MALLAT, Professeur, Département Informatique, Ecole Normale Supérieure	Examinateur
M. Liva RALAIVOLA, Professeur, LIF , Université Aix-Marseille II	Examinateur
M. Pierre LEVEAU, Docteur, Directeur de la recherche AUDIONAMIX	Examinateur
M. Laurent DAUDET, Professeur, Institut Langevin, Univ. Paris 7 Diderot	Directeur
M. Gaël RICHARD, Professeur, LTCI, Telecom ParisTech	Directeur

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

T H È S E

Remerciements

Autant le dire tout de suite, les remerciements constituent sans doute la partie la plus difficile à écrire d'une thèse. Immanquablement on se verra partagé entre l'envie de faire court et classique (et donc de remercier ceux qui ont vraiment contribué à l'élaboration du texte final) et celle de faire plaisir à tous ceux qui comptent, et qui cherchent peut-être en ce moment même avec angoisse la mention de leur nom, voire son rang d'apparition !

Ayant particulièrement raté mes remerciements oraux le jour de la soutenance, la pression n'en est que plus importante lors de l'écriture de ces derniers, au point qu'après plus d'un mois d'hésitation (et de nombreux essais) je suis arrivé à la conclusion suivante : je vais faire les deux, et que chacun trouve ici ce qu'il est venu y chercher. Une chose mérite néanmoins d'être soulignée : chaque remerciement est sincère, depuis les membres du jury jusqu'aux autres gens.

Remerciements courts et en rapport avec l'élaboration de cette thèse

En premier lieu je tiens à remercier très profondément les rapporteurs de ce texte, Laurent Girin et Rémi Gribonval. D'abord pour le temps et le travail qu'ils ont bien voulu consacrer à cette thèse, ensuite pour la qualité et la précision des très nombreuses remarques sur le manuscrit qu'ils ont pu me communiquer, et qui, je l'espère, sont prises en compte dans cette version finale. Je remercie ensuite Stéphane Mallat d'avoir accepter de présider au jugement de ce travail, Liva Ralaivola pour son intérêt bienveillant et pour m'avoir fait découvrir les matroïdes, et Pierre Leveau, à qui je dois certainement la possibilité originale de faire cette thèse.

Je ne remercierais jamais assez Gaël et Laurent d'avoir accepté de m'encadrer durant ces trois années. Grâce à eux j'ai appris énormément. Ils ont su me guider tout au long de ce travail et dans le même temps me laisser chercher des solutions par moi-même. J'en garderais un excellent souvenir ainsi qu'une amitié durable pour chacun deux. Je souhaite par ailleurs à leurs futurs doctorants communs, d'avoir la chance de pouvoir les écouter chanter en duo au fond d'un karaoké japonais, une expérience fascinante en soi.

Enfin il y a tous ceux qui m'ont aidé lors de la rédaction de ce manuscrit. Mes collègues : Angélique, Antoine et Sébastien pour leurs relectures, conseils et commentaires pertinents, mais aussi Mounira, Benoît, Rémi et Thomas pour leur soutien et leur bonne humeur constante durant ces trois années. Enfin je dois remercier tout particulièrement Mathilde pour le courage dont elle a fait preuve en acceptant (en plus de m'avoir supporté durant cette phase tendue qu'est la rédaction) de relire et corriger des parties de ce manuscrit.

Remerciements étendus

De ces trois ans passés au sein de l'équipe AAO à Telecom ParisTech, je garderais un excellent souvenir. L'ambiance joyeuse qui y règne est d'abord le fait de son directeur Gaël Richard, mais aussi de ses membres. Je remercie donc encore une fois mes collègues doctorants et post-doctorants du début : Benoît, Rémi, Mounira, Antoine, Thomas, Sébastien mais aussi ceux arrivés en cours de route : François, Angélique, Nicolas, Aymeric, Anne-claire, Cécilia et ceux qui sont partis : Romain, Félicien, Cyril, Honoré, Benoît et mes deux camarades du tout-début : Valentina et Shamil. A ceux-la il faut ajouter des permanents très sympathiques : Slim, Roland, Yves, Bertrand ainsi que le petit nouveau Alexandre et bien sûr Fabrice.

Une part très importante de ma vie (en dehors de la recherche) est liée à la musique. Je profite donc de ce paragraphe pour saluer mes amis et camarades du groupe *Spaghetti Warriors :* Simon, Alex, Yann et Valentin, avec qui j'ai la chance de jouer depuis plus de 10 ans maintenant. Encore plus fort, je salue le Big Band de Joinville le pont, rejoint en 1998, et tout particulièrement François parmi eux. Je salue également les membres de l'éphémère groupe Alarm Alarm : Benoît, Olivier, Arnaud et Arthur et du toujours vivant Tak'One : Romain, Seb et Romain. Je serais très heureux de rejouer avec chacun d'eux très prochainement.

Je tiens à saluer également tous mes camarades de promotion ATIAM, et parmi eux Ben, Arnaud, Benoît, Rémi, Gilles, Greg, Filou, Pierre, Antonio, Simon, Mike, Delphine, David et tous les autres.

Pour finir, je voudrais remercier tous ceux qui ont pu venir assister à ma soutenance (et/ou au pot qui a suivi) : Bounine, Julien, Simon, Pauline, Safou, Alex, Yann, Nour, François, Eve, Olivier et également mon frère Yves (Nadim et Lisa, vous êtes excusés), Gaby, Nathalie, Anne, Maya, Raphaël et bien sur Mathilde, qui a supporté mon existence nocturne durant trois mois et mon existence tout court durant ces sept dernières années.

Enfin, je remercie mes parents. J'espère les avoir rendus fiers et je leur dédie cette thèse.

Résumé

L'objet de cette thèse est l'analyse et le traitement automatique de grands volumes de données audio. Plus particulièrement, on s'intéresse à l'archivage, tâche qui regroupe, au moins, deux problématiques : la compression des données, et l'indexation du contenu de celles-ci. Ces deux problématiques définissent chacune des objectifs, parfois concurrents, dont la prise en compte simultanée s'avère donc difficile. Au centre de cette thèse, il y a donc la volonté de construire un cadre cohérent à la fois pour la compression et pour l'indexation d'archives sonores.

Les représentations parcimonieuses de signaux dans des dictionnaires redondants ont récemment montré leur capacité à remplir une telle fonction. Leurs propriétés ainsi que les méthodes et algorithmes permettant de les obtenir sont donc étudiés dans une première partie de cette thèse. Le cadre applicatif relativement contraignant (volume des données) va nous amener à choisir parmi ces derniers des algorithmes itératifs, appelés également *gloutons*.

Une première contribution de cette thèse consiste en la proposition de variantes du célèbre *Matching Pursuit* basées sur un sous-échantillonnage aléatoire et dynamique de dictionnaires. L'adaptation au cas de dictionnaires temps-fréquence structurés (union de bases de cosinus locaux) nous permet d'espérer une amélioration significative des performances en compression de scènes sonores. Ces nouveaux algorithmes s'accompagnent d'une modélisation statistique originale des propriétés de convergence usant d'outils empruntés à la théorie des valeurs extrêmes.

Les autres contributions de cette thèse s'attaquent au second membre du problème d'archivage : l'indexation. Le même cadre est cette fois-ci envisagé pour mettre à jour les différents niveaux de structuration des données. Au premier plan, la détection de redondances et répétitions. A grande échelle, un système robuste de détection de motifs récurrents dans un flux radiophonique par comparaison d'empreintes est proposé. Ses performances comparatives sur une campagne d'évaluation du projet QUAERO confirment la pertinence de cette approche.

L'exploitation des structures pour un contexte autre que la compression est également envisagé. Nous proposons en particulier une application à la séparation de sources informée par la redondance pour illustrer la variété de traitements que le cadre choisi autorise. La synthèse des différents éléments permet alors d'envisager un système d'archivage répondant aux contraintes par la hiérarchisation des objectifs et des traitements.

Abstract

The main goal of this work is automated processing of large volumes of audio data. Most specifically, one is interested in archiving, a process that encompass at least two distinct problems: data compression and data indexing. Jointly addressing these problems is a difficult task since many of their objectives may be concurrent. Therefore, building a consistent framework for audio archival is the matter of this thesis.

Sparse representations of signals in redundant dictionaries have recently been found of interest for many sub-problems of the archival task. Sparsity is a desirable property both for compression and for indexing. Methods and algorithms to build such representations are the first topic of this thesis. Given the dimensionality of the considered data, *greedy* algorithms will be particularly studied.

A first contribution of this thesis is the proposal of a variant of the famous Matching Pursuit algorithm, that exploits randomness and sub-sampling of very large time frequency dictionaries. We show that audio compression (especially at low bit-rate) can be improved using this method. This new algorithms comes with an original modeling of asymptotic pursuit behaviors, using order statistics and tools from extreme values theory.

Other contributions deal with the second member of the archival problem: indexing. The same framework is used and applied to different layers of signal structures. First, redundancies and musical repetition detection is addressed. At larger scale, we investigate audio fingerprinting schemes and apply it to radio broadcast on-line segmentation. Performances have been evaluated during an international campaign within the QUAERO project. Finally, the same framework is used to perform source separation informed by the redundancy.

All these elements validate the proposed framework for the audio archiving task. The layered structures of audio data are accessed hierarchically by greedy decomposition algorithms and allow processing the different objectives of archival at different steps, thus addressing them within the same framework.

Table des matières

Chapitre 1

Introduction

Dans son *Quart-livre*, paru en 1552, RABELAIS fait traverser au géant Pantagruel et ses compagnons d'aventure une vaste étendue appelée mer Glaciale, dans laquelle les sons d'une bataille s'étant déroulée l'hiver précédent se sont retrouvés gelés, emprisonnés par la glace. Au contact du navire, les sons dégèlent et les personnages peuvent alors entendre le fracas des canons et les cris poussés par les soldats plusieurs mois plus tôt. Cette séquence est communément comprise comme une réflexion humaniste sur le langage et le bouleversement du rapport au temps et au savoir qu'autorise l'écriture. Mais au sens premier, RABELAIS est le premier à imaginer ce que nous appelons aujourd'hui un enregistrement d'une scène sonore.

Depuis les enregistrements sont devenus une réalité tangible. D'abord sur des cylindres de cire, puis des disques de vinyle, des bandes magnétiques et désormais sur des supports numériques, les paroles et les autres sons peuvent être *gelés* et *dégelés* presque à l'infini. La mer glaciale est devenue un océan gigantesque et qui ne cesse de grandir. Avec l'explosion du volume de données audiovisuelles disponibles sont apparus de nouveaux questionnements que RABELAIS ne pouvait soupçonner. Comment traiter ces enregistrements, les classer, les stocker, les conserver et les étiqueter de sorte que l'information (les humanistes auraient dit le *savoir*) qu'ils contiennent demeure à la fois accessible et préservée ?

Cette question, c'est celle de l'archivage et c'est elle qui est à l'origine de ce travail de thèse. L'archivage est, on va le voir, un problème intéressant car complexe, au sens où on regroupe sous cette dénomination un ensemble de tâches bien distinctes, parfois même concurrentes. En tout état de cause, l'archivage de scènes sonores offre un cadre pratique aux problématiques modernes des Sciences et Technologies de l'Information et de la Communication (STIC), et en particulier à celles du traitement des signaux audio.

Dans cette thèse, nous allons nous intéresser à l'étude de représentations des scènes sonores qui rendent possible leur archivage. Mais avant de présenter ces méthodes, il nous faut expliciter le contexte et les contraintes qui délimitent le périmètre de notre étude.

1.1 Contexte et problématiques de l'Archivage

1.1.1 Notions et Définitions

L'archivage est un problème communément mal défini. Lorsqu'on parle d'archiver un document électronique, un livre, un objet, on cherche généralement à conserver cet objet dans des conditions garantissant son intégrité et son accès, tout en réduisant l'espace nécessaire à son stockage. C'est une

vision pratique de l'archivage comme processus de conservation. Dans cette vision, graver un morceau de musique en format numérique non compressé sur un support amovible, par exemple, est une forme d'archivage. En revanche, réaliser une compression avec pertes de ce même morceau n'entre pas dans le domaine de l'archivage.

Cette vision centrée sur la conservation est complétée par une vision documentaliste : archiver c'est classer, trier, ranger et décrire. Toutes ces tâches impliquent l'existence de méta-données ou encore de descripteurs sur le contenu des archives. Il ne s'agit pas seulement de préserver l'information existante, dans cette vision, l'archivage est un processus créateur d'information. Étiqueter ou classer par nom de l'artiste une collection de morceaux de musique est alors un exemple d'archivage. Plus formellement, l'archivage est constitué de deux disciplines : *l'archivistique* et la *diplomatique*. La première s'attache à collecter, analyser, étiqueter et mettre en valeur des archives, tandis que la seconde traite des modes de conservation, de garanties d'intégrité et d'authenticité. Un système complet d'archivage, dont le schéma-bloc est présenté Figure 1.1.1, nécessite donc de réaliser analyse et compression en parallèle. Une introduction à l'archivage et à ses problématiques se trouve dans le *Nouveau glossaire de l'archivage* de M.A. Chabin, disponible gratuitement en ligne.

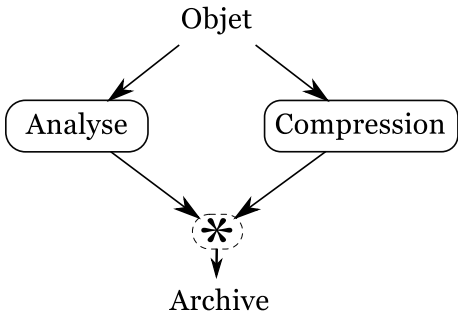

Figure 1.1.1: Schéma bloc d'un système d'archivage

Normes et standards Historiquement, l'archivage est une tâche dévolue aux bibliothécaires et aux moines. Elle consiste à assurer la conservation de textes (éventuellement de leurs copies) et leur indexation en vue d'une consultation future. Au cours des siècles, toutes les grandes civilisations maîtrisant l'écriture ont compris l'importance de la conservation du savoir. La nécessité de mettre en place des systèmes de centralisation des écrits, ne serait-ce que pour des besoins de contrôle et de censure se fait plus évidente avec la banalisation de l'imprimerie en Europe. En France, le principe du dépôt légal de tout ouvrage édité remonte à 1537 et les Archives Nationales conservent les documents administratifs depuis la Révolution de 1789.

La nature des objets archivés évolue au cours du temps. Des manuscrits aux traités, comptes-rendus, procès verbaux et autres formulaires administratifs, puis des supports différents (étalons de poids, de mesures, sceaux, etc..), jusqu'aux œuvres d'art, tapisseries, tableaux puis photographies et enfin enregistrements de scènes sonores et visuelles, sur des supports analogiques puis numériques. La dématérialisation a fait apparaître des besoins spécifiques dont une liste exhaustive se trouve dans [CRE06] de même qu'une analyse des enjeux et des contraintes engendrées.

Il existe en France depuis 2009 une norme décrivant les Systèmes d'Archivage Électronique (SAE), à savoir la norme AFNOR NF Z 42-013, reprise début janvier 2012 dans la norme ISO 14621. Ces normes préconisent essentiellement la mise en place de deux systèmes concomitants :

- Un système de stockage garantissant l'intégrité des données
- Un système de description du contenu à l'aide de méta-données

Plutôt vagues, elles ne font qu'officialiser la dualité inhérente à tout système d'archivage. Les deux facettes – conservation et indexation – sont nécessaires et c'est cette nature double qui rend le problème intéressant.

Archivage de scènes sonores En France, l'Institut National de l'Audiovisuel (INA) est créé en 1975, avec parmi ses missions l'archivage des données audiovisuelles et en particulier les flux télévisuels et radiophoniques (aujourd'hui, l'INA est le dépôt légal d'une vingtaine de stations de radio et 88 chaînes de télévision). De nombreuses autres bases de données audiovisuelles existent également, citons par exemple les archives ethnologiques CORPUS [1].

Les problématiques rencontrées par ces institutions se rattachent au domaine de la fouille de données (*data mining*). A ce titre, Youtube®, propriété du moteur de recherche Google® depuis 2006, constitue sans doute la plus grande base de données audiovisuelles mondiale, accessible en ligne (plus de deux milliards de visionnages par jour). Pour autant, l'intégrité des objets – de même que la qualité de l'indexation – est loin d'être garantie (recours à du codage destructif, lacunes dans le contrôle des étiquettes, redondances, etc...). Ce type de plate-forme ne remplit donc pas les critères d'un SAE.

De même, de plus en plus de services d'écoute de musique à la demande proposent l'accès (payant) à des données musicales en ligne (Spotify®, Deezer®, Grooveshark®, MusicMe®, etc...) dont la qualité de l'étiquetage est également variable. Avec l'augmentation continue des moyens en bande passante, le codage sans pertes commence à faire son apparition sur ces plate-formes, mais la possibilité donnée aux utilisateurs d'enrichir eux-mêmes le catalogue rend parfois le travail d'indexation sisyphien. Plus grave, la pérennité de ces plate-formes privées (c.f. MegaUpload) est loin d'être garantie, et la législation en la matière est appelée à évoluer dans les prochaines années.

Parmi la multitude d'offres commerciales existantes, citons encore LastFM® dont la particularité consiste à confier au plus grand nombre le soin d'indexer (et même de recommander) les morceaux de musique. Ces technologies participatives dépoussièrent les problématiques monacales de l'archivage et les placent au centre du développement des outils de traitement multimédia massifs.

Dans le monde académique, de très nombreux chercheurs se sont attachés aux problèmes liés à la fouille de données. Certains de ces travaux sur les scènes sonores ont aboutis à la création d'acteurs important, comme The Echo Nest [2] fondé par T. Jehan sur la base de ses travaux de thèse au MIT [Jeh05]. En Europe, le projet QUAERO [3], lancé en 2008 par la France et l'Allemagne regroupe des acteurs académiques et industriels sur des tâches de fouille de données, notamment de scènes sonores. Plusieurs axes de recherche menés au sein de ce projet portent sur des problématiques en lien avec l'archivage. Parmi les laboratoires français collaborant sur ces tâches se trouve l'Institut Telecom et plus particulièrement le LTCI à Telecom ParisTech dans lequel cette thèse s'est déroulée.

Archivage et Représentation Ni dans ces normes ni dans les différentes études ne sont imposés de formats ou de règles définissant les propriétés générales des archives. Un SAE est uniquement défini par sa finalité. Les moyens à mettre en œuvre pour implémenter ce système sont laissés à la discrétion de l'utilisateur, et l'on devine que c'est la nature des objets à archiver qui contraint le plus fortement une architecture d'archivage.

Nous sommes intéressés d'une façon générale, par l'archivage de signaux numériques. Dans ce cadre, l'archivage va inévitablement être une transformation de l'objet archivé en un autre, plus facile à stocker, ranger, étiqueter ou transmettre. Il est donc intéressant de voir l'archivage de signaux comme un processus de représentation. Il y a plusieurs définitions possibles de la représentation d'un

1. CORPUS : http ://www.corpus-ir.fr/index.php ?page=cae
2. http ://the.echonest.com/
3. http ://www.quaero.org/

objet. Dans une formulation mathématique, une représentation est une relation $\mathcal{R}$ liant un objet x appartenant à un espace $\mathcal{X}$ à un objet y dans un espace $\mathcal{Y}$. On dira ainsi que y est le représentant de x dans l'espace $\mathcal{Y}$ des représentations et on notera cette relation par

$$\begin{array}{ccc} \mathcal{X} & \rightarrow & \mathcal{Y} \\ x & \overset{\mathcal{R}}{\rightarrow} & y \end{array}$$

Selon le contexte, cette relation peut se comprendre de différentes façons :

- y est un substitut de x
- y est un équivalent de x
- y est l'image de x

Chacune de ces formulations pose un rapport particulier entre le représentant y et le représenté x. Par abus de langage on dira souvent que y est une représentation de x. La multiplicité de cette relation (*c-à.d.* le nombre de représentants de x que l'on peut construire et le nombre d'éléments dont le représentant est y) est un paramètre important. Par exemple, $\mathcal{R}$ peut réaliser une bijection entre deux espaces. Les propriétés souhaitables de $\mathcal{R}$ sont directement liées aux problématiques que l'on cherche à résoudre.

1.1.2 Problématiques

Si l'on envisage des représentations des scènes sonores permettant leur archivage, on peut lister six propriétés souhaitables de ces dernières :

Fidélité : Dans un but de conservation des données, la représentation se doit d'être inversible ou quasi inversible, c'est-à-dire que la reconstruction des données doit être non seulement possible, mais dans l'idéal sans pertes. Ce critère peut s'évaluer à l'aide de mesures de distance (normes) ou de distorsion subjectives. Soit $\mathcal{R}$ une représentation, on dénote, s'il existe, $\mathcal{R}^\dagger$ son inverse (exact ou approché). On cherche à minimiser (idéalement annuler) une mesure $L_d(x, \mathcal{R}^\dagger(\mathcal{R}(x)))$ de distorsion entre un signal x et l'inverse $\mathcal{R}^\dagger(\mathcal{R}(x))$ de sa représentation.

Comparabilité : Si nous voulons pouvoir détecter les redondances dans une collection de données, leur représentation doit permettre une comparaison simple et efficace. Deux signaux équivalents doivent avoir une représentation équivalente. Ce critère peut s'évaluer par des performances empiriques sur une base de données de test. Il peut également se comprendre comme une forme de robustesse aux distorsions ou au bruit additif.

$$\forall (x, y) \in \mathcal{X}^2, x \simeq y \rightarrow \mathcal{R}(x) \simeq \mathcal{R}(y) \tag{1.1.1}$$

Idéalement, $\mathcal{R}$ préserve les structures et similarités inter-signaux. Par exemple, une application k-lipschitzienne dans des espaces métriques $(\mathcal{X}, L_\mathcal{X})$ et$(\mathcal{Y}, L_\mathcal{Y})$ assure la relation suivante :

$$\forall (x, y) \in \mathcal{X}^2, L_\mathcal{Y}(\mathcal{R}(x), \mathcal{R}(y)) \leq k \cdot L_\mathcal{X}(x - y) \tag{1.1.2}$$

Séparabilité :

C'est le pendant de la comparabilité. Au delà des signaux identiques, des classes de signaux doivent pouvoir être distinguées par leurs représentations. Ce critère peut s'évaluer par des performances empiriques sur une base de données de test. Deux signaux appartenant à des classes différentes (par exemple voix/musique) doivent pouvoir être séparés dans le domaine de la représentation.

Concision

Optimiser l'espace de stockage nécessaire est fondamental pour l'archivage. En conséquence, la représentation doit être la plus réduite possible. Ce critère peut se mesurer par des mesures de débit ou de taille des données. De façon équivalente, on cherche à minimiser une fonctionnelle de parcimonie sur la représentation. On favorisera, par exemple, les représentations qui réduisent la pseudo-norme ℓ_0 d'un vecteur x, c'est-à-dire le nombre d'éléments non nuls de x :

$$\forall x \in \mathcal{X}, \|\mathcal{R}(x)\|_0 \leq \|x\|_0 \tag{1.1.3}$$

Simplicité :

C'est plus naturellement la mesure inverse de complexité qui sera utilisée. En raison des contraintes volumiques très fortes, l'archivage est un problème complexe qui doit être traité simplement. Cet objectif est également mesurable sous la forme de temps de calcul, de nombre d'opérations à virgule flottante, ou de ressources nécessaires.

Lisibilité : C'est un objectif plus difficile à définir, la représentation doit permettre de réaliser des tâches de haut niveau sémantique sans avoir à reconstruire/inverser la représentation. Le fait qu'un être humain puisse interpréter une représentation est en soi appréciable. Plus généralement on cherche des représentations qui permettent des traitements et manipulations de haut niveau sur les signaux. La transcription automatique de musique, la reconnaissance de locuteur ou encore la séparation de sources sont autant d'exemples de traitements envisageables sur des signaux audio.

Chacune des propriétés décrites ci-dessus dessine une contrainte particulière sur la représentation. Il est vraisemblable qu'il ne soit pas possible de les satisfaire toutes en même temps. On peut noter en effet des couples de propriétés *a priori* contradictoires (Fidélité-Concision, Séparabilité-Comparabilité, Lisibilité-Simplicité). On retrouve ces couples de contraintes pour différents problèmes courants de traitements des signaux :

Reconstruction sous contrainte de parcimonie Le couple Fidélité-Concision est décrit par un problème d'optimisation très étudié, celui du codage et du compromis débit-distorsion :

$$\min_{\mathcal{R}} L_d\left(x, \mathcal{R}^\dagger(\mathcal{R}(x))\right) \text{ et } \min_{\mathcal{R}} \|\mathcal{R}(x)\|_p \tag{1.1.4}$$

ou L_d est une mesure de distorsion entre x et la reconstruction $\mathcal{R}^\dagger(\mathcal{R}(x))$, et $\|\cdot\|_p$ est une mesure de parcimonie sur la représentation qui définit un débit.

Discrimination La séparabilité et la comparabilité sont des objectifs communément atteints par des techniques d'apprentissage. Le signal est décrit (représenté) par des descripteurs (ici notre représentation $\mathcal{R}$). Une tâche d'indexation (*p.ex.* de classification) se fait sur ces descripteurs à l'aide d'une fonction $\mathcal{F}$ apprise (optimisée) sur un ensemble d'échantillons x_i, ou donnée *a priori*. Le problème est ensuite de maximiser une fonction de score $\mathcal{S}$ sur une base de données de test B_{test} :

$$\max_{\mathcal{R},\mathcal{F}} \mathcal{S}\left(\mathcal{F}(\mathcal{R}), B_{test}\right) \tag{1.1.5}$$

Il est donc malaisé d'évaluer directement la performance d'une représentation sur ce type de tâche, car c'est le couple Données-Représentation qui doit être optimisé. On peut par exemple chercher une séparation linéaire sous la forme d'une frontière de décision dans l'espace de la représentation.

Contraintes Algorithmiques Dans le cas général, il existe une infinité d'algorithmes pour calculer une représentation. La simplicité d'une représentation $\mathcal{R}$ peut alors se mesurer à l'aune de la complexité de Kolmogorov. Néanmoins, le calcul d'une représentation obéit forcément à des contraintes algorithmiques qui vont peser sur la simplicité et la lisibilité. En premier lieu, la représentation doit être calculable. Cela nécessite l'existence d'un algorithme fini capable de la calculer pour toute une classe de signaux. Les contraintes sur cet algorithme sont alors :

1. Une garantie de convergence
2. Une complexité réduite (*p. ex.* en nombre d'opération à virgule flottante)
3. Une quantité de ressources nécessaire finie.

En particulier les algorithmes basés sur des recherches exhaustives s'avèrent rapidement inutilisables sur de problèmes de dimension importante – comme c'est souvent le cas en analyse de scènes sonores – les temps de calcul devenant rapidement prohibitifs. On leur préfèrera alors des approches sous-optimales mais plus rapides. En définitive, avec une représentation doit être fourni un algorithme de complexité minimale, utilisant des ressources finies et dont la convergence est garantie.

Pour résumer, on a affaire à un problème (l'archivage) mal posé car sur-contraint. En plus de devoir trouver une représentation adaptée (satisfaisant toutes les contraintes) il nous faut encore prouver qu'elle est calculable et proposer un algorithme pour la construire. Le cadre de ce travail étant posé, nous pouvons maintenant présenter les stratégies et outils que nous allons mettre en place pour tenter de résoudre ce problème.

1.2 Stratégies et Outils

L'archivage rassemble différents problèmes, et les traiter simultanément semble irréalisable. Il va donc falloir envisager des stratégies de contournement et pour cela, inventorier les outils dont nous disposons, au premier rang desquels on trouve les techniques de représentations et la notion de parcimonie.

1.2.1 Représentation de signaux

On considérera essentiellement dans ce travail les signaux numériques (discrets) sous la forme de vecteurs dans des espaces Hilbertiens de dimension finie. Plus généralement, on peut définir un signal x comme un élément dans un espace Hilbertien $\mathcal{X}$. On cherche une représentation de x sous la forme d'une application $\mathcal{R}$ de $\mathcal{X}$ dans $\mathcal{Y}$ où $\mathcal{Y}$ est appelé espace de la représentation.

Un exemple d'application est celle qui relie un signal de dimension N (à valeurs réelles ou complexes) à sa transformée de Fourier :

$$\mathcal{R}_{Fourier} : \begin{cases} \mathcal{X} \to \mathcal{X} \\ x[n] \to \hat{x}[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-2i\pi k \frac{n}{N}} dt \end{cases} \tag{1.2.1}$$

Un autre exemple d'application associe à une variable aléatoire X discrète à N symboles x_i de loi $P(X = x_i) = p_i$ son entropie H. Soit $(\Omega, \mathcal{A}, P)$ un espace probabilisé :

$$\mathcal{R}_{Entropie} : \begin{cases} (\Omega, \mathcal{A}, P) \to \mathbb{R}^+ \\ X \to H(X) = -\sum_{i=1}^{N} p_i \log p_i \end{cases} \tag{1.2.2}$$

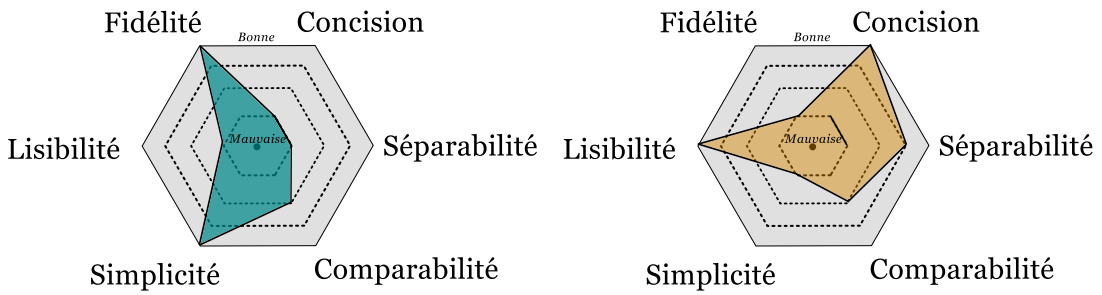

FIGURE 1.2.1: Diagramme radar des objectifs pour deux représentations. Gauche pour la transformée de Fourier $\mathcal{R}_{Fourier}$. Droite pour la représentation par étiquettes $\mathcal{R}_{Label}$.

On peut envisager beaucoup d'autres applications, pas uniquement sous la forme de fonctions. Par exemple une application qui, à une image ou un son associe une description du contenu sous la forme d'une ou plusieurs étiquettes l_i parmi un ensemble L :

$$\mathcal{R}_{Label} : \begin{cases} \mathbb{R}^N \to L \\ x \to \{l_i\} \end{cases} \tag{1.2.3}$$

Ces trois exemples illustrent les principales caractéristiques des représentations de signaux. En général une représentation présente un intérêt dans la mesure où elle permet de mettre à jour de l'information sur un signal. $\mathcal{R}_{Fourier}$ est une fonction calculable, linéaire, inversible qui dévoile l'information fréquentielle contenue dans un signal. $\mathcal{R}_{Entropie}$ n'est ni linéaire ni inversible, mais apporte néanmoins une information cruciale pour le codage et la transmission de messages. Enfin, $\mathcal{R}_{Label}$ n'est ni linéaire, ni inversible ni même obligatoirement calculable, peut nécessiter un apprentissage, mais dans une certaine mesure, apporte une information sémantique sur le signal. Si on s'en réfère aux propriétés souhaitables définies plus haut pour l'archivage, il est possible d'illustrer les forces et faiblesses de ces représentations sur des diagrammes radar tel que présenté Figure 1.2.1. On a alors une idée plus nette des compromis auquel on doit faire face ; chaque représentation va remplir certains critères mais échouer sur d'autres.

Une représentation rend accessible une information d'une certaine nature, c'est le contexte qui détermine l'intérêt, la valeur de cette information. Avant de choisir une représentation, il peut donc être nécessaire de limiter l'étude à une classe de signaux pour lesquels on dispose d'information *a priori* sur leur nature ou leur contexte d'analyse. Dans cette étude, on s'intéresse aux signaux numériques en général, de scènes sonores en particulier.

Même en posant ces limites, trouver une représentation qui remplisse tous les critères simultanément va s'avérer problématique. Avant de présenter la stratégie adoptée dans cette thèse, il faut introduire un concept essentiel dans ce travail, la parcimonie.

1.2.2 Parcimonie

Parcimonie en philosophie : Rasoir d'OCKHAM La parcimonie est une caractéristique souhaitable reconnue depuis longtemps pour permettre de choisir entre deux modèles. On la retrouve par exemple chez ARISTOTE sous la forme suivante : "il vaut mieux prendre des principes moins nombreux, et de nombre limité" [4]. Ce principe est généralement connu sous le nom de rasoir d'OCKHAM et stipule

4. ARISTOTE, *Physique,* I, 4, 188a17

qu'entre deux explications possibles, il faut préférer celle qui est la plus simple, au sens de celle qui nécessite le moins d'hypothèses, le nombre le plus faible de causes.

Philosophiquement, cet argument a servi à de nombreuses reprises, on le trouve notamment chez Wittgenstein dans sa critique du solipsisme[5] reprise depuis de nombreuses fois (par exemple par D. Deutsch[6]). Le rasoir d'Ockham est également utilisé pour justifier l'adoption de modèles mathématiques. MacKay justifie ainsi l'utilisation de l'inférence Bayésienne comme une application de ce principe [Mac03]. En effet, entre deux modèles explicatifs équivalents, la fonction de vraisemblance va privilégier celui qui nécessite le moins de paramètres.

Parcimonie en sciences En mathématiques, la parcimonie sous la forme de concision d'une démonstration, est une qualité prisée. On le verra dans cette thèse, des problèmes fondamentaux mettant en œuvre la parcimonie sont au cœur de nombreux théorèmes récents d'algèbre et d'analyse. Historiquement, on retrouve la parcimonie dans de nombreux éléments et énoncés classiques (*p.ex.* tout entier naturel pair supérieur à 2 est la somme de deux nombres premiers – Conjecture de Goldbach).

Dans le domaine des sciences physiques et de l'information, l'utilisation de la parcimonie est caractéristique d'un changement de paradigme récent. Le paradigme en vigueur au XIXe siècle (et pour une large part du XXe siècle) avait pour critère central l'énergie (ou la puissance). Les problèmes pratiques au cœur de la révolution industrielle s'écrivaient essentiellement sous la forme de minimisation de puissance mécanique ou électrique, de recherche d'équilibre énergétique et de maximisation de rendements. Ainsi, du principe fondamental de la dynamique, aux lois de la thermodynamique jusqu'à l'équation de Navier-Stokes, c'est l'énergie d'un système, son évolution et ses transformations qui sont décrits. Une solution était préférable si elle optimisait un critère énergétique et ce pour des raisons pratiques évidentes. Minimiser l'énergie nécessaire à la propulsion d'une locomotive, trouver les points d'équilibre d'un système mécanique, les puits de potentiels énergétiques, minimiser l'erreur quadratique entre une prévision et une observation, c'est ce type de tâches qui sous-tendait le paradigme énergétique.

La parcimonie est invoquée lorsque la solution du paradigme énergétique n'est plus satisfaisante, ce qui est souvent le cas des problèmes sous-déterminés ou combinatoires ou encore lorsque l'on s'intéresse à la modélisation des causes d'un phénomène plutôt qu'a décrire son comportement.

Dans le domaine de la biologie, la parcimonie est à la base du système de caractérisation des distances génétiques (phylogénétique). La fréquence d'occurrence des mutations étant extrêmement faible, le nombre de variations entre deux génomes est un marqueur fiable de l'évolution. A l'inverse, la modification génétique de plus faible cardinalité est de fait la plus probable.

En algorithmique, la parcimonie sous-tend la notion de complexité (au sens de Kolmogorov), de taille du langage et de ressources (espace mémoire, nombre d'opérations en virgule flottante, etc...). En linguistique, la parcimonie se retrouve dans la définition des grammaires et des ensembles de règles de transformation.

En traitement de l'information enfin, la parcimonie se retrouve d'abord sous la forme de l'entropie pour le codage de sources et de canaux. Le développement des canaux de communication a amené les chercheurs (et notamment Shannon) à proposer des mesures de quantité d'information et de débit minimum nécessaire à la transmission des messages.

5. Wittgenstein, *Tractatus Logico-Philosophicus* , Aphorisme 5.631
6. D. Deutsch, *The Fabric of Reality* , Chap. 4

Depuis quelques décennies, les problèmes de représentations parcimonieuses de signaux occupent une place grandissante dans les travaux scientifiques en traitement du signal. Les résultats fondamentaux présentés par Candès *et al [CRT06]* et Donoho [Don06] en 2006 et l'avènement consécutif de la discipline d'acquisition comprimée (en anglais *Compressed Sensing* ou *Compressive Sampling*) ont confirmé cet engouement pour la recherche de solutions aux problèmes combinatoires à l'aide de contraintes de parcimonie.

Dans ce travail, la parcimonie est envisagée non seulement dans une optique de réduction de dimension, mais aussi d'extraction d'information. Si la parcimonie de la représentation présente un intérêt évident pour le stockage et la compression de scènes sonores, nous verrons que la structure de cette parcimonie rend également possible leur indexation.

1.2.3 Hiérarchisation des contraintes

Les objectifs et contraintes de l'archivage sont trop nombreux pour être pris en compte simultanément. Une stratégie va donc consister à hiérarchiser ces objectifs. Pour cela, il faut d'abord remarquer que les tâches que regroupe l'archivage correspondent à des profondeurs d'analyse différentes :

- Une classification simple (*p.ex.* voix/musique/bruit/silence) suppose une représentation très synthétique, qui ne retire que la substantifique moelle du signal, ses caractéristiques les plus grossières. Pour ce type de tâches, nous sommes intéressés par le traitement rapide, simple, de grands volumes de données (collections de scènes sonores, flux radiophoniques de plusieurs jours etc..).
- La transcription (*p.ex.* de parole, de musique), la détection de similarités (*p.ex.* musicales, de locuteurs, etc..), la classification fine (*p.ex.* en genres musicaux) et toutes les tâches d'indexation plus complexes, nécessitent une analyse plus poussée des scènes sonores. Les échelles considérées sont plus petites que pour la classification simple (*p.ex.* pièces musicales de plusieurs secondes, voire minutes)
- La compression, le stockage (en vue d'une décompression) supposent une analyse en profondeur, qui rend possible la synthèse fidèle de scènes sonores. L'échelle considérée est alors celle de l'échantillon même, ou de la trame d'analyse, soit de quelques dizaines à une centaine de millisecondes.

Le tableau présenté en Figure 1.2.2 résume ces considérations. Ainsi, les différents objectifs doivent être considérés avec plus ou moins de force selon le niveau d'analyse recherché et les tâches envisagées. Pour la compression, les principaux objectifs seront la concision et la fidélité, ce sous-problème de l'archivage se ramène donc à un problème de parcimonie sous contrainte de reconstruction (*c-à-d.* de *synthèse*).

Pour l'indexation, différents niveaux d'analyse peuvent aussi être définis, en fonction de la difficulté de la tâche. Dans ces situations, l'archivage se ramène aux problèmes de détection de motifs récurrents, de similarité et de classification (*c-à-d.* d'*analyse*).

L'objectif de cette thèse est donc de proposer des méthodes de représentations qui soient pertinentes à toutes ces échelles et donc à la fois pour l'analyse et la synthèse de scènes sonores.

Nous avons choisi d'étudier les représentations parcimonieuses, soit la représentation des scènes sonores comme combinaisons linéaires d'un nombre réduit d'éléments d'un dictionnaire redondant. Pour construire ces représentations, nous nous concentrons sur un type d'algorithme appelé *Matching Pursuit*. Dans ce travail, nous étudions son potentiel tant pour la compression que pour l'indexation

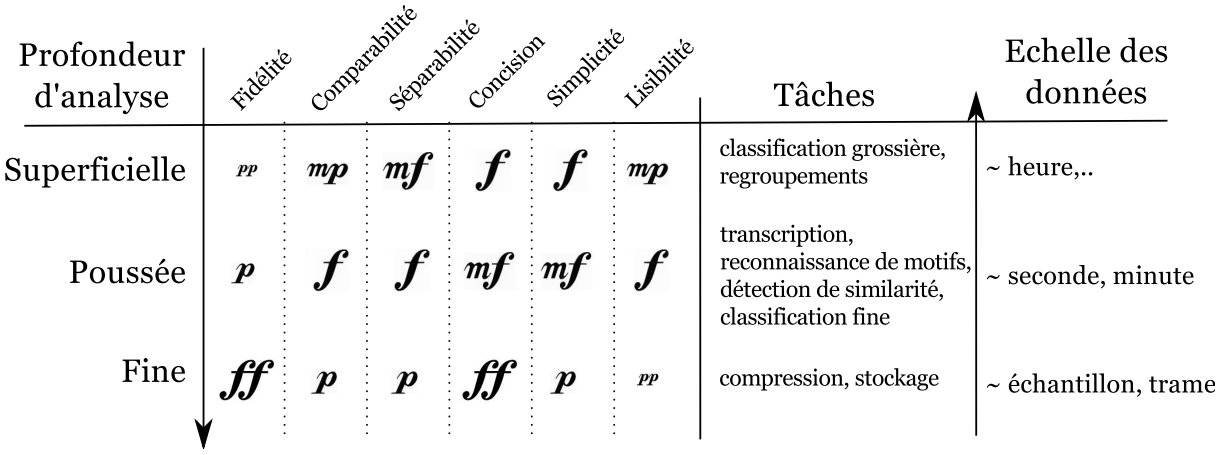

FIGURE 1.2.2: Hiérarchisation des objectifs en fonction de la profondeur d'analyse et de l'échelle des données considérées. Les symboles utilisés expriment la force de la contrainte correspondante.

avec l'espoir que les performances sur ces deux missions justifient la pertinence de cette approche pour résoudre les problèmes d'archivage de scènes sonores.

Notre ambition est d'utiliser ce type d'algorithmes aux différents niveaux d'analyse exposés plus haut, et d'expliciter les conditions qui peuvent permettre de respecter toutes les contraintes de l'archivage. Une architecture d'archivage, basée sur une hiérarchisation des contraintes, pourra alors être envisagée.

1.3 Contributions

Les contributions de cette thèse se situent à différents niveaux, correspondant aux échelles de données et de profondeur d'analyse.

1.3.1 Algorithmes gloutons aléatoires et dynamiques

Dans le but d'adapter la représentation aux contraintes spécifiques à chaque niveau d'analyse, il est intéressant d'envisager les approches qui construisent itérativement ces représentations. Dans ce travail, nous nous concentrons sur les algorithmes dits gloutons de la famille *Matching Pursuit*. Nous proposons en particulier une variante utilisant une séquence de sous-dictionnaires pseudo-aléatoire au lieu d'un dictionnaire fixe. Nous montrons que dans une application de compression de scènes sonores, cet algorithme peut permettre d'obtenir une amélioration substantielle du compromis débit-distorsion avec un modèle simple de codage. Cette méthode a fait l'objet d'une publication de revue dans *Signal Processing* en octobre 2012 [MDR12a].

Nous proposons également une étude du comportement asymptotique de ces algorithmes. En utilisant une modélisation originale de la convergence basée sur les statistiques d'ordre, nous verrons que ce comportement incite à étendre la paramétrisation dynamique de l'algorithme à la taille du sous-dictionnaire au fil de la décomposition.

1.3.2 Méthodes de détections de motifs récurrents

Les algorithmes gloutons sont ensuite envisagés pour la détection de structures et plus précisément de redondances et de similarités. Nous nous sommes d'abord intéressés à la définition d'une mesure de similarité basée sur les principes de codage distribué. Nous proposons une méthode, appelée factorisation, qui évalue la similarité de deux signaux par la faculté de la représentation de l'un à représenter l'autre. Ce faisant, les bases d'un codage joint sont également posées. Ce travail à fait l'objet d'une publication lors de la conférence ICASSP 2011 [MDR11].

A une tout autre échelle, nous avons étudié la pertinence d'une décomposition superficielle par algorithme glouton pour une tâche de détection de récurrences par comparaison d'empreintes. Au sein d'une collaboration avec S. Fenet de Telecom ParisTech, nous avons proposé une architecture robuste de segmentation de flux radiophonique en objets récurrents. Dans cette architecture, nous montrons que des empreintes très simples basées sur une décomposition superficielle permettent d'obtenir des résultats compétitifs, comme le montrent les scores obtenus lors d'une campagne d'évaluation menée au sein du projet QUAERO. Ce travail a fait l'objet d'une publication lors de la conférence EUSIPCO 2012 [FMG+12].

1.3.3 Méthode de séparation de sources communes

Nous proposons enfin une méthode originale de séparation de sources informée par la redondance. Là encore, un algorithme glouton est proposé pour résoudre un problème de construction d'approximations jointes de scènes sonores répétitives. Cette application semble *a priori*, assez éloignée des problématiques de l'archivage. Mais dans ce cas particulier, la séparation peut se comprendre comme un effet secondaire, indirect, résultant d'une approche de décomposition distribuée. Cet exemple d'application illustre un phénomène intéressant pour l'archivage : dans cette situation en effet, nous observons que les critères de reconstruction et de séparation de sources sont optimisés simultanément. Ceci ouvre des perspectives plus larges dans lesquelles compression et indexation sont interdépendantes et gagnent à être effectuées de concert. Ce travail a fait l'objet d'une publication lors de la conférence EUSIPCO 2012 [MRD12].

1.4 Plan du manuscrit

Le manuscrit est divisé en trois parties. La première se consacre à l'état de l'art et ne contient pas de contributions originales. Celles-ci se répartissent dans les parties 2 et 3. Le détail des chapitres est le suivant :

Partie 1 : Techniques de Représentations Parcimonieuses

Chapitre 2 Ce chapitre s'attache à présenter les représentations usuelles des signaux audio ainsi que les méthodes de l'état de l'art sur les représentations parcimonieuses. Il permet de situer ce travail dans ce domaine de recherche très actif, en présentant la formulation générale du problème de parcimonie sous contrainte de reconstruction et l'éventail des méthodes et algorithmes s'attaquant à ce problème.

Chapitre 3 Ce chapitre s'attarde sur les algorithmes *gloutons* qui sont au coeur de ce travail. Il tente de dresser un inventaire des variantes existantes dans l'état de l'art, tant en terme de règles de mise à jour que de critères de sélection.

Partie 2 : Poursuites Aléatoires et Dynamiques

Chapitre 4 Ce chapitre présente une variante de *Matching Pursuit* développée durant cette thèse pour la compression de scènes sonores. L'algorithme y est présenté ainsi qu'une adaptation au cas de dictionnaires temps-fréquence. Les principaux résultats sur une tâche de codage sont présentés.

Chapitre 5 Ce chapitre étend la réflexion et les concepts développés pour l'algorithme présenté au chapitre précédent et propose une étude du comportement dynamique des algorithmes gloutons. L'étude asymptotique de ce comportement permet d'envisager une évolution intelligente de la taille des sous-dictionnaires utilisés au cours de la décomposition. Des résultats préliminaires sur une tâche de codage sont également présentés.

Partie 3 : Redondances et Structures

Chapitre 6 Ce chapitre introductif de la 3e partie commence par décrire les différents niveaux de structure qu'il est utile de considérer dans un contexte d'archivage. Il introduit ensuite les méthodes existantes pour découvrir et exploiter ces structures et présente quelques exemples d'applications pratiques.

Chapitre 7 Ce chapitre présente les méthodes proposées pour la détection de motifs récurrents à différentes échelles. Sur des échelles courtes, un algorithme de factorisation des représentations est présenté, qui permet en outre de définir une mesure de similarité efficace. Sur des échelles plus grandes, une empreinte acoustique construite sur une décomposition très superficielle est utilisée sur une tâche de segmentation de flux radiophoniques en objets redondants.

Chapitre 8 Dans ce chapitre, nous présentons une méthode originale de séparation de sources, à l'aide d'un algorithme de décompositions jointes de scènes répétitives.

Enfin, le Chapitre 9 conclut cette thèse et expose quelques-unes des perspectives ouvertes par ce travail.

Publications en lien avec cette thèse

Articles parus dans des revues internationales

[MDR12a] M. Moussallam, L. Daudet, et G. Richard, "Matching Pursuits with Random Sequential Subdictionaries," Signal Processing, vol. 92, pp. 2532–2544, 2012.

Articles parus dans les actes de conférences internationales

[MRD12] M. Moussallam, G. Richard, et L. Daudet, "Audio Source Separation informed by Redundancy with greedy multiscale Decomposition," in EUSIPCO 2012, pp. 2644–2648.

[FMG+12] S. Fenet, M. Moussallam, Y. Grenier, L. Daudet, et G. Richard, "A Framework for Fingerprint-Based detection of Repeating Objects in Multimedia Streams," in EUSIPCO 2012, pp. 1464–1468.

[MDR12b] M. Moussallam, L. Daudet, et G. Richard, "Random time-frequency Subdictionary design for sparse representation with greedy algorithms," in ICASSP 2012.

[MDR11] M. Moussallam, L. Daudet, et G. Richard, "Audio Signal Representations for Factorization in the sparse Domain," in ICASSP 2011, pp. 513–516.

[MFRD10] M. Moussallam, T. Fillon, G. Richard, et L. Daudet, "How Sparsely Can A SIgnal Be Approximated While Keeping Its Class Identity," in Workshop on Music and Machine Learning , ACM Multimedia, 2010.

Autres publications

- Y. Moussallam, C. Oppenheimer, A. Aiuppa, G. Giudice, M. Moussallam, et P. Kyle, "Hydrogen emissions from Erebus volcano, Antarctica," Bulletin of Volcanology, vol. 74(9), pp. 2109-2120, 2012.

Première partie

Techniques de Représentations Parcimonieuses

Chapitre 2

Représentations de signaux Audio

Dans ce chapitre, nous tenterons de dresser un état de l'art du problème de représentation des signaux en général, et des scènes sonores en particulier. Pour cela nous aborderons en premier lieu les représentations usuelles de signaux audio en section §2.1, réparties en trois catégories. Les représentations *sémantiques (*en section 2.1.1) sont les plus souvent rencontrées par le grand public. Elles procèdent de la transcription du signal vers un langage – naturel, musical ou autre – mais ne sont généralement pertinentes que pour un sous-ensemble de scènes sonores. Les représentations par paquets de descripteurs (section 2.1.2), sont très utilisées pour la classification, la segmentation, et plus généralement l'extraction d'information du signal. Enfin, parmi les représentations inversibles des signaux audio, celles basées sur des transformées temps-fréquence seront particulièrement étudiées dans la section 2.1.3.

Dans un second temps, nous présenterons en section §2.2 le formalisme particulier qui sous-tend cette thèse, celui des représentations parcimonieuses. Ce cadre très général nous permettra de présenter différents angles de vue. L'apprentissage de dictionnaire (section 2.2.2), problème voisin qui tire également profit des contraintes de parcimonie, mais pour lequel il est encore plus malaisé de trouver l'optimum. A l'opposé, l'utilisation de dictionnaires structurés (section 2.2.3) permet de réduire grandement la complexité des méthodes et présente de nombreux avantages pour les signaux non stationnaires de grande dimension tels que les scènes sonores. Enfin, une facette désormais très importante est celle qui s'intéresse à la reconstruction parcimonieuse de signaux sous-échantillonnés (section 2.2.4), au cœur du champ du *Compressed Sensing* qui occupe ces dernières années une grande partie de la communauté des représentations parcimonieuses.

La dernière section §2.3 de ce chapitre sera consacrée aux algorithmes et méthodes de calcul des représentations mentionnées, avec une subdivision relativement classique en trois groupes : les méthodes convexes (section 2.3.1), Bayésiennes (section 2.3.2) et itératives (section 2.3.3).

2.1 Représentations usuelles de signaux audio

Pour illustrer les différentes représentations usuelles des signaux audio, nous utiliserons quatre exemples monophoniques illustrant la diversité des scènes sonores :

—∘ Un enregistrement de glockenspiel solo (5 secondes)

—∘ Un enregistrement de voix d'homme (langue allemande, 5 secondes)

—∘ Un extrait d'un morceau de musique populaire (chant, guitare, batterie, basse, 5 secondes)

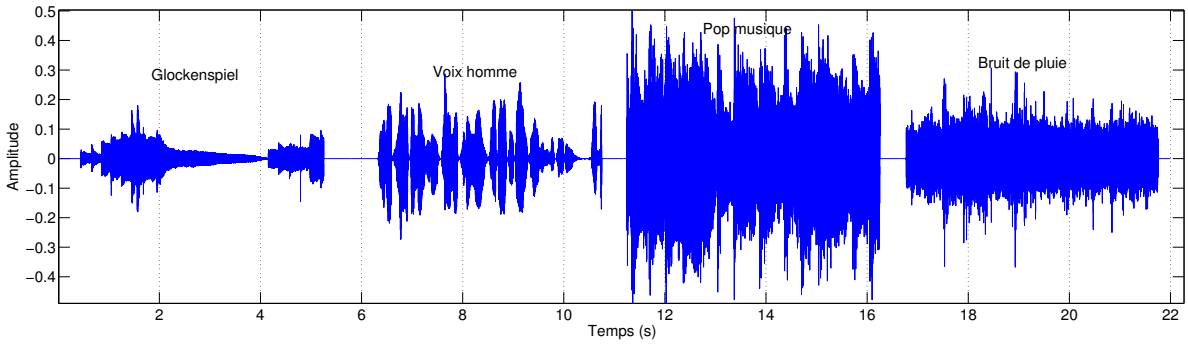

FIGURE 2.1.1: Formes d'ondes des exemples de scènes sonores. Fréquence d'échantillonnage de 32000 Hz.

—∘ Un enregistrement de bruit de pluie sur gazon en extérieur (5 secondes)

La figure 2.1.1 présente les formes d'onde (échantillonnées à 32000 Hz) de ces quatre scènes.

2.1.1 Représentations sémantiques

On parle de représentation *sémantique* lorsque celle-ci est porteuse de *sens*, ce qui suppose un contexte, une **ontologie** pré-existante. Par ontologie on entend un ensemble de concepts structurés, c'est à dire une terminologie et un ensemble logique de description des relations entre les termes. L'étendue de cet espace est fortement limitée par les présupposés qui fondent ces concepts. En conséquence, il sera difficile d'envisager des espaces sémantiques pouvant contenir l'ensemble des représentations des scènes sonores existantes. Au contraire, plus un espace sémantique est riche de concepts, plus le sous-ensemble de signaux ayant un représentant dans cet espace sera réduit.

La musique, et en particulier la musique occidentale, est un exemple intéressant d'espace sémantique auquel il serait souhaitable de pouvoir associer des représentations de scènes sonores. Néanmoins, parmi les quatre exemples de scènes sonores présentées, seuls la première et la troisième se prêtent à des représentations musicales. Les partitions, les grilles d'accord, l'écriture du rythme, tous les moyens d'écriture et d'analyse musicale sont pertinents sur ces deux exemples, mais ne peuvent s'appliquer à la voix humaine parlée ni à un enregistrement de pluie. Des compositeurs contemporains s'attachent certes à étendre l'écriture musicale à ce type de scènes, mais chaque compositeur - sinon chaque œuvre - adopte un formalisme propre. Le passage d'un signal audio vers une représentation musicale (Figure 2.1.2) est appelé transcription et constitue en soi un domaine de recherche important. Une introduction à cette thématique se trouve dans les travaux de thèse de N. BERTIN [Ber09].

Citons également les travaux de RAIMOND [RA07] visant à intégrer les concepts musicaux au sein du formalisme OWL [1] (pour *Ontology Web Language*) à la base des techniques d'extraction d'information souvent regroupées sous le nom de Web sémantique.

Le langage définit également un espace sémantique. La deuxième scène sonore possède une représentation cohérente dans cet espace, et d'une certaine manière l'extrait de musique populaire également à travers le texte chanté. Le passage d'un enregistrement de voix vers une représentation en langage naturel (Figure 2.1.3) définit le champ de la reconnaissance automatique de parole (RAP). Une vue d'ensemble des techniques de RAP se trouve dans le livre de J.P. HATON [HCF+06].

1. http ://www.w3.org/TR/owl-features/

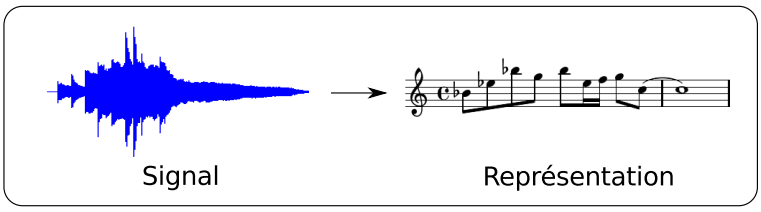

FIGURE 2.1.2: Transcription : Représentation sémantique d'une scène sonore (extrait numéro 1) utilisant un langage musical (partition).

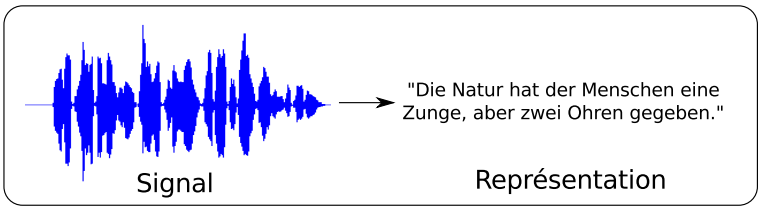

FIGURE 2.1.3: Transcription : Représentation sémantique d'une scène sonore (extrait numéro 2) utilisant un langage naturel (allemand).

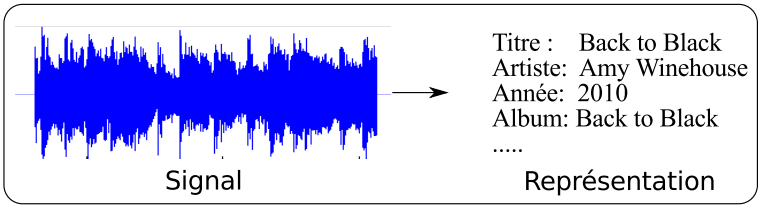

FIGURE 2.1.4: Indexation : Représentation sémantique d'une scène sonore (extrait numéro 3) utilisant une collection d'étiquettes.

Un champ important de recherche de la communauté du traitement du signal s'intéresse à la recherche d'information musicale (MIR pour *Music Information Retrieval*). Une approche commune de MIR est d'associer à une scène sonore une représentation sémantique sous la forme d'une collection d'étiquettes, choisie parmi un ensemble pré-existant. Le contexte sémantique est dans ce cas, limité par la variété des étiquettes disponibles. Le passage du signal vers une collection d'étiquettes (Figure 2.1.4) est un problème complexe souvent traité par des méthodes d'apprentissage statistique. Un ensemble de signaux de tests est préalablement étiqueté à la main et sert à entraîner un système de classification, que l'on utilise ensuite sur des données nouvelles (non étiquetées). Ce processus découple d'une certaine façon le problème de l'interprétation sémantique de celui de la représentation. Le *sens* n'apparait que dans la mesure où les étiquettes ont une signification pour celui qui les manipule (voir l'expérience proposée par STURM [SN12]). La représentation, elle, n'a pour seul but que d'associer un signal à un objet dans un espace défini. A proprement parler, il s'agit donc plutôt de représentations intermédiaires entre le signal et un espace sémantique.

Nous pouvons supposer que l'interprétation humaine des scènes sonores est basée sur des représentations sémantiques. Ainsi à l'écoute du bruit de la pluie, un individu peut y associer des images, des odeurs. A la musique, l'être humain associe des émotions, qui bien souvent lui sont propres de par son histoire personnelle. Toutes ces représentations internes sont porteuses de sens mais leur contexte d'existence se restreint à un groupe voire un seul individu, ou même parfois à un instant donné

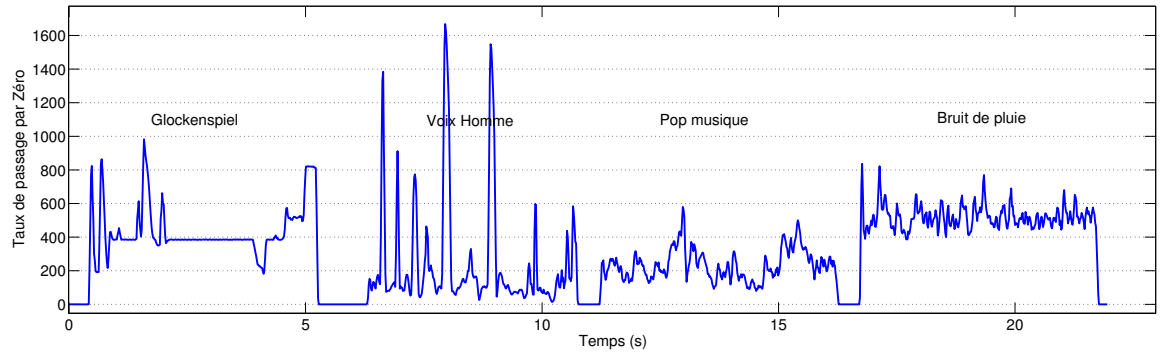

FIGURE 2.1.5: Taux de passage par zéro au cours du temps (trames de 32 ms) pour 4 scènes sonores

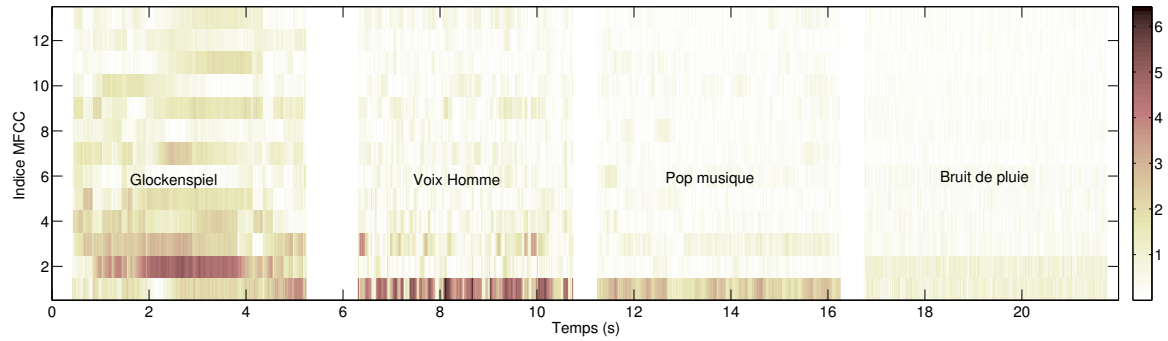

FIGURE 2.1.6: 13 premiers MFCC (valeur absolue) de 4 scènes sonores (glockenspiel, voix homme, musique pop et bruit de pluie).

de la vie d'un individu précis. Cette impossibilité de généralisation pose problème et questionne sur l'opportunité de l'utilisation de ce type de représentations dans le contexte de ce travail.

Passer du signal à ces représentations sémantiques est un vrai défi car cela suppose l'existence d'une ontologie dans laquelle ces représentations ont un sens, et de telles ontologies ne peuvent être définies pour l'ensemble des scènes sonores envisagées dans ce travail. Pourtant, ces représentations semblent idéales pour réaliser des tâches d'indexation. Si l'on s'en réfère aux objectifs définis dans l'introduction (voir 1.1.2 page 4), les représentations sémantiques remplissent quatre critères : Comparabilité, Séparabilité, Concision, Lisibilité. En revanche, elles ne sont pas inversibles et leur calcul à partir du signal, quand il est possible, peut être complexe.

2.1.2 Représentations par paquets de descripteurs

Les représentations sémantiques ont l'inconvénient de nécessiter la mise en place d'un contexte ontologique complexe et sont difficilement généralisables à toutes les scènes sonores. Une catégorie plus simple de représentations est définie par l'extraction de descripteurs à partir du signal. Ceux-ci (généralement de dimension très inférieure) décrivent des propriétés du signal (harmonicité, fréquence fondamentale, répartition spectrale etc..).

Parmi les tâches qui composent le problème de l'archivage de scène sonores, la classification de celles-ci et plus généralement l'extraction d'information (MIR) incite à choisir ce type de représenta-

tions par paquets de descripteurs. De même que les représentations sémantiques, celles-ci ne permettent (généralement) qu'une phase d'analyse et ne sont pas inversibles (en dehors de quelques tentatives notamment celle de ELLIS [2]). En revanche, leur calcul est simple. Un ensemble de K descripteurs d'un signal x tient lieu de représentation, chacun de ces descripteurs d_k peut vivre dans un espace spécifique D_k :

$$\mathcal{X} \to D_1 \times D_2 \times .. \times D_K$$
$$x \to (d_1, d_2, .., d_K)$$

En MIR, de très nombreux descripteurs ont été proposés. La norme MPEG 7 - Audio (ISO/IEC 15938-4 :2002) propose une classification des descripteurs audio en 6 groupes, plus un descripteur de silence. Il est possible de simplifier cette nomenclature et distinguer trois grand ensembles :

- Les descripteurs temporels : Ces descripteurs sont calculés directement sur la forme d'onde et sur ses moments statistiques. Par exemple, le taux de passage par zéro par trame d'analyse qui donne une indication de la fréquence dominante d'un signal et de sa stabilité harmonique. La détection d'évènements très courts (p. ex. les attaques de notes [BDA+05]) ou de silence, peut également se faire dans le domaine temporel en analysant, par exemple, la variation de l'énergie au cours du temps.
- Les descripteurs fréquentiels ou spectraux : Principalement calculés sur une transformée temps-fréquence de type TFCT ou CQT (voir 2.1.3) et sur les moments statistiques spectraux (moyenne, asymétrie, etc..). Les vecteurs de chroma sont particulièrement utilisés dans le monde de la recherche d'information musicale [PLR02, Got03, JE09, RR09]. Les informations de hauteurs de notes (*pitch*), de fréquence fondamentale et d'amplitudes de partiels lorsqu'un mode de production harmonique est supposé. Les caractéristiques des filtres (*p. ex.* les coefficients *Linear Predictive Coding* (LPC)) lorsqu'un modèle de production de type source/filtre est adopté.
- Les descripteurs timbraux : le timbre est une notion difficile à définir, cette catégorie regroupe les descripteurs de la dynamique spectrale d'un signal. On y retrouve les descripteurs cepstraux de type *Mel-Frequency Cepstrum Coefficients* (MFCC), omniprésents en traitement de la parole, mais également le flux spectral, utilisé pour la détection d'attaques [BDA+05] et d'autres descripteurs de la variation spectrale au cours du temps.

Ce type de représentation est souvent un préalable à une tâche d'apprentissage statistique, inenvisageable sur les signaux bruts en raison de leurs dimensions. De plus, le passage vers les espaces de descripteurs peut faciliter grandement l'analyse d'une scène. La Figure 2.1.5 présente le taux de franchissement en zéro, calculé sur des fenêtres glissantes de 30 ms, de nos différents exemples de scènes sonores. On remarque que les signaux harmoniques de la première scène présentent une forte stabilité de ce descripteur. Le signal de parole présente une alternance de valeurs basses et de pics, correspondant à l'enchaînement de phonèmes voisés et non voisés. Les signaux de musique et de bruit donnent des valeurs moins lisibles, avec relativement peu de variations.

Il faut distinguer la vision "sac de descripteurs" qui consiste à représenter un signal par une collection de descripteurs, dans un but d'apprentissage non-supervisé, et les travaux qui construisent des modèles dynamiques des signaux à partir de ces descripteurs. Pionniers des méthodes "sac de descripteurs", SCHEIRER *et al* [SS97] proposent ainsi un assemblage de descripteurs pour une tâche de discrimination voix / musique. Une étude des différents descripteurs et de leur utilité dans ce type de

2. http ://labrosa.ee.columbia.edu/matlab/rastamat/

tâche d'apprentissage est effectuée par PEETERS [Pee03]. Un état de l'art plus complet ainsi qu'une discussion plus poussée des différentes approches se trouve dans la thèse de S. ESSID [Ess05], et plus récemment dans les travaux de JODER *et al* [JER08, JE09], RAMONA *et al* [RR09, RP11B] et FOUCARD *et al* [FELR11]

2.1.3 Représentations par transformée temps-fréquence

Les représentations sémantiques et par paquets de descripteurs sont, *a priori,* non inversibles et constituent donc un moyen d'analyse et d'extraction d'information des scènes sonores, mais pas de les générer. Dans cette partie nous décrivons des représentations adaptées à l'analyse *et* à la synthèse de scènes sonores.

Les signaux audio numériques sont des ondes mécaniques (des sons) échantillonnées (*p. ex.* par le biais d'un convertisseur analogique-numérique après captation par un dispositif de type microphone). La nature ondulatoire des sources (cordes vocales, instruments de musique, ...) fait de l'analyse fréquentielle un outil privilégié de représentation. On a vu plus haut la transformée de Fourier 1.2.1, parfaitement inversible et instrument de base de l'analyse des signaux stationnaires et systèmes linéaires et invariants.

Transformée de Fourier à court terme La transformée de Fourier réalise une projection du signal sur une base de fonctions trigonométriques stationnaires. Pour l'étude de signaux réels, non stationnaires, il devient intéressant d'utiliser des projections localisés dans le plan temps-fréquence. Ces transformations peuvent se voir sous la forme de bancs de filtres de largeur constante. Cette caractéristique permet de les calculer à l'aide d'algorithmes rapides, en particulier la Transformée de Fourier Rapide (TFR ou FFT pour *Fast Fourier Transform*) basée sur l'algorithme de COOLEY-TUKEY de complexité $\mathcal{O}(N \log N)$, où N est la dimension du signal.

La plus fréquemment rencontrée de ces représentations est la transformée de Fourier à court terme (TFCT). Cette transformée réalise localement (pour une trame p) une transformée de Fourier sur une fenêtre d'analyse glissante w de taille L, réelle symétrique (*p. ex.* une fenêtre de Hann) normalisée ($\|w\|_2 = 1$), choisie de telle sorte qu'il existe une transformée inverse pour des signaux x discrets :

$$\mathcal{R}_{TFCT}(x)[p,k] = \frac{1}{\gamma} \sum_{n=0}^{L-1} w[n] \cdot x[n - p\Delta_n] \cdot \exp\left(-j2\pi k \frac{n}{L}\right) \tag{2.1.1}$$

où k est la variable fréquentielle discrète appelée *bin,* Δ_n est un pas d'avancement entre deux trames consécutives *et* γ est un facteur de normalisation. Cette transformée est inversible si elle définit un repère, il suffit de réaliser pour chaque trame une Transformée de Fourier discrète inverse, puis de sommer ces contributions avant de normaliser par la somme des P trames (méthode *Overlap-Add*) :

$$x[n] = \frac{1}{P} \sum_{p=0}^{P} w^*[n - p\Delta_n] \sum_{k=0}^{L-1} \mathcal{R}_{TFCT}(x)[p,k] \exp\left(\frac{j2\pi kn}{L}\right) \tag{2.1.2}$$

où w^* est une fenêtre de synthèse, parfois différente de la fenêtre d'analyse. La reconstruction est parfaite si la condition (*Constant Overlap-Add*) :

$$\forall n, \sum_{p=0}^{P} w[n - p\Delta_n] w^*[n - p\Delta_n] = 1 \tag{2.1.3}$$

est vérifiée.

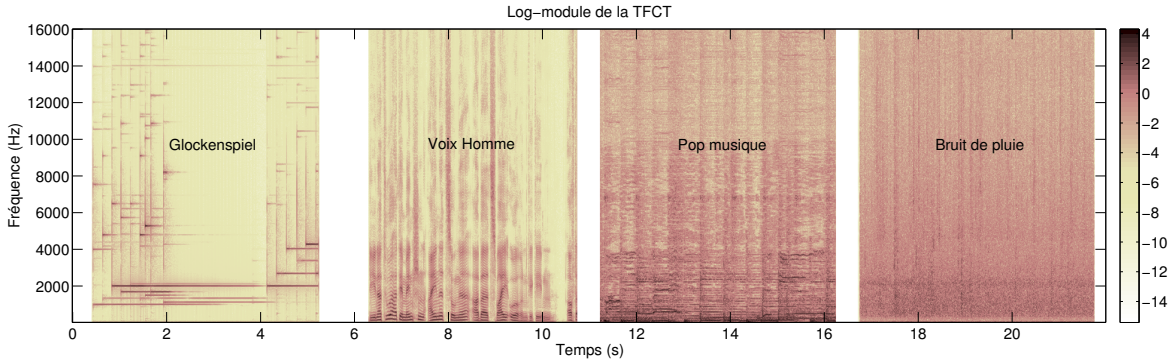

FIGURE 2.1.7: Logarithme du module de la TFCT de 4 signaux audio (glockenspiel, voix homme, musique pop et bruit de pluie), fenêtre de Hann (30 ms) recouvrement de 50%.

En revanche, la précision de la localisation des composantes dans le plan temps-fréquence obéit à un compromis lié au principe d'incertitude d'Heisenberg. La complexité de $\mathcal{R}_{TFCT}$ est dominée par celle d'une TFR. Son calcul nécessite en effet le calcul de P TFRs de taille L, soit $\mathcal{O}(PL \log L)$.

Un certain nombre de transformées temps-fréquence sont disponibles. La TFCT réalise une projection locale sur une échelle de fréquences linéairement espacées $f_k = k\frac{f_s}{L}$ où f_s est la fréquence d'échantillonnage. La Figure 2.1.7 présente le logarithme du module de la TFCT des 4 scènes sonores précédentes. Les fortes valeurs figurent la distribution de l'énergie dans le plan temps fréquence. Pour le signal de Glockenspiel, l'énergie est localisée sur des composantes horizontales (résonances harmoniques) et verticales (attaques transitoires). Pour le signal de voix d'homme, on observe un spectre caractéristique d'une alternance de spectres localement pseudo-harmoniques et de sons bruités, filtrés, correspondant au mode de production de la parole. La troisième scène sonore est plus confuse car plus riche, mais on y distingue toujours des composantes transitoires et harmoniques. Enfin pour le bruit de pluie, en dehors de quelques transitoires (lors d'impacts de gouttes dans un voisinage du microphone), le spectre est uniformément bruité.

Transformée en cosinus discrets Une transformée couramment utilisée pour l'analyse des signaux audio est la Transformée en Cosinus Discrète (DCT), et son avatar appelée *Modified Discrete Cosine Transform* (MDCT)[PB86], que l'on retrouve dans la plupart des standards de codage audio MPEG[MPE99]. La MDCT et ses variantes (*p. ex.* la version complexe présenté par MALVAR [Mal99]), sont basées sur la transformée en cosinus discrète de type IV qui s'écrit pour un signal x de taille L :

$$\mathcal{R}_{DCT-IV}[k] = \sqrt{\frac{2}{M}} \sum_{n=0}^{L-1} x[n] \cdot cos\left[(\frac{1}{2}+n)\cdot(\frac{1}{2}+k)\cdot\frac{\pi}{L}\right] \tag{2.1.4}$$

la version présentée par PRINCEN et BRADLEY [PB86] permet de garantir une reconstruction parfaite à la synthèse grâce à un échantillonnage critique. Cette propriété se comprend plus aisément lorsqu'on reformule la MDCT sous forme matricielle. Soit x un vecteur de taille $N = PK$ composé de P segments de taille K. La MDCT de taille $L = 2K$ s'écrit comme une matrice de transformation $\boldsymbol{T}$ de taille $N \times N$:

$$\boldsymbol{T}[n, pK+k] = \phi_{p,k}[n] \text{ pour } p \in [0..P-1] \text{ , } k \in [0..K-1] \text{ et } n \in [0..N-1] \tag{2.1.5}$$

où :

$$\phi_{p,k}[n] = w_p[u]\sqrt{\frac{2}{K}}\cos\left[\frac{\pi}{K}\left(u+\frac{K+1}{2}\right)\left(k+\frac{1}{2}\right)\right] \tag{2.1.6}$$

avec $u = n-(p-\frac{1}{2})K$ et w_p une fenêtre d'analyse sur la trame p. La condition d'inversion qui permet une reconstruction parfaite se comprend alors comme une condition d'orthogonalité sur la matrice $\boldsymbol{T}$:

$$\boldsymbol{T}\boldsymbol{T}^T = \boldsymbol{I} \tag{2.1.7}$$

il est ensuite aisé de montrer [Rav08] que cette condition est obtenue lorsque la fenêtre choisie vérifie pour $u \in [0..\frac{L}{2}-1]$:

$$w_0[u] = 1 \tag{2.1.8}$$

$$w_p^2[u+\frac{L}{2}] + w_{p+1}^2[u] = 1, p \in [0..P-2] \tag{2.1.9}$$

$$w_{P-1}[u+\frac{L}{2}] = 1 \tag{2.1.10}$$

Les conditions 2.1.8 et 2.1.10 assurent la conservation aux bords du signal. Parmi les fenêtres vérifiant la condition 2.1.9, on trouve la fenêtre de Kaiser-Bessel dérivée [FBD+96] ou la fenêtre sinusoïdale proposée par Malvar [Mal99] :

$$w[n] = \sin\left[\frac{\pi}{L}\left(u+\frac{1}{2}\right)\right] \tag{2.1.11}$$

En pratique, l'utilisation de TFRs permet d'éviter le calcul de la matrice $\boldsymbol{T}$ et du produit matriciel :

$$c = \boldsymbol{T}^T x \tag{2.1.12}$$

donnant le vecteur c de coefficients de la MDCT.

La TFCT et la MDCT sont simples et aisément calculables, elle explicitent l'information fréquentielle instantanée des signaux, mais souffrent de deux handicaps :

- La résolution fréquentielle est contrainte par la taille de la fenêtre d'analyse choisie. De plus cette résolution est constante pour toutes les fréquences.
- La résolution temporelle est contrainte par le pas d'avancement entre deux fenêtres consécutives, en particulier pour la MDCT où ce pas d'avancement est fixé à exactement une demi-fenêtre pour assurer une inversion parfaite.

Transformée à Q constant Pour certaines applications (*p.ex.* musicales), des échelles fréquentielles logarithmiques peuvent être préférables. Parmi ces représentations, la transformée à Q constant (CQT) ajuste la résolution fréquentielle en utilisant des fenêtres de taille différente pour chaque bin fréquentiel :

$$\mathcal{R}_{CQT}[k] = \frac{1}{L_k}\sum_{n=0}^{L_k-1} w_k[n]\cdot x[n]e^{-j2\pi n\frac{f_k}{f_s}} \tag{2.1.13}$$

où w_k est une fenêtre d'analyse de longueur L_k propre à chaque fréquence. Celles-ci sont géométriquement espacés sur une échelle tempérée définie par la fréquence f_{min} minimale correspondant au bin k et à une constante B définissant le nombre de bins par octave (typiquement $B = 12, 24, 36..$) :

$$f_k = (2^{1/B})^k \cdot f_{min} \tag{2.1.14}$$

La Figure 2.1.8 présente le logarithme du module de la CQT des scènes sonores précédentes. L'échelle des fréquences est ici l'échelle de hauteur de la norme MIDI. Cette dernière établit une correspondance

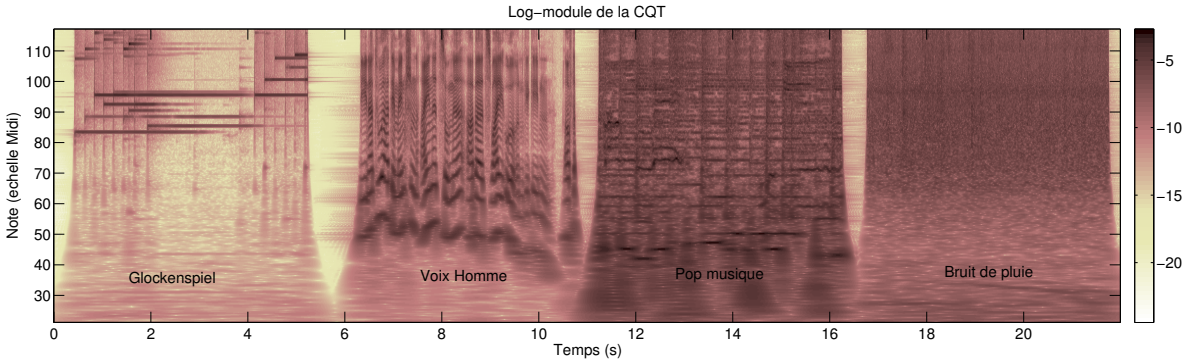

FIGURE 2.1.8: Logarithme du module de la CQT de 4 signaux audio (glockenspiel, voix homme, musique pop et bruit de pluie), 8 octaves, 36 bins par octave .

entre l'échelle logarithmique des fréquence et la gamme tempérée où la référence est donnée par le La du diapason à 440 Hz, correspondant à l'indice MIDI 69. La conversion s'opère avec :

$$p = 69 + 12\log_2\left(\frac{f}{440}\right) \tag{2.1.15}$$

ce qui garantit qu'un intervalle de fréquence définissant une octave ($f_1$et $f_2 = 2f_1$) est représentés par 12 hauteurs MIDI, correspondant aux 12 notes de la gamme tempérée.

En général, on peut définir n'importe quelle transformation temps-fréquence basée sur une transformée de Fourier (voir l'étude de FILLON et PRADO[FP12]) par :

$$\forall x \in \mathbb{R}^N, \mathcal{R}_{TF}(x)[p,k] = \frac{1}{\gamma_k}\sum_{n=0}^{N-1} w_{p,k}[n]\cdot x[n]\cdot e^{-j2\pi n\frac{f_k}{f_s}} \tag{2.1.16}$$

où la fenêtre d'analyse $w_{p,k}$ et le facteur de normalisation γ_k sont spécifiques à chaque bin fréquentiel.

2.1.4 Repères

La théorie des repères permet de faire le lien entre les représentations temps-fréquences présentées ci-dessus et les représentations parcimonieuses en introduisant le concept de redondance. Le lecteur intéressé pourra se référer à [Mal09]. Soit $\mathcal{H}$ un espace de Hilbert et $\{\phi_i\}_{i\in I}$ une famille de vecteurs de $\mathcal{H}$ (finie ou infinie), on dit que $\{\phi_i\}_{i\in I}$ est un repère de $\mathcal{H}$ s'il existe deux constantes réelles $B \geq A > 0$ telles que

$$\forall x \in \mathcal{H},\ A\|x\|^2 \leq \sum_{i\in I} |\langle x, \phi_i\rangle|^2 \leq B\|x\|^2 \tag{2.1.17}$$

lorsque $A = B$ on parle de repère ajusté (*tight frame*). La famille $\{\phi_i\}_{i\in I}$ décrit un opérateur Φ tel que :

$$\forall i \in I, \forall x \in \mathcal{H}, \Phi x_i = \langle x, \phi_i\rangle \tag{2.1.18}$$

Un cas particulier de repère ajusté est une base orthonormale. En dimension finie N une base orthonormale est définie par la matrice carrée Φ dont les colonnes $\{\phi_i\}_{i\in[0..N-1]}$ sont deux à deux orthogonales : $\forall i,j\ \langle\phi_j,\phi_i\rangle = \delta_{ij}$ où δ_{ij} est le symbole de Kronecker. Une représentation dans une base orthonormale est unique et s'écrit :

$$\mathcal{R}(x) = c = \Phi x \tag{2.1.19}$$

Cette représentation est inversible, les coefficients d'analyse peuvent être utilisés directement à la synthèse :

$$x = \mathcal{R}^{\dagger}(c) = \Phi^T c \tag{2.1.20}$$

Dans le cas général en revanche ($A \neq B$), il n'est pas possible d'utiliser les vecteurs d'analyse directement pour la synthèse. L'opérateur adjoint Φ^* de Φ est défini sur l'ensemble des vecteurs d'énergie finie $\ell^2(I)$:

$$\forall x \in \mathcal{H}, \forall c \in \ell^2(I), \langle \Phi^* c, x \rangle = \langle c, \Phi x \rangle = \sum_{i \in I} c_i \langle x, \phi_i \rangle^* \tag{2.1.21}$$

et permet la synthèse à partir des coefficients d'analyse :

$$\Phi^* \Phi x = \sum_{i \in I} \langle x, \phi_i \rangle \phi_i \tag{2.1.22}$$

L'inversibilité de l'opérateur dépend de la nature du repère. Si les vecteurs de la famille sont linéairement indépendants (on parle alors d'une base de Riesz), le repère dual est une base biorthogonale. Dans le cas contraire, et en l'absence d'hypothèse supplémentaire, la reconstruction s'effectue à partir des coefficients d'analyse à l'aide d'une pseudo-inversion de Moore-Penrose.

Redondance Si les vecteurs du repère sont normalisés ($\forall i \in I, \|\phi_i\|_2 = 1$) la redondance est définie par le rapport entre le nombre de vecteurs $P = card(I)$ et la dimension N. Le Théorème 5.2 [Mal09] stipule de plus :

$$A \leq \frac{P}{N} \leq B \tag{2.1.23}$$

On montre ainsi par exemple que l'union de m bases orthonormales forme un repère dont la redondance est m.

Repères de Fourier Dans le cas de la MDCT, Φ est un repère ajusté définissant une base orthonormale ce qui rend l'inversion très simple. Dans le cas général d'une représentation temps fréquence, l'équation (**??**) permet de réaliser la synthèse.

Il est possible de construire un repère de Fourier discret (ou repère de Gabor) à partir d'une base de Fourier discrète $\{e^{j2\pi nk/L}\}_{0 \leq k < L} \in \mathbb{C}^L$. Pour cela on peut utiliser une fenêtre w symétrique (de même qu'en (2.1.16)) et un décalage Δ_n tel que $N = P\Delta_n$. La collection de vecteurs $\{\phi_{p,k}\}_{(p,k) \in \mathbb{Z}^2}$ de la forme :

$$\phi_{p,k} = w[n - p\Delta_n] \exp\left(\frac{j2\pi kn}{L}\right) \tag{2.1.24}$$

est un repère de Fourier ajusté à la condition ([Mal09], Théorème 5.18) :

$$\forall 0 \leq n \leq N,\ L \sum_{p=0}^{P-1} |w[n - p\Delta_n]|^2 = A > 0 \tag{2.1.25}$$

La collection $\{\phi_{p,k}\}_{(p,k) \in \mathbb{Z}^2}$ réalise une couverture du plan temps-fréquence dont le maillage régulier est déterminé par le couple de pas temps-fréquence (Δ_n, Δ_k). On peut alors, par exemple, interpréter la transformée de Fourier à court terme sous forme de produits scalaires :

$$\mathcal{R}_{TFCT}[p,k](x) = \langle x, \phi_{p,k} \rangle \tag{2.1.26}$$

De la même façon on peut envisager les transformées temps-fréquence décrites ci dessus, sous la forme de produit scalaires dans des repères ajustés (avec quelques restrictions pour la CQT [Bro91]).

Dans la suite, nous considèrerons des dictionnaires sous la forme de repères. La famille de vecteurs $\{\phi_i\}_{i\in I}$ sera choisie complète et redondante. Les repères ajustés définis par les transformées temps-fréquence de type Fourier offrent le grand avantage d'avoir un repère dual identique, ce qui rend l'inversion très simple. En revanche, ces repères ne garantissent pas toujours la concision des représentations. Nous allons considérer des repères redondants, dans lesquels on espère voir apparaître une propriété essentielle : la parcimonie.

Objectif	Représentations		
	Sémantiques	Descripteurs	Temps-Fréquence
Fidélité	non-inversible	non-inversible	inversible
Comparabilité	oui	possible	possible
Séparabilité	hypothétique	possible	possible
Concision	oui	oui	non
Simplicité	non	oui	oui
Lisibilité	haut-niveau	intermédiaire	bas niveau

TABLE 2.1.1: Récapitulatif de l'adéquation entre les objectifs et les propriétés des représentations temps-fréquence

2.2 Représentations Parcimonieuses

La grande majorité des signaux naturels complexes (en particulier les scènes sonores) ne peuvent dans le cas général être représentés à l'aide d'un nombre réduit de coefficients dans une seule base. Pour obtenir des représentations exhibant cette propriété de parcimonie, il est nécessaire d'envisager des dictionnaires redondants. Ce paradigme est à la base du champ de recherche sur les représentations parcimonieuses.

Une étude exhaustive de l'état de l'art sur les représentations parcimonieuses se trouve dans le livre de S. MALLAT [Mal09]. Dans cette partie, nous éluderons la formulation continue des atomes et présenterons directement les formules discrètes. Nous nous concentrerons sur les applications et travaux récents.

2.2.1 Formulation

Représentations exactes Dans le paradigme des représentations parcimonieuses, on cherche une représentation d'un signal x sous la forme d'une combinaison linéaire des éléments appelés *atomes* d'un ensemble appelé *dictionnaire :*

$$x = \sum_{i=0}^{M-1} \alpha_i d_i \tag{2.2.1}$$

où M est le nombre d'éléments du dictionnaire $\mathcal{D} = \{d_i\}_{i=0..M-1}$. Pondérés par l'ensemble α de coefficients α_i, $i = [0..M-1]$, ces éléments constituent la représentation de x. Une formulation du problème de parcimonie sous contrainte de reconstruction (1.1.4) est alors :

$$\min_{\alpha} \mathcal{P}(\alpha)\,, \text{ soumis à } x = \sum_{i=0}^{M-1} \alpha_i d_i \tag{2.2.2}$$

où $\mathcal{P}(\alpha)$ est une fonctionnelle de parcimonie sur les coefficients α_i. Couramment, on utilise une norme du type ℓ_p pour mesurer la parcimonie d'un vecteur :

$$\|\alpha\|_p = \left(\sum_{i=0}^{M-1} |\alpha_i|^p \right)^{\frac{1}{p}} \tag{2.2.3}$$

mais d'autres mesures sont envisageables (*p. ex.* l'entropie [CW92]). On définit ainsi une classe de problèmes :

$$(P_p) : \min_{\alpha} \|\alpha\|_p\,, \text{ soumis à } x = \sum_{i=0}^{M-1} \alpha_i d_i \tag{2.2.4}$$

Le problème (P_0) utilise la pseudo-norme ℓ_0 qui réalise un simple comptage des coefficients non nuls. Ce problème est combinatoire ; il formule une recherche de la plus petite combinaison linéaire d'éléments d'un dictionnaire qui permet de reconstruire exactement x :

$$(P_0) : \min_{\alpha} \|\alpha\|_0\,, \text{ soumis à } x = \sum_{i=0}^{M-1} \alpha_i d_i \tag{2.2.5}$$

or la seule façon de résoudre (P_0) dans le cas général (*c-à-d.* sans contraintes additionnelles sur le dictionnaire ou le signal) est de réaliser une recherche exhaustive des combinaisons d'atomes du dictionnaire.

Représentations approchées La classe de problèmes (P_p) nécessite une reconstruction parfaite du signal. On peut définir une classe de problèmes qui cherche une représentation de x sous la forme d'une approximation :

$$x \approx \sum_{i=0}^{M-1} \alpha_i d_i \tag{2.2.6}$$

On cherche alors la minimisation duale de la fonctionnelle de parcimonie sous contrainte de reconstruction (2.2.7) ou alternativement la formulation pénalisée (2.2.8) :

$$(P_p^{\epsilon}) : \min_{\alpha} \|\alpha\|_p \text{ , soumis à } \|x - \sum_{i=0}^{M-1} \alpha_i d_i\|_2^2 \leq \epsilon \tag{2.2.7}$$

$$(L_p^{\lambda}) : \min_{\alpha} \left(\|x - \sum_{i=0}^{M-1} \alpha_i d_i\|_2^2 + \lambda \|\alpha\|_p \right) \tag{2.2.8}$$

L'équivalence $P_p^{\epsilon=0} \Longleftrightarrow P_p$ est évidente. Le coefficient λ permet de régler le compromis entre parcimonie de la synthèse et fidélité de la reconstruction. La formulation (L_p^{λ}) fait apparaître la fonctionnelle $\mathcal{L}(\alpha, \lambda) = \|x - \sum_{i=0}^{M-1} \alpha_i d_i\|_2^2 + \lambda \|\alpha\|_p$ dont la minimisation (dans le cas convexe $p > 0$) procède de la méthode du multiplicateur de Lagrange par annulation du gradient $\nabla \mathcal{L}(\alpha, \lambda)$.

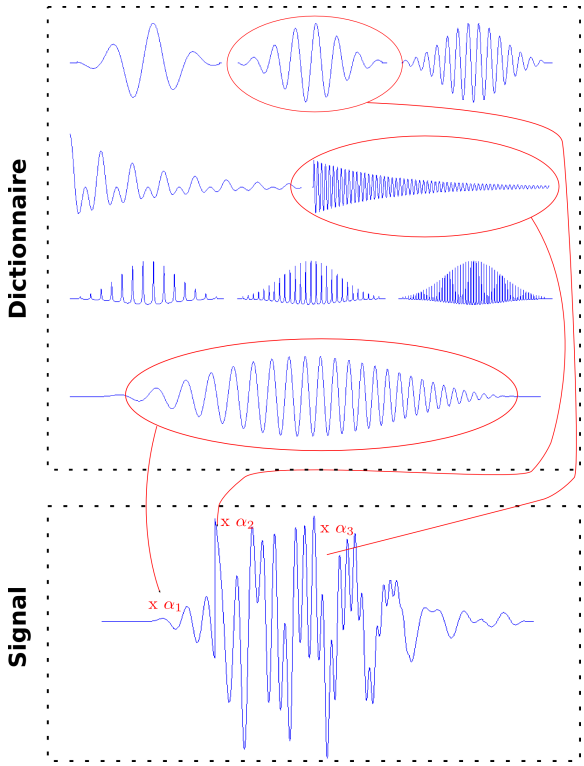

FIGURE 2.2.1: Dictionnaire constitué d'un ensemble de formes d'ondes. Le signal temporel en exemple a une représentation parcimonieuse sous la forme d'une combinaison linéaire de trois éléments du dictionnaire.

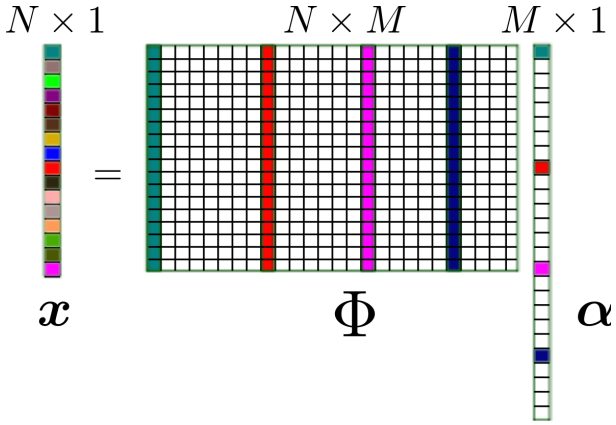

FIGURE 2.2.2: x a une représentation parcimonieuse α dans le dictionnaire Φ redondant ($M > N$).

Formulations matricielles Dans le cas de signaux complexes de dimension N finie, on utilisera plutôt une écriture matricielle, soit $\boldsymbol{\Phi} \in \mathbb{C}^{N \times M}$ la matrice dictionnaire, une représentation $\boldsymbol{\alpha} \in \mathbb{C}^M$ de $\boldsymbol{x} \in \mathbb{C}^N$ dans $\boldsymbol{\Phi}$ s'écrit :

$$\boldsymbol{x} = \boldsymbol{\Phi} \cdot \boldsymbol{\alpha} \tag{2.2.9}$$

Il est alors possible de reformuler (2.2.4), (2.2.7) et (2.2.8) :

$$(\mathbf{P}_p) : \min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_p \text{ , soumis à } \boldsymbol{x} = \boldsymbol{\Phi} \cdot \boldsymbol{\alpha} \tag{2.2.10}$$

$$(\mathbf{P}_p^{\epsilon}) : \min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_p \text{ , soumis à } \|\boldsymbol{x} - \boldsymbol{\Phi} \cdot \boldsymbol{\alpha}\|_2^2 \leq \epsilon \tag{2.2.11}$$

$$(\mathbf{L}_p^{\lambda}) : \min_{\boldsymbol{\alpha}} \left(\|\boldsymbol{x} - \boldsymbol{\Phi} \cdot \boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_p\right) \tag{2.2.12}$$

Ces problèmes sont particulièrement intéressants dans le cas sous-determiné $N < M$. Dans le cas $p = 0$, ces problèmes sont combinatoires. De nombreux travaux proposent une résolution sous-optimale de ce problème (voir notamment les travaux de Dymarski *et al* [DMV90]). Ces recherches ont donné naissance aux algorithmes dit *gloutons* que nous présentons plus en détail au chapitre suivant.

Une alternative est de choisir p de façon à hériter d'un problème convexe, pour lequel il existe des méthodes établies pour trouver l'optimum.

Le cas de figure $p = 2$ est une forme particulière de régularisation de Tikhonov. Une solution analytique $\boldsymbol{\alpha}^*$ est dérivable et s'écrit sous forme matricielle :

$$\boldsymbol{\alpha}^* = \boldsymbol{\Phi}^{\dagger} \boldsymbol{x}$$

où $\boldsymbol{\Phi}^{\dagger} = \boldsymbol{\Phi}^T (\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}$ est le pseudo-inverse de Moore-Penrose. Cette solution est celle qui minimise l'énergie de la solution, néanmoins ce critère énergétique n'est pas une garantie de parcimonie sur la solution.

Le cas de figure $p = 1$ est une alternative très étudiée car la fonctionnelle ℓ_1 présente à la fois de bonnes propriété de parcimonie et un profil convexe. Il a donné lieu notamment aux approches LASSO[Tib94] et *Basis Pursuit* [CDS98]. La norme ℓ_1 est en effet une mesure qui favorise la parcimonie tout en garantissant la convexité du problème. En revanche, il n'existe en général pas de forme analytique simple de la solution.

2.2.2 Apprentissage de dictionnaire

Construire le dictionnaire à partir d'exemples de signaux est une idée raisonnable. La communauté du traitement de la parole (et en particulier du codage [DMV90]) a été la première à envisager les possibilités d'apprentissage de dictionnaires de phonèmes sur des données annotées [SW96]. Cette idée se retrouve également pour des images [BM95], pour le codage de stimuli visuels dans le cortex [OF96, LS00], et plus récemment pour la musique avec l'apprentissage de molécules harmoniques [LVRD08, GB03].

Plus généralement, l'apprentissage de dictionnaire peut être considéré comme un champ spécifique de recherche s'attaquant au problème suivant (sous forme matricielle) : Soit $\boldsymbol{X} = [\boldsymbol{x}^1, ..., \boldsymbol{x}^k] \in \mathbb{C}^{N \times K}$ un ensemble de K signaux de dimension N, on cherche une matrice $\boldsymbol{\Phi} \in \mathbb{C}^{N \times M}$ d'éléments ϕ_i ainsi qu'une matrice $\boldsymbol{A} = [\boldsymbol{\alpha}^1, ..., \boldsymbol{\alpha}^k] \in \mathbb{C}^{M \times K}$ telles que $\boldsymbol{X} \approx \boldsymbol{\Phi} \cdot \boldsymbol{A}$. La forme pénalisée de ce problème est :

$$(DL_p^{\lambda}) : \min_{\boldsymbol{\Phi}, \boldsymbol{A}} \frac{1}{K} \sum_{i=1}^{K} \left(\|\boldsymbol{x}^i - \boldsymbol{\Phi} \cdot \boldsymbol{\alpha}^i\|_2^2 + \lambda \|\boldsymbol{\alpha}^i\|_p\right) \tag{2.2.13}$$

Des méthodes récentes [MBP+08, RZE10, JM10] proposent des solutions à ce type de problèmes avec des contraintes supplémentaire sur la forme et la structure des dictionnaires recherchés. Ces méthodes font également le lien avec des techniques de factorisations matricielle de type *Analyse en Composantes Principales* (PCA).

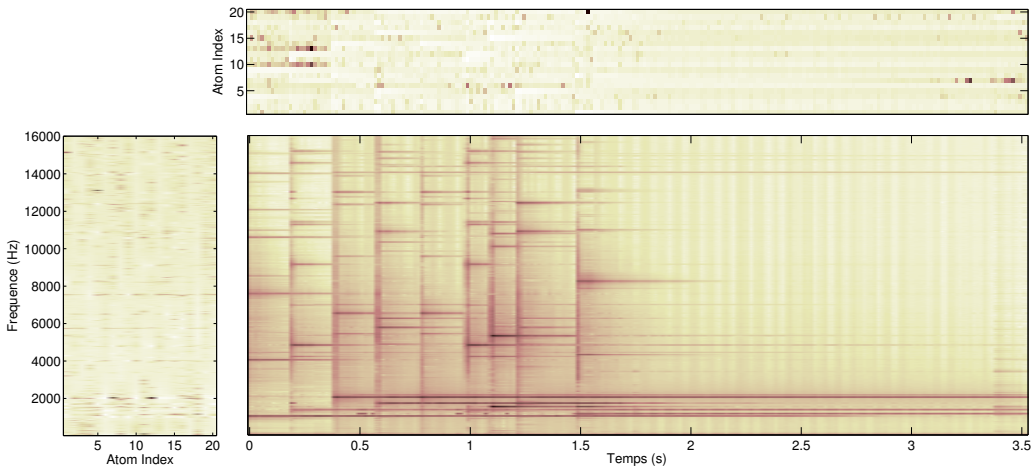

FIGURE 2.2.3: Exemple de factorisation en Matrices non négatives (utilisant la divergence d'Itakura-Saito). Le Spectrogramme d'un signal de Glockenspiel est approché par le produit de deux matrices W (dictionnaire) et H (activations).

Factorisations matricielles Il convient également ici de citer d'autres approches de réduction de dimension par factorisation matricielle. En particulier pour des données non-négatives telles que le module de la TFCT. La factorisation en matrice non-négatives (NMF [LS01]) est une technique prisée permettant d'approcher une matrice X (par exemple un spectrogramme, soit le module au carré de la TFCT) par un produit WH où W est une matrice de spectres élémentaires appelée dictionnaire et H contient les activations au cours du temps de ces spectres de base. Dans la version originale, à la fois W et H sont apprises à partir des signaux. De nombreux travaux récents de traitement de signaux audio sont basés sur ce type de décompositions (*p. ex.* pour la transcription automatique [FBD09, HBD11a, FLBR12], la séparation de sources [LBF11, FBR12] etc..). La Figure 2.2.3 présente un exemple d'une telle factorisation.

La plupart des méthodes de l'état de l'art résolvent ce problème en alternant des phases d'optimisation du dictionnaire et de la représentation. Un article récent de RAKOTOMAMONJY [Rak12] propose une optimisation directe basée sur le calcul de gradient proximaux.

Un grand avantage des dictionnaires appris, c'est qu'une information additionnelle peut être attachée aux atomes pourvu que l'échantillon d'apprentissage soit annoté. Cette information peut ensuite être utilisée pour de la reconnaissance ou de la classification[RBPU08, LKCC07, HA06] ou pour la reconstruction de parties manquantes (*Inpainting*) [MLA10, AEJ+12]. En revanche, la classe de problème définie par (DL_p^λ) est en général, encore plus complexe à résoudre, au sens où de nombreux minima locaux apparaissent.

2.2.3 Dictionnaires structurés

Dictionnaires basés sur la transformée de Fourier La plupart des transformées temps-fréquence décrites plus haut peuvent se comprendre comme des projections dans des dictionnaires structurés, et en particulier des repères ajustés étudiés en 2.1.4 page 25. Chaque atome de ces dictionnaires est une translation en temps et en fréquence d'un atome de base w, avec éventuellement une dilatation (dans le cas des ondelettes).

D'une façon générale, on définit un atome de Gabor discret de taille L centré en temps sur $p\Delta_n$

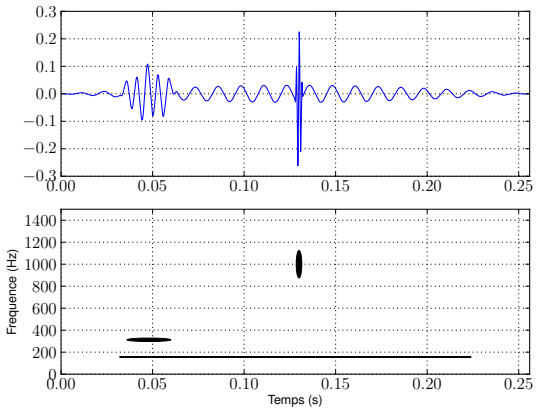

FIGURE 2.2.4: Visualisations temporelles et dans le plan temps-fréquence de 3 atomes MDCT d'échelles différentes.

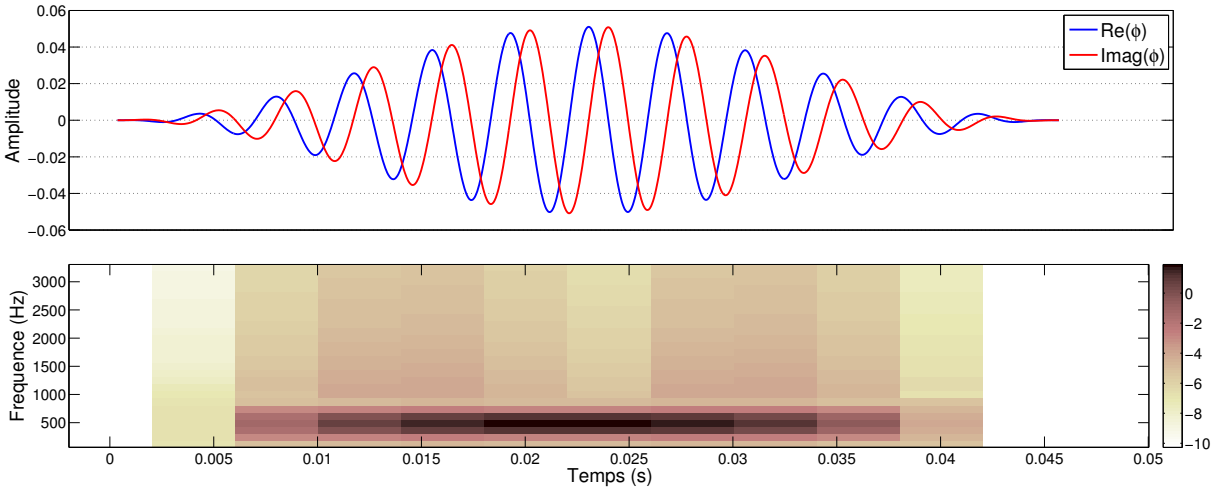

FIGURE 2.2.5: Haut : Partie réelles et imaginaires d'un atome de Gabor complexe ϕ. Bas : Logarithme du module de la TFCT.

et en fréquence sur k par :

$$\phi^{Gabor}_{L,p,k}[n] = w\left[\frac{n - p\Delta_n}{L}\right] \cdot \exp(\frac{2i\pi kn}{L}) \tag{2.2.14}$$

avec w une fenêtre réelle et symétrique (*p. ex.* une fenêtre de Hann comme illustré Figure 2.2.5) et normalisée ($\|\phi_{L,p,k}\|_2 = 1$). On retrouve le k-ième bin de la p-ième trame de la TFCT comme le produit scalaire hermitien de x avec l'atome $\phi_{L,p,k}$:

$$\langle x, \phi_{L,p,k}\rangle = \sum_{n=0}^{N-1} x[n] w\left[\frac{n - p\Delta_n}{L}\right] \exp(\frac{-2i\pi kn}{L}) \tag{2.2.15}$$

où l'on retrouve la même chose qu'en (2.1.1). De la même façon, on peut définir un atome de MDCT de taille L par :

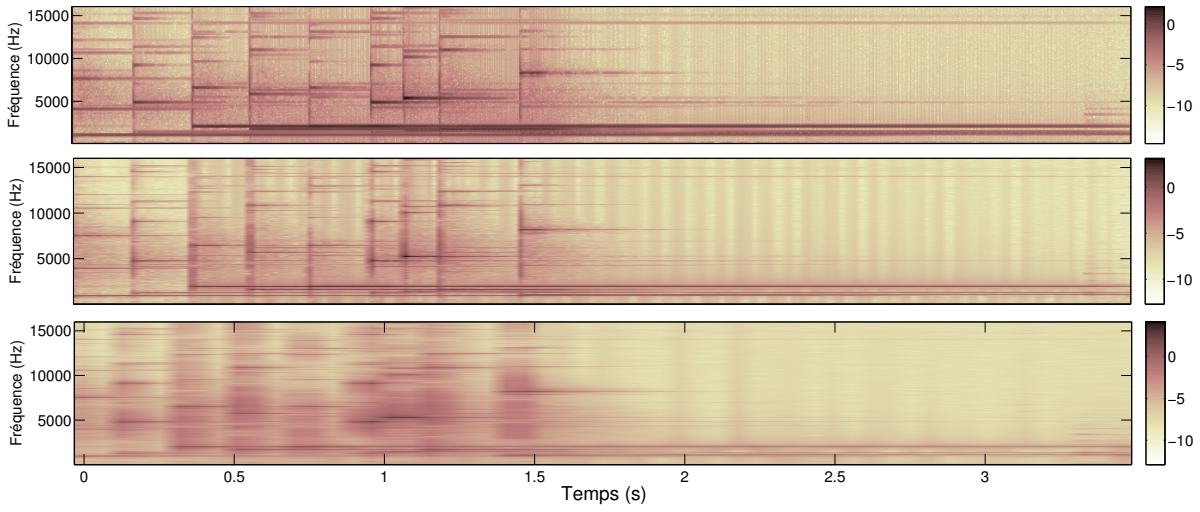

FIGURE 2.2.6: Logarithme du module de la TFCT pour 3 échelles différentes (de haut en bas : fenêtre de Hann de 8, 32 et 128ms, recouvrement de 50%) sur un signal de glockenspiel. La résolution temps-fréquence obéit à un compromis.

$$\phi_{L,p,k}^{MDCT}[n] = w_L[u]\sqrt{\frac{2}{L}}\cos\left[\frac{\pi}{L}\left(u+\frac{L+1}{2}\right)\left(k+\frac{1}{2}\right)\right] \tag{2.2.16}$$

avec $u = n - pL - T$ (T est une variable permettant d'aligner plusieurs échelles MDCT voir [RRD08]). On a vu que les projections dans ces repères ne garantissent pas une parcimonie des représentations. On peut en revanche, aisément construire des dictionnaires redondants en concaténant plusieurs repères de Fourier ajustés de type Gabor ou MDCT.

Union de bases orthonormales Un bon moyen d'obtenir des dictionnaires structurés redondants est la concaténation de bases orthonormales de type MDCT d'échelles différentes. Un tel dictionnaire s'écrit :

$$\Phi = \bigcup_{s=1}^{S} \Phi_s \tag{2.2.17}$$

où Φ_s est un dictionnaire temps-fréquence d'échelle s. La redondance introduite par cette concaténation va favoriser la parcimonie des représentations. Les différentes échelles sont adaptées à la variété des composantes de scènes sonores (longues échelles pour les composantes harmoniques, courtes échelles pour les composantes transitoires comme illustré Figure 2.2.6).

Ondelettes, Gammachirps, ridgelets, curvelets, etc.. Dans sa thèse [Lev07], LEVEAU propose un bestiaire des différents atomes qu'il est possible d'envisager. On trouvera également dans [Mal09] une description complète des dictionnaires structurés et de leurs applications en traitement du signal.

2.2.4 Parcimonie et sous-échantillonnage

Récemment, CANDÈS *et al* [CRT06] et DONOHO *et al* [DET06, Don06] ont mis à jour un pan entier de cas de figure pour lesquels la solution du problème (P_1^0) est presque sûrement celle du problème (P_0^0). Ce paradigme est connu sous le nom de *Compressed Sensing* (CS). Il s'attaque au problème de

reconstruction des signaux parcimonieux dans une base orthonormale $\Psi \in \mathbb{R}^{M\times M}$, sous échantillonnés à travers une matrice Φ, le produit des deux matrices $D = \Phi\Psi$ constituant un dictionnaire redondant. Soit $y \in \mathbb{R}^M$ un vecteur parcimonieux dans $\Psi(\exists c, y = \Psi c$ avec $\|c\|_0 = m \ll M)$ échantillonné à l'aide d'une matrice $\Phi \in \mathbb{R}^{N\times M}$ (avec $N < M$) pour obtenir $x \in \mathbb{R}^N$:

$$x = \Phi y = Dc \tag{2.2.18}$$

Retrouver c à partir de x s'apparente au problème (P_0^0) en l'absence de bruit et à (P_0^ϵ) dans le cas bruité ($x \approx \Phi y$). Ce problème est connu sous le nom de recouvrement parcimonieux (de l'anglais *recovery*).

Donoho et Tanner [DT10] ont explicité en fonction de la nature du dictionnaire D (*p.ex.* une base de Fourier incomplète [CRT06], dictionnaire aléatoire [TG07]) et des rapports $\rho = m/M$ et $\delta = N/M$, des diagrammes de transition de phase. Ces derniers servent de marqueurs théoriques sur la probabilité de succès du recouvrement parcimonieux par les algorithmes d'optimisation convexe de type LASSO. Ce champ de recherche étant très actif, de nouvelles avancées tant théoriques que pratiques et algorithmiques apparaissent en permanence. La grande majorité de ces méthodes reposent sur une relaxation particulière du problème P_0^0.

De très nombreuses applications du CS sont apparues, par exemple en acoustique [CLD11], en codage distribué de signaux multi-capteurs [BDW+05, ZMW+10], en imagerie médicale et satellitaire [KMS+12], et même pour le calcul rapide de transformées de Fourier [Iwe10, HIK12].

2.2.5 Parcimonie à l'analyse

Très récemment, Nam *et al* [NDEG12] se sont intéressés au problème dual de parcimonie à l'analyse. Le problème (P_0) est en effet une formulation d'un problème de synthèse : trouver la combinaison la plus simple d'atomes du dictionnaire qui permet de synthétiser un signal.

Soit $\Omega \in \mathbb{R}^{M\times N}$ une transformation ou un opérateur d'analyse appliqué au signal x avec typiquement $M \geq N$. Nam définit la co-parcimonie comme la mesure :

$$\ell := M - \|\Omega x\|_0 \tag{2.2.19}$$

Ce qui l'amène à proposer une régularisation différente de P_0 , où cette fois-ci la parcimonie est recherchée à travers l'opérateur d'analyse.

$$(P_0^\ell) : \min_x \|\Omega x\|_0 \text{ , soumis à } y = Mx \tag{2.2.20}$$

où $M \in \mathbb{R}^{m\times N}$ est une matrice de mesure avec $m < N$ et y un vecteur de mesure. Résoudre ce problème revient à maximiser la co-parcimonie, le paradigme est ainsi inversé : au lieu de chercher le plus petit sous-espace contenant (approximativement) l'information d'un signal, on cherche le plus grand sous-espace orthogonal à cette information. Cette formulation originale s'avère intéressante dans un certain nombre de cas pratiques de problèmes inverses et de résolution de systèmes d'équations aux dérivées partielles sous-déterminés. Pour résoudre P_0^ℓ, Nam propose deux algorithmes, l'un basé sur une relaxation du problème avec une norme ℓ_1, et un algorithme glouton (GAP pour *Greedy Analysis Pursuit*). Des travaux récents [GE12] présentent des adaptations d'algorithmes de recouvrement (*CoSaMP* et *Subspace Pursuit*) à ce formalisme.

2.3 Algorithmes

On peut regrouper les algorithmes s'attaquant au problème (P_0) en trois grandes catégories, les algorithmes basés sur une relaxation de la contrainte, les algorithmes basés sur un modèle Bayésien et enfin les algorithmes itératifs ou de poursuite. La dernière de ces catégories sera décrite plus en détail dans le chapitre suivant.

2.3.1 Algorithmes basés sur une relaxation

Le relâchement de (P_0) en (P_p) avec $1 \leq p$ rend le problème convexe. Pour le cas de figure $M > N$, (P_2) est connu sous le nom de régularisation de Tikhonov (ou *Ridge regression*).

Le cas de figure $p = 1$ est une alternative très étudiée. Il a donné lieu notamment aux approches *Least Absolute Shrinkage and Selection Operator* (LASSO)[Tib94], Basis Pursuit [CDS98] ou encore *Focal Underdetermined System Solver* (FOCUSS [GR97]). On trouvera également des variantes plus ou moins complexes, telles que le *Re-Weighted Least Squares* et sa version itérative proposée par Daubechies [DDFG10]. La norme ℓ_1 est en effet une mesure qui favorise la parcimonie tout en garantissant la convexité du problème. En revanche, il n'existe en général pas de forme analytique simple de la solution, ces algorithmes (et leurs nombreuses variantes) sont donc basés sur des méthodes du type descente de gradients et méthodes du point intérieur.

Pour le cas des images, la norme de variation totale est également utilisée depuis une quinzaine d'années [CL97]. On notera également que pour résoudre le problème de co-parcimonie (P_0^ℓ)(2.2.20), il est possible d'utiliser une relaxation convexe de type ℓ_1[NDEG12].

Algorithmes proximaux Une extension de l'approche par descente de gradient permettant une accélération importante de la convergence est connue sous le nom de *Iterative Shrinkage-Thresholding Algorithm* (ISTA). Ces méthodes construisent une suite de solutions $\boldsymbol{\alpha}_n$ au problème pénalisé $(\boldsymbol{L}_1^\lambda)$ (2.2.12) en effectuant :

$$\boldsymbol{\alpha}_{n+1} = \mathcal{T}_{\lambda t}\left(\boldsymbol{\alpha}_n - 2t\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{\alpha}_n - \boldsymbol{x})\right) \tag{2.3.1}$$

où $\mathcal{T}_{\lambda t}$ est un opérateur de rétrécissement :

$$\begin{aligned} \mathbb{R}^M &\rightarrow \mathbb{R}^M \\ \mathcal{T}_{\lambda t}(\alpha)_i &= (|\alpha_i| - \lambda t)_+ sgn(\alpha_i) \end{aligned} \tag{2.3.2}$$

où t est un pas d'avancement. Une étude de la convergence de ces méthodes dites proximales (car basées sur le calcul d'opérateurs proximaux), se trouve par exemple dans [CW05].

Plus récemment, Beck et Teboulle [BT09] (voir aussi Nesterov [Nes07]) ont proposé une variante accélérée d'ISTA dénommée *FISTA (Fast Iterative Shrinkage-Thresholding Algorithm)*. De nombreux travaux récents étendent ce formalisme et cet algorithme, le lecteur intéressé trouvera une description détaillée de ce type d'approche dans [BJM11].

Algorithmes basés sur des normes mixtes Dans les problèmes d'apprentissage de dictionnaire de la forme 2.2.13, on souhaite souvent obtenir une structure particulière de la représentation, par exemple pour faire en sorte que chaque élément du dictionnaire soit utilisé pour représenter un maximum de signaux d'apprentissage. Ce type de parcimonie structurée peut s'obtenir à l'aide de normes

mixtes $\ell_{p,q}$. En utilisant les notations de la section 2.2.2 :

$$\|\boldsymbol{A}\|_{p,q} = \left(\sum_{i=1}^{K} \|\boldsymbol{\alpha^i}\|_q^p \right)^{1/p} \tag{2.3.3}$$

en utilisant la norme mixte $\ell_{1,2}$ on obtient la formulation :

$$(DL_{1,2}^{\lambda}) : \min_{\boldsymbol{\Phi},\boldsymbol{A}} \left(\|\boldsymbol{X} - \boldsymbol{\Phi} \cdot \boldsymbol{A}\|_F^2 + \lambda \|\boldsymbol{A}\|_{1,2} \right) \tag{2.3.4}$$

qui permet d'obtenir une parcimonie structurée sur les lignes de $\boldsymbol{A}$. Les algorithmes de la famille Group-LASSO (et Multiple Basis Pursuit) sont basés sur cette formulation. Rakotomamonjy [Rak11] présente une étude des variantes disponibles. Le lecteur intéressé trouvera également des détails dans l'article de Mairal *et al* [MBP+08].

2.3.2 Algorithmes Bayésiens

Ces méthodes proposent de modéliser $\boldsymbol{\alpha}$ comme une variable aléatoire continue dont la densité de probabilité *a priori* $p(\alpha)$ présente un profil parcimonieux. La parcimonie ne s'entend dans ce cas de figure pas nécessairement de façon stricte (nombre faible de coefficients non nuls) mais par la grande disparité entre les coefficients, induite par la longueur de la queue de la distribution et la valeur du pic en zéro. Une mesure possible de cette caractéristique est le *kurtosis* ou coefficient d'aplatissement. Le signal x est appelé observation et son modèle s'écrit dans le cas bruité

$$x = \sum_{i=0}^{M-1} \alpha_i d_i + e \tag{2.3.5}$$

où e est généralement modélisé par un bruit blanc gaussien centré de variance σ_e^2 :

$$p(x|\boldsymbol{\alpha}) = \frac{1}{Z} \exp\left(\frac{-|x - \sum_{i=0}^{M-1} \alpha_i d_i|^2}{2\sigma_e^2} \right) \tag{2.3.6}$$

Dans un cadre Bayésien, on écrit :

$$p(\alpha|x) = \frac{1}{Z(x)} p(x|\boldsymbol{\alpha}) p(\boldsymbol{\alpha}) \tag{2.3.7}$$

où $Z(x)$ est un facteur de normalisation et le problème consiste désormais à trouver $\boldsymbol{\alpha}$ qui maximise la distribution *a posteriori* $p(\boldsymbol{\alpha}|x)$ en prenant comme distribution *a priori* :

$$p(\boldsymbol{\alpha}) \propto \exp\left(-\lambda \|\boldsymbol{\alpha}\|_p\right) \tag{2.3.8}$$

où λ est désormais un hyper-paramètre. Ainsi la solution du problème P_1 peut se voir comme l'estimateur du maximum a posteriori (MAP) :

$$\arg\max_{\boldsymbol{\alpha}} p(\boldsymbol{\alpha}|x) = \arg\min_{\alpha} \{-\log(p(\boldsymbol{\alpha}|x)\}$$

où l'on fait l'hypothèse que $\boldsymbol{\alpha}$ est i.i.d selon $p(\alpha_i) \propto \exp(-\lambda|\alpha_i|)$.

Différents estimateurs Bayésiens peuvent alternativement servir à la définition de la solution : erreur quadratique moyenne [SPZ08], maximum de vraisemblance [OF97], etc..). On retrouve par exemple ces méthodes chez Olshausen et Field [OF97] avec une distribution de Cauchy, chez Lewicki et Sejnowski [LS00] avec une distribution Laplacienne. Plus récemment, Févotte et Godsill utilisent

une distribution de Student-t [FG05] puis une distribution inverse de Jeffrey [FG06] et appliquent ce formalisme à la séparation de sources aveugles.

Mais les modèles les plus récents s'appuient surtout sur une double modélisation du support s (c-à-d. de l'ensemble des atomes de la représentation) et des poids des atomes w :

$$x = \sum_{i=0}^{M-1} w_i s_i d_i + e \tag{2.3.9}$$

avec $s_i \in \{0,1\}$ une variable aléatoire binaire et w_i une seconde variable aléatoire décrivant le poids de l'atome d_i. Le plus utilisé de ces modèles est le modèle Bernoulli-Gaussien [SPZ08, ZBZJ09], qui s'accompagne de façon standard d'une hypothèse d'indépendance sur les composants permettant d'écrire :

$$p(s) = \prod_{i=0}^{M-1} p(s_i) \text{ avec } p(s_i) = Ber(p_i) \tag{2.3.10}$$

$$p(w) = \prod_{i=0}^{M-1} p(w_i) \text{ avec } p(w_i) = \mathcal{N}(0, \sigma_{x_i}^2) \tag{2.3.11}$$

où $\sigma_{x_i}^2$ est la variance associée à la i-ième composante de x, et dans lequel les poids sont gaussiens et les activations s_i suivent une loi de Bernoulli de paramètre $p_i \ll 1$. Le support et les amplitudes sont ensuite estimés (*p. ex.* alternativement) par maximisation des vraisemblances marginales *a posteriori*. Ce type d'approches est particulièrement adapté au problème de recouvrement de supports parcimonieux.

Des travaux récents [Cev08, GCD12] s'attachent à montrer les liens entre ces modèles probabilistes et les algorithmes de reconstruction basés sur des relaxations de la contrainte. Un résultat important obtenu par Gribonval *et al* [GCD12] est de démontrer la non équivalence du problème (P_1) avec le problème *MAP* Bayésien basé sur un *a priori* Laplacien.

2.3.3 Algorithmes itératifs

Parmi les algorithmes itératifs, on trouve ceux basés sur un seuillage (doux ou fort) des projections du signal sur le dictionnaire, et les algorithmes gloutons. Étant au centre de ce travail de thèse, ce deuxième type d'approches fait l'objet du chapitre suivant on se limite donc ici à mentionner quelques exemples d'algorithmes itératifs basés sur un seuillage.

Blumensath *et al* [BYD07], reprenant les travaux de Herrity *et al* [HGT06], proposent une régularisation de P_0 à l'aide d'une fonction de coût alternative, la forme pénalisée du problème s'écrit :

$$(L_0^\beta) : \min_{\boldsymbol{\alpha}} \|\boldsymbol{x} - \boldsymbol{\Phi} \cdot \boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_0 - \|\boldsymbol{\Phi} \cdot \boldsymbol{\alpha} - \boldsymbol{\Phi} \cdot \boldsymbol{\beta}\|_2^2 + \|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2^2 \tag{2.3.12}$$

Cette formulation permet de découpler les éléments de $\boldsymbol{\alpha}$ dans l'optimisation. Il est donc possible d'optimiser selon chaque composante de $\boldsymbol{\alpha}$ indépendamment des autres. Les auteurs montrent alors une convergence garantie vers un minimum local du problème (P_0), et présentent un algorithme construisant une suite de solutions $\boldsymbol{\alpha}^n$:

$$\boldsymbol{\alpha}^{n+1} = H_{\lambda^{0.5}}\left(\boldsymbol{\alpha}^n + \boldsymbol{\Phi}^H(\boldsymbol{x} - \boldsymbol{\Phi}\boldsymbol{\alpha}^n)\right) \tag{2.3.13}$$

où

$$H_{\lambda^{0.5}}(\alpha_i) = \begin{cases} 0 \text{ si } |\alpha_i| \leq \lambda^{0.5} \\ \alpha_i \text{ si } |\alpha_i| > \lambda^{0.5} \end{cases}$$

Dans un travail plus récent [DMM09], DONOHO *et al* décrivent un algorithme de seuillage itératif nommé *Approximate Message Passing* (AMP) qui reprend et améliore l'idée ci-dessus. Les auteurs montrent de très bonnes performances de cet algorithme pour des problèmes de recouvrement (globalement très similaires aux performances pratiques et théoriques des méthodes convexes) à un coût très inférieur en termes de complexité.

Avertissement Ce type d'algorithme (et ses variantes récentes [KMS+12]) sont *de facto* les plus efficaces à ce jour sur (certains) problèmes de *Compressed Sensing*. La grande variété des champs d'applications a pour conséquence la répartition de certains types d'algorithmes sur certains sous problèmes où la physique ou les dispositifs expérimentaux contraignent la forme de dictionnaire et introduisent de la structure dans la parcimonie. Ce champs de recherche est, de plus, particulièrement actif. Tous les jours de nouveaux travaux sont publiés et des variantes algorithmiques proposées. Il est donc illusoire d'espérer en dresser (surtout ici) un panorama exhaustif ou définitif et l'on ne peut que supposer que cet inventaire sera obsolète dans un futur assez proche.

Pour pallier ces inévitables lacunes, le lecteur intéressé pourra se référer au blog *Nuit Blanche* tenu par I. CARRON[3]. Il y tient une description des dernières avancées tant algorithmiques que théoriques et propose des liens vers des implémentations.

3. http ://nuit-blanche.blogspot.fr/

Chapitre 3

Algorithmes gloutons de décompositions parcimonieuses

Les algorithmes gloutons (de l'anglais *greedy*), construisent une suite de solutions aux problèmes de représentations parcimonieuses en sélectionnant les composantes de façon itérative. Dans ce chapitre nous présentons le *Matching Pursuit* (MP) ainsi que ses variantes les plus connues. Dans le cas général, on dénote $\mathcal{D}$ un dictionnaire d'éléments d_i. Lorsque tous les éléments ont mêmes dimensions, on utilisera plutôt l'écriture matricielle $\mathbf{\Phi}$ où les atomes ϕ_i sont les colonnes de la matrice $\mathbf{\Phi}$.

3.1 Matching Pursuit

3.1.1 Formulation standard

L'algorithme de Matching Pursuit (MP), introduit par Mallat et Zhang [MZ93], est connu dans la communauté statistique sous le nom de Projection Pursuit [Hub85]. On le trouve également dans la communauté de codage de la parole [DMV90] pour la quantification vectorielle adaptative. Une présentation des nombreuses variantes est donnée par Dymarski *et al* [DMR11]. Celles-ci partagent toutes une structure itérative sous-jacente à deux étapes. Soit un espace de Hilbert $\mathcal{H}$ et soit un dictionnaire $\mathcal{D}$ d'éléments $d_\gamma \in \mathcal{H}$ (alternativement une matrice Φ d'atomes ϕ_γ), les algorithmes de la famille MP construisent une suite d'approximations $\tilde{x}_n = \sum_{i=1}^{n} \alpha_i d_{\gamma^i}$ (alternativement $\tilde{x}_n = \Phi_{\Gamma^n}\boldsymbol{\alpha}$, où $\boldsymbol{\alpha}$ est un vecteur parcimonieux dont les éléments non nuls sont indexés par l'ensemble Γ^n) de $x \in \mathcal{H}$ après n itérations en alternant comme montré Figure 3.1.1 :

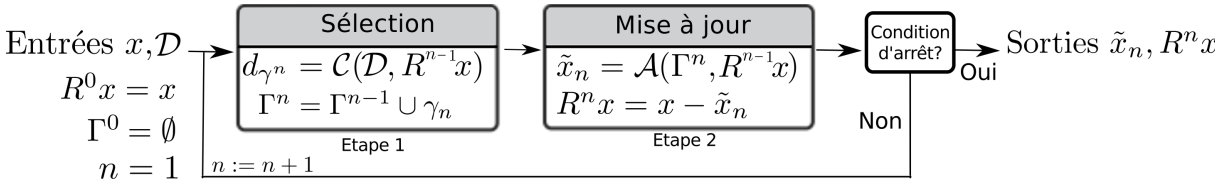

Figure 3.1.1: Diagramme fonctionnel des algorithmes gloutons : alternance d'étapes de sélection et de mise à jour jusqu'à remplir une condition d'arrêt. . L'algorithme construit une suite d'approximations $\tilde{x}_n$ de x en : 1) sélectionnant parmi les éléments du dictionnaire $\mathcal{D}$ d'après un critère $\mathcal{C}(\mathcal{D}, R^{n-1}x)$ sur le signal résiduel courant $R^{n-1}x$ et 2) effectuant une mise à jour de l'approximation et du signal résiduel.

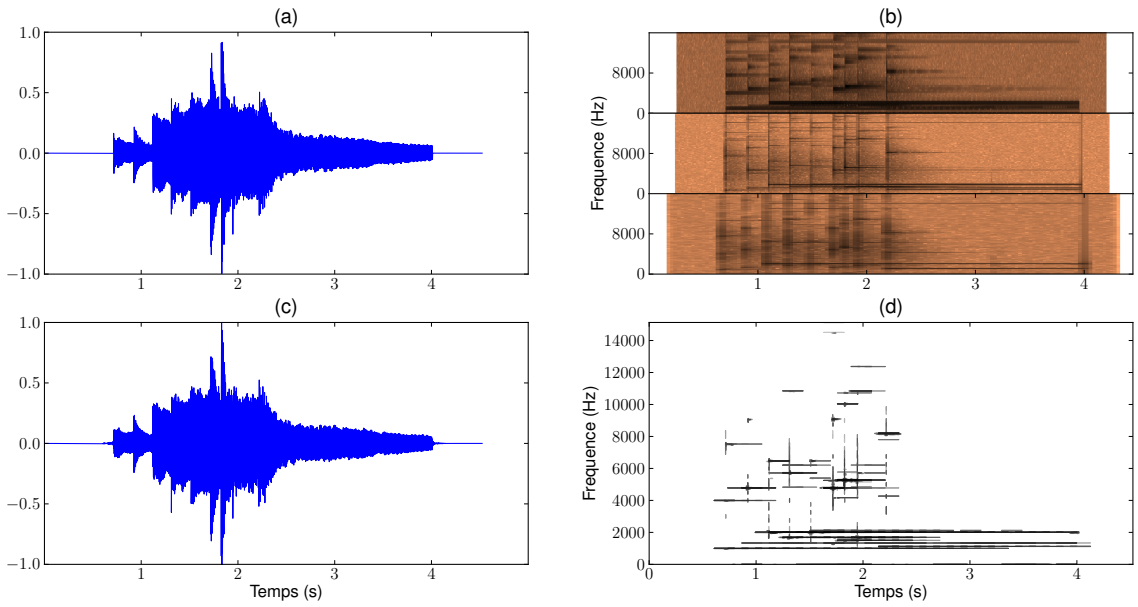

FIGURE 3.1.2: Exemple d'approximation obtenue par *Matching Pursuit* simple. (a) forme d'onde du signal original x (échantillonnage à 32000 Hz). (b) projections du résiduel initial (c-à-d. $R^0 x = x$) sur un dictionnaire constitué d'une union de bases MDCT (échelles de 8, 32 et 256 ms). (c) approximation obtenue au bout de 1000 itérations $\tilde{x}_{n=1000}$ et (d) visualisation de $\tilde{x}_{n=1000}$ dans le plan temps fréquence, chaque atome est figuré par une ellipse localisée en temps et en fréquence et de dispersion contrainte par l'échelle de l'atome.

- Étape 1 : La sélection d'un (ou plusieurs) élément du dictionnaire selon un critère $\mathcal{C}$
- Étape 2 : La mise à jour de l'approximation selon une règle $\mathcal{A}$

A chaque étape, MP met à jour le signal résiduel $R^n x = x - \tilde{x}_n$. Les différentes variantes se distinguent par leur mise en oeuvre de ces deux étapes. Le MP standard est caractérisé par le couple :

$$d_{\gamma^n} = \mathcal{C}_{MP}(\mathcal{D}, R^{n-1}x) = \arg\max_{d_i \in \mathcal{D}} |\langle R^{n-1}x, d_i \rangle| \tag{3.1.1}$$

$$\tilde{x}_n = \mathcal{A}_{MP}(\Gamma^n, R^{n-1}x) = \sum_{i=1}^{n} \langle R^{i-1}x, d_{\gamma^i} \rangle \cdot d_{\gamma^i} \tag{3.1.2}$$

où Γ^n est l'ensemble des indices $\{\gamma^i\}_{i=1..n}$ des éléments successivement sélectionnés. Parmi les caractéristiques principales de MP, on peut citer :

- Critère de sélection basé sur les produits scalaires entre le résiduel courant et les éléments du dictionnaire.
- Mise à jour simple, ne nécessite que la connaissance du dernier élément choisi ($\tilde{x}_n = \tilde{x}_{n-1} + \langle R^{n-1}x, d_{\gamma^n} \rangle \cdot d_{\gamma^n}$)
- Garantie de convergence exponentielle de l'erreur $\epsilon_n = \|x - \tilde{x}_n\|^2$ [MZ93] en dimension finie si le dictionnaire est complet ($span(\mathcal{D}) = \mathcal{H}$).

La vitesse de convergence est fonction de l'adéquation du dictionnaire au signal et de l'inter-corrélation entre les atomes.

Outre sa simplicité, les implémentations du MP sont souvent efficaces, en particulier dans le cas de dictionnaires structurés. La plupart de ces optimisations sont mises en oeuvre, par exemple, dans le *Matching Pursuit ToolKit* (MPTK) et décrites dans [KG06].

Algorithm 1 Matching Poursuit (MP)

Entrées: x , $\mathcal{D}$

1: $R^0x := x$, $\Gamma^0 = \emptyset$, $n = 1$
2: **Répéter**
3: **Etape 1** : Selection indice γ_n :
$d_{\gamma_n} \leftarrow \mathcal{C}(\mathcal{D}, R^{n-1}x)$
$\Gamma^n \leftarrow \Gamma^{n-1} \cup \gamma_n$
4: **Etape 2** : Mise à jour de l'approximation et du résiduel :
$\tilde{x}_n \leftarrow \mathcal{A}(\Gamma^n, R^{n-1}x)$
$R^nx \leftarrow x - \tilde{x}_n$
5: **Jusqu'à** ce qu'une condition d'arrêt soit remplie

Sorties: $\tilde{x}_n$, R^nx

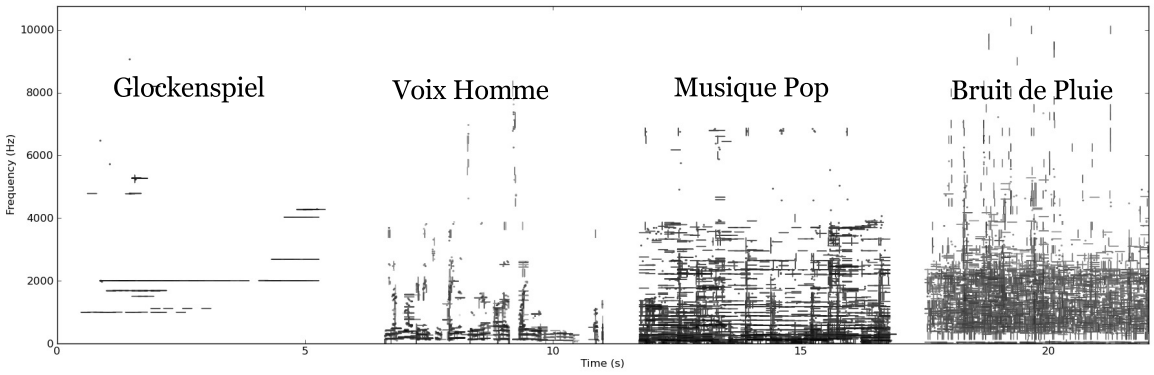

FIGURE 3.1.3: Approximation parcimonieuses de 4 scènes sonores dans une union de bases MDCT (échelles de 8, 32 et 256ms), obtenues par MP stoppé lorsqu'un SRR de 10 dB est atteint. Visualisation dans le plan temps-fréquence : chaque atome est figuré par une ellipse localisée en temps et en fréquence et de dispersion contrainte par l'échelle de l'atome.

Le critère d'arrêt du MP peut être :

- Un nombre maximum d'itérations (ou de façon équivalente un débit fixe pour un scénario de codage)
- Une qualité d'approximation, souvent sous la forme d'un Rapport Signal à Résiduel (SRR)

$$SRR(n) = 10 \log_{10} \frac{\|\tilde{x}_n\|^2}{\|x - \tilde{x}_n\|^2} \tag{3.1.3}$$

- Un critère de temps de calcul, de mémoire, etc..

Les profils de ces représentations sont très dépendants du signal considéré. La Figure 3.1.3 nous montre, dans le plan temps-fréquence, les représentations parcimonieuses des quatre scènes sonores prises en exemple au chapitre précédent, obtenues par MP dans un dictionnaire multi-échelles MDCT. Le critère d'arrêt choisi pour chacune de ces scènes est un critère de qualité de reconstruction, en l'occurrence un SRR de 10 dB. Nous pouvons observer que pour la scène de glockenspiel, très peu d'atomes sont nécessaires pour atteindre cette valeur. L'énergie est, en effet, très concentrée dans le plan temps-fréquence et l'algorithme parvient rapidement à reconstituer cette énergie à l'aide des atomes du dictionnaire. A l'inverse, pour le signal de bruit, l'algorithme n'atteint la qualité requise qu'au bout d'un très grand nombre d'itérations. La représentation du bruit dans le dictionnaire con-

sidérée est inefficace, du fait de l'inadéquation (on parlera d'incohérence) entre le dictionnaire et le signal. Pour décrire ces différents cas de figure, il faut étudier la convergence de MP.

3.1.2 Convergence

La convergence de l'algorithme se démontre [MZ93] à partir de la règle de mise à jour (3.1.2) qui permet d'écrire :

$$\|R^n x\|^2 = \|R^{n-1}x\|^2 - |\langle R^{n-1}x, d_{\gamma^n}\rangle|^2 \tag{3.1.4}$$

qui est une forme d'équation de conservation de l'énergie. Dès lors :

$$\frac{\|R^n x\|^2}{\|R^{n-1}x\|^2} = 1 - \left|\frac{\langle R^{n-1}x, d_{\gamma^n}\rangle}{\|R^{n-1}x\|}\right|^2 \tag{3.1.5}$$

ce qui fait apparaître le terme de cohérence $\mu(R^n x, \mathcal{D})$ qui définit le produit scalaire maximal entre un vecteur normalisé et les éléments du dictionnaire $\mathcal{D}$:

$$\mu(x, \mathcal{D}) = \max_{d_\gamma \in \mathcal{D}} \left|\langle \frac{x}{\|x\|}, d_\gamma\rangle\right| \leq 1 \tag{3.1.6}$$

Le théorème 12.6 de [Mal09] prouve le résultat suivant :

$$\mu_{inf}(\mathcal{D}) = \inf_{x\in\mathbb{C}^N, x\neq 0} \mu(x, \mathcal{D}) > 0 \tag{3.1.7}$$

$\mu_{inf}(\mathcal{D})$ se comprend comme la plus petite corrélation normalisée entre un vecteur non nul de $\mathcal{H}$ et le dictionnaire $\mathcal{D}$. En injectant dans (3.1.5) :

$$\frac{\|R^n x\|^2}{\|R^{n-1}x\|^2} \leq 1 - \mu_{inf}^2(\mathcal{D}) \tag{3.1.8}$$

on obtient la borne :

$$\|R^n x\|^2 \leq \left(1 - \mu_{inf}^2(\mathcal{D})\right)^n \|x\|^2 \tag{3.1.9}$$

La vitesse de convergence de l'algorithme dépend de l'adéquation signal-dictionnaire. Plus les composantes d'un signal sont *cohérentes* dans un dictionnaire (*c-à-d.* fortement corrélées avec certains atomes du dictionnaires) plus cette convergence est rapide.

3.1.3 Stabilité

La question de savoir si MP trouve la meilleure approximation à n-termes en n itérations est dénotée stabilité. Meilleure s'entend le plus souvent au sens des moindres carrés, c'est à dire celle qui minimise l'erreur de reconstruction :

$$\tilde{x}_n^\star = \arg\min_{\tilde{x}_n} \|x - \tilde{x}_n\|_2^2 \text{ soumis à } \|\Gamma^n\|_0 \leq n \tag{3.1.10}$$

où Γ^n est l'ensemble des indices des éléments du dictionnaire qui supportent l'approximation $\tilde{x}_n$. La solution est évidente dans le cas de bases orthonormales, car il suffit de sélectionner les n plus grands coefficients. Dans la mesure où $\tilde{x}_n$ est choisi dans un dictionnaire redondant, les choses sont plus complexes, mais il est toujours possible de définir la meilleure approximation à n-termes comme la projection orthogonale sur le sous espace de dimension au plus n qui minimise l'erreur de reconstruction :

$$\tilde{x}_n^\star = \arg\min_{\Gamma^n \subset \Lambda} \|x - \mathcal{P}_{\mathcal{V}(\Gamma^n)}(x)\|_2^2 \text{ soumis à } \|\Gamma^n\|_0 \leq n \tag{3.1.11}$$

autrement dit, il s'agit de trouver le support Γ^n optimal, les coefficients de l'approximation se calculent ensuite directement par projection orthogonale sur le sous-espace engendré par ce support.

Pour cette raison, la stabilité se formule souvent comme un problème de recouvrement parcimonieux. Sachant x parcimonieux dans $\mathcal{D}$ (c-à-d. $x = \sum_{i=1}^{m} \alpha_i d_{\gamma^i}$ avec $m \ll M$) on note Λ le support ($\Lambda = \{\gamma^i\}_{i=1..m}$), si MP délivre une approximation $\tilde{x}_m = \sum_{j=1}^{m} \beta_j d_{\gamma^j}$ au bout de m itérations alors on souhaite avoir $\tilde{x}_m = x$ et en particulier on souhaite que le support Γ^m de $\tilde{x}_m$ soit le même que Λ. Davis *et al* [DMZ94] ont montré que ce problème est NP-complet. La stabilité des algorithmes de Matching Pursuit Orthogonaux (voir 3.2.1) est étudiée notamment par Tropp [Tro04] et par Gribonval et Vandergheynst [GV06].

La stabilité se comprend comme une contrainte forte sur l'unicité du jeu d'atomes utilisé dans une représentation. Si cette unicité est importante, comme c'est le cas par exemple pour des problèmes de reconnaissance, de classification, où cet ensemble va servir d'identifiant ou de signature, alors la stabilité des algorithmes mis en place sera importante. A l'inverse, dans une tâche d'approximation, le but recherché est de minimiser une erreur, une distorsion, et le jeu précis d'atome utilisé dans ce but n'est pas une préoccupation centrale. Dans le cadre de l'archivage, la stabilité est donc une propriété souhaitable sur un sous-ensemble des problématiques, celles liées à l'indexation.

3.2 Variantes sur la mise à jour

3.2.1 Orthogonalisation

Le principal défaut du MP standard est que chaque itération réalise une optimisation locale sans prise en compte des atomes précédemment sélectionnés. L'approximation, en effet, n'est mise à jour que dans la direction du dernier atome sélectionné d_{γ^n}. En envisageant le sous-espace engendré par l'ensemble des atomes $\{d_{\Gamma^i}\}_{i=1..n}$ on peut réaliser une optimisation plus efficace (au sens des moindres carrés).

Dans cette optique, l'*Orthogonal Matching Pursuit* [PRK93](OMP) propose de prendre comme approximation la projection orthogonale du signal de départ sur le sous espace $\mathcal{V}(\Gamma^n)$ engendré par l'ensemble des éléments sélectionnés : $\mathcal{A}_{OMP}(\Gamma^n, R^n x) = P_{\mathcal{V}(\Gamma^n)}(x)$. Cette règle de mise à jour permet de s'assurer qu'à chaque itération, l'approximation construite $\tilde{x}_n$ est la meilleure possible au sens de l'erreur quadratique :

$$\forall y \in span(\mathcal{V}_{\Gamma^n}), \|x - y\|_2^2 \geq \|x - P_{\mathcal{V}(\Gamma^n)}(x)\|_2^2$$

Lorsqu'une écriture matricielle du dictionnaire est possible, on écrit :

$$\tilde{x}_n = \Phi_{\Gamma^n}^{\dagger} \cdot x \tag{3.2.1}$$

où $\dagger$ dénote la pseudo-inversion et Φ_{Γ^n} est une matrice composée des colonnes indexées par Γ^n.

3.2.2 Poursuites directionnelles

En revanche, cette projection coûte cher en termes de complexité. La pseudo inversion de la matrice Φ_{Γ^n} peut être accélérée en gardant une factorisation QR en mémoire d'une itération sur l'autre [DMZ94]. Mailhe *et al* [MGBV09] proposent une accélération notable dans le cas de dictionnaires structurés, à l'aide de projections locales. Dans le même temps, Blumensath et Davies [BD08] décrivent un cadre plus général de poursuite directionnelle, avec des règles de mises à jour plus rapides

et une convergence quasi-équivalente à celle d'OMP. A l'itération n, l'atome d_n est sélectionné et une direction de mise à jour est calculée sur l'ensemble Γ^n des atomes.

En adoptant les notations matricielles, l'erreur quadratique à l'itération n s'écrit $\|x - \Phi_{\Gamma^n} c_{\gamma^n}\|_2^2$. Le gradient de cette expression relativement à c s'écrit(3.2.2)

$$g_{\Gamma^n} = \Phi_{\Gamma^n}^T \cdot (x - \Phi_{\Gamma^n} c_{\gamma^n}) \tag{3.2.2}$$

le pas de descente optimal est alors :

$$a^n = \frac{\langle R^{n-1}x, b_n \rangle}{\|b_n\|_2^2} \tag{3.2.3}$$

où $b_n = \Phi_{\Gamma^n} \cdot g_{\Gamma^n}$ et la mise à jour se fait selon(3.2.4) :

$$\tilde{x}_n = \tilde{x}_{n-1} + a^n g_{\Gamma^n} \tag{3.2.4}$$

Cet algorithme est alors noté *Gradient Pursuit* (GP). D'autres directions sont envisageables et font écho aux méthodes traditionnelles de descente de gradient (p. ex. gradients conjugués) pour résoudre les problèmes d'optimisation convexes (et en particulier lorsque la fonction de coût est quadratique).

3.2.3 Matching Pursuit Cyclique

Une autre variante, nommée *Cyclic MP* [CJ07] autorise la correction des paramètres des atomes précédemment sélectionnés de façon cyclique. Contrairement à OMP ou GP, chaque atome d_i est raffiné en prenant en compte un critère intermédiaire de distorsion :

$$d_i^{(1)} \leftarrow \arg\max_{d \in \mathcal{D}} \|R_i^n x - \langle R_i^n x, d \rangle d\| \tag{3.2.5}$$

où

$$R_i^n x = x - \sum_{m=1}^{i} c_i d_{\gamma^m}^{(1)} - \sum_{m=i+1}^{n} c_i d_{\gamma^m} \tag{3.2.6}$$

Ainsi l'atome d_i est raffiné en utilisant le résiduel intermédiaire $R_i^n x$ qui décrit la contribution des atomes déjà raffinés. L'optimisation n'est pas globale comme pour OMP ou GP mais l'ensemble des atomes est néanmoins considéré lors de la phase de mise à jour. De plus, *Cyclic* MP permet de remplacer des atomes précédemment sélectionnés par d'autres, meilleurs. Cette caractéristique permet à la variante orthogonale de *Cyclic* MP (COLS) proposée par Sturm et Christensen [SC10] de présenter de bonnes performances sur le problème de recouvrement.

3.3 Variantes sur le critère de sélection

3.3.1 Sélection haute résolution

De façon complémentaire, il est possible de modifier le critère de sélection des atomes pour adapter le MP standard et améliorer ses performances. Une première de ces variantes est proposées par Jaggi *et al* [JKMW98] et Gribonval *et al* [GBM+96] sous le nom de *High Resolution Pursuit* (HRP). Partant de l'observation que dans certains cas, la sélection des atomes par MP ne permet pas de trouver une solution physiquement réaliste, les auteurs mettent à jour un phénomène connu sous le nom de *Dark Energy* [SSDR08] et qui, sur des signaux audio se traduit souvent par un phénomène de pré-écho [RRD08].

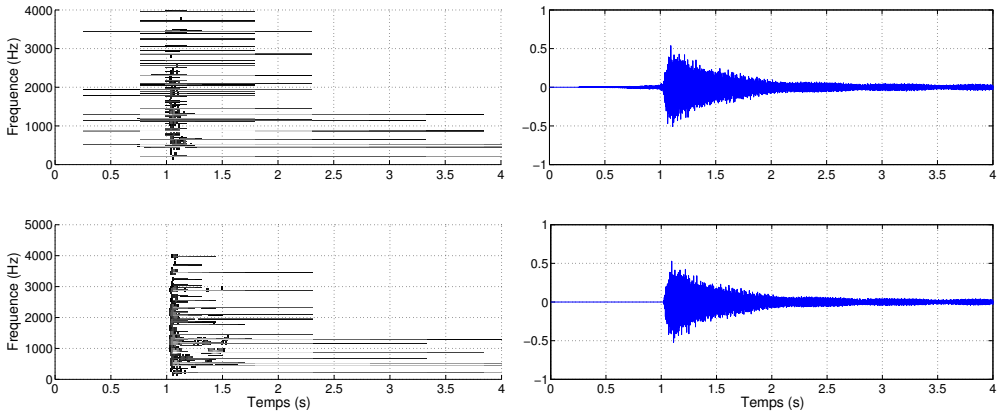

FIGURE 3.3.2: Décompositions d'un signal de cloche par Matching Pursuit (Ravelli [RRD08]) sans (haut) et avec (bas) mécanisme de contrôle du pré-echo. Le mécanisme empêche l'apparition d'énergie avant l'attaque et préserve mieux les caractéristiques perceptives du son. En revanche, plus d'atomes sont nécessaires pour atteindre un même niveau de reconstruction (SRR = 20dB).

En cause, la corrélation à l'échelle sub-atomique entre le signal et l'atome choisi par MP. Un exemple est la décomposition d'un signal percussif dans un dictionnaire harmonique.

Le produit scalaire $\langle x, d_j \rangle$ peut alors se voir comme la moyenne de corrélations plus localisées $\langle x, d_j^l \rangle$ où $d_j = \sum d_j^l$. Pour pallier ce phénomène, Jaggi *et al* affinent l'atome sélectionné en prenant comme critère le pire des sous-produits scalaires :

$$\mathcal{C}_{HRP} = \arg\max_{d_j \in \mathcal{D}} \min_l \left| \frac{\langle R^n x, d_j \rangle}{\langle d_j, d_j^l \rangle} \right| \qquad (3.3.1)$$

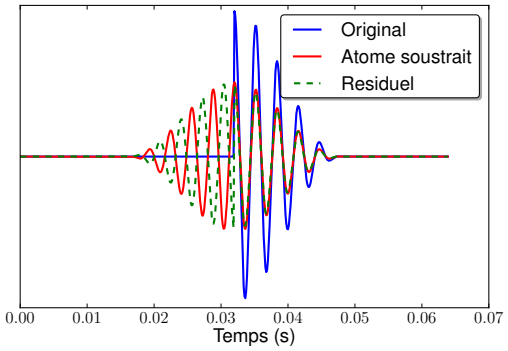

FIGURE 3.3.1: Effet de pré-écho lié à la sélection d'un atome symétrique. Bien que son énergie totale décroisse, le résiduel voit son énergie locale augmenter en début de signal.

le gain en résolution nécessite donc un calcul supplémentaire, mais pour toute une gamme de signaux (en particulier les sons percussifs) la préservation de l'attaque permet une amélioration significative de la qualité perceptive de l'approximation.

Plus récemment, RAVELLI *et al* [RRD08] ont proposé un mécanisme similaire de contrôle du pré-echo, qui consiste, après la sélection d'un atome, à annuler une partie de son support temporel en cas de mauvaise corrélation locale avec le résiduel.

3.3.2 Sélection sous-optimale : *Weak* MP

Souvent, la taille du dictionnaire $\mathcal{D}$ rend la recherche du maximum des produit scalaire trop coûteuse. C'est en particulier le cas pour des dictionnaires structurés extrêmement redondants, voire pour des dictionnaires de taille infinie (comme c'est le cas par exemple pour des dictionnaire d'atomes

chirpés [Gri01]). Dans ces cas de figure, on doit se contenter de sélectionner un atome sous-optimal au regard du critère $\mathcal{C}$ choisi.

On parle alors de MP *faible* (Weak MP) car on se contente de sélectionner à l'itération n un atome d_{γ^n} tel que :

$$|\langle R^{n-1}x, d_{\gamma^n}\rangle| \geq t_n \sup_{d_i \in \mathcal{D}} |\langle R^{n-1}x, d_i\rangle| \tag{3.3.2}$$

où $0 < t_n \leq 1$ est un facteur de sous-optimalité. En pratique ce type de sélection est obtenue en limitant le nombre de produits scalaires effectivement calculés à un sous-ensemble représentatif, et en cherchant parmi ce sous-ensemble l'élément maximum. Un choix approprié du sous-ensemble permettra de garantir un facteur de sous-optimalité $t_n \geq \alpha$ et d'assurer une convergence rapide du résiduel. Temlyakov [Tem02] a montré que *Weak* MP converge si et seulement si :

$$\sum \frac{t_n}{n} = \infty \tag{3.3.3}$$

La stabilité du *Weak MP* (c-à-d. sa capacité à retrouver la meilleure approximation à m-termes en m itérations) a été étudiée par Gribonval et Vandergheynst [GV06].

3.3.3 Sélection adaptative

Un cas particulier de MP faible consiste, dans des dictionnaires structurés, à effectuer une sélection sous-optimale en deux étapes :

- Recherche du maximum parmi un sous-ensemble d'atomes (cette étape s'apparente à une sélection faible standard) suivie de
- Optimisation de l'atome sélectionné

Typiquement, un atome est sélectionné dans une grille temps-fréquence grossière, puis sa localisation temps-fréquence est optimisée. Plus généralement on peut rapprocher ces techniques des méthodes de réassignement temporel (Auger et Flandrin [AF95]) qui s'attachent à relocaliser dans le plan temps-fréquence les composantes d'un signal. Cette sélection en deux étapes se trouve déja dans l'article de Mallat et Zhang [MZ93] ou une recherche d'optimum dans le voisinage d'un atome sous-optimal par méthode de Newton est proposée. Goodwin et Vetterli [GV99] proposent pour leur part une optimisation du paramètre de phase, Gribonval propose une telle méthode pour des atomes chirpés [Gri01] et Christensen et Jensen [CJ07] proposent une optimisation de la phase et de la fréquence des atomes.

A titre d'exemple, l'algorithme 2 présente un *Matching Pursuit* avec une phase d'optimisation locale du paramètre de localisation temporelle de l'atome sélectionné. Il est possible avec ce type d'algorithmes, de simplifier l'étape de sélection, quitte à devoir, pour chaque atome, estimer des paramètres supplémentaires d'optimisation. Ainsi pour l'exemple de l'algorithme 2, la représentation $\tilde{x}_n$ de x nécessite, en plus des indices Γ^n des atomes sélectionnés et de leurs poids, une séquence $\{\tau_i\}_{i=1..n}$ de paramètres temporels.

3.3.4 Sélection de molécules

Dans l'article [JVF06], Jost *et al* proposent une variante de MP tirant profit d'une structure de dictionnaire en arbre. Le processus de sélection est présenté sous la forme d'une classification de plus en plus fine. Les atomes du dictionnaire sont préalablement regroupés en arbre.

Algorithm 2 Matching Pursuit - Adaptatif

Entrées: x , $\mathcal{D}$

1: $R^0x := x$, $\Gamma^0 = \emptyset$, $n = 1$
2: **Répéter**
3: **Etape 1** : Selection indice γ_n :
$d_{\gamma_n} \leftarrow \mathcal{C}_{Faible}(\mathcal{D}, R^{n-1}x)$
$\Gamma^n \leftarrow \Gamma^{n-1} \cup \gamma_n$
4: **Etape 1-bis : Optimisation locale de l'atome**
$\tau_n = \arg\max_\tau |\left\langle R^{n-1}x, (d_{\gamma_n} * \delta_\tau)\right\rangle|$
5: **Etape 2** : Mise à jour de l'approximation et du résiduel :
$\tilde{x}_n \leftarrow \mathcal{A}(\Gamma^n, R^{n-1}x)$
$R^nx \leftarrow x - \tilde{x}_n$
6: **Jusqu'à** ce qu'une condition d'arrêt soit remplie

Sorties: $\tilde{x}_n$, R^nx , $\{\tau_i\}_{i=1..n}$

Cette structure permet d'accroître la vitesse de l'algorithme, le nombre de produits scalaires à calculer est d'autant plus réduit que le dictionnaire est structuré. La construction de molécules permet de forcer cette structuration.

Plus généralement, il est possible d'étendre le MP en autorisant la sélection d'une combinaison d'atomes à chaque itération. De telles combinaisons sont dénotées *molécules* et les algorithmes basés sur ce principe sont appelés : *Molecular Matching Pursuit* [Dau06](MMP). Le *Tree-Based* Pursuit peut se comprendre comme un MMP facilité par une structure en arbre. Daudet [Dau06] propose d'utiliser un dictionnaire redondant construit à partir d'une base MDCT et d'une base d'ondelettes dyadiques pour décomposer efficacement les composantes harmoniques et transitoire d'un son. Il présente ensuite deux types de molécules :

- tonale : construite comme un ensemble d'atomes MDCT voisins dans le plan temps-fréquence
- transitoire : construite comme un ensemble d'ondelettes formant un arbre dans le plan temps-échelle

Il est également possible de définir des molécules harmoniques, cela a notamment été proposé pour des tâches de reconnaissance d'instrument [LVRD08] et d'*Inpainting* [MLA10].

3.3.5 Sélection de sous-espaces

La sélection de molécules requiert la définition préalable des structures (arbre, transitoire, partiels, etc..) d'atomes recherchés. Dans un cadre plus large, la sélection d'un ensemble d'atomes à chaque itération peut se voir dans un espace de Hilbert comme la sélection, non pas d'un vecteur, mais d'un sous-espace. Cette idée permet d'accélerer la convergence, et se trouve donc à la base de nombreux algorithmes de reconstructions parcimonieuses tels que *Regularized OMP* [NV10], *CoSaMP* [NT10], *Subspace Pursuit* [DM09] ou encore *Stagewise OMP* [DTDS06, BD09]

Cette sélection multiple permet de réduire considérablement le nombre d'étapes nécesaire pour atteindre la précision recherchée. Cette propriété est particulièrement intéressante lorsque la mise à jour est lourde (p. ex. projection orthogonale). De fait, ces approches s'avèrent efficaces dans le cas de signaux très parcimonieux ou compressibles échantillonnés à travers une matrice préservant la géométrie de cette parcimonie. Les performances de ces algorithmes sont en particulier liées à la propriété d'isométrie restreinte (RIP) proposée par Candès et Tao (et démontrée par Baraniuk *et al* [BDDW08]). L'algorithme *Stagewise Weak Conjugate Gradient Pursuit* proposé par Blumensath

et Davies [BD09] rassemble ainsi un grand nombre des variations mises en oeuvre autour du Matching Pursuit :

- Sélection d'un sous-espace (*Stagewise*)
- Critère de sélection sous-optimal (*Weak*)
- Mise à jour dans une direction basée sur un calcul approché de gradient (*Conjugate Gradient).*

On notera également l'adaptation de ce type d'algorithmes au paradigme de parcimonie à l'analyse [GE12]. Encore une fois, nous insistons sur la difficulté de tenir à jour une liste exhaustive des variantes du Matching Pursuit dans la littérature.

3.4 Matching Pursuit Stochastiques

3.4.1 Matching pursuits et probabilités

Dans leur article original [MZ93], Mallat et Zhang proposent un critère d'arrêt similaire à un seuillage dans une base orthonormale. Plus précisément, soit w un bruit blanc gaussien, la cohérence de w est notée $\mu(w, \mathcal{D})$ et son espérance $E(\mu(w, \mathcal{D}))$. Un signal contient p composantes cohérentes si et seulement si, pour tout $0 \leq n < p$:

$$\mu(R^n x, \mathcal{D}) > E(\mu(R^n w, \mathcal{D})) \tag{3.4.1}$$

et

$$\mu(R^p x, \mathcal{D}) \leq E(\mu(R^p w, \mathcal{D})) \tag{3.4.2}$$

L'algorithme est donc stoppé lorsque les corrélations entre le résiduel et le dictionnaire sont comparables aux corrélations moyennes d'un bruit. Il est difficile néanmoins de modéliser les propriétés d'un tel signal de bruit à partir d'un seul exemple. Ferrando *et al* [FDBB00] partent de ce constat et proposent de lancer plusieurs poursuites aléatoires sous-optimales. Ces multiples décompositions permettent d'estimer plus finement la variance du processus de décomposition du bruit $E(\mu(R^p w, \mathcal{D})$.

Une approche similaire est proposée par Durka *et al* [DIB01] mais dans un objectif légèrement différent. En étudiant les décompositions de signaux biomédicaux à l'aide de dictionnaires structurés, les auteurs mettent en évidence un biais statistique important lié au choix a priori du dictionnaire (en particulier les dictionnaires basés sur des transformées temps-fréquence). Ils proposent donc d'initialiser de façon aléatoire les paramètres du dictionnaire (en particulier la localisation d'atomes en temps et en fréquence) avant de lancer plusieurs décompositions, dans un esprit proche des méthodes de Monte-Carlo. Le but est, en fusionnant les différents supports obtenus, de s'affranchir du biais introduit par le choix de l'un ou l'autre jeu de paramètres du dictionnaire.

Le même paradigme (que l'on retrouve illustré en Figure 3.4.1) se retrouve dans le travail de Elad et Yavneh [EY09], qui mettent là aussi en exergue l'intérêt de construire plusieurs approximations sous-optimales et de les combiner par la suite. Cette fois-ci ce ne sont pas les paramètres du dictionnaire qui sont confiés au hasard, mais le choix des atomes sélectionnés eux-mêmes. La moyenne de ces décompositions sous-optimales multiples permet, dans certains cas, d'approcher (au sens des moindres carrés) plus finement le signal cible que ne pourrait le faire une seule décomposition, fût-elle optimale.

Une autre approche, dénotée *Statistical Matching Pursuit* (SMP), est proposée par Wang *et al* [WLMJ97] (reprise sous le nom de *Stochastic Matching Pursuit* par exemple dans [WG97]). Dans ces travaux, les auteurs s'attachent à construire une approximation non pas d'un signal x (vu ici comme

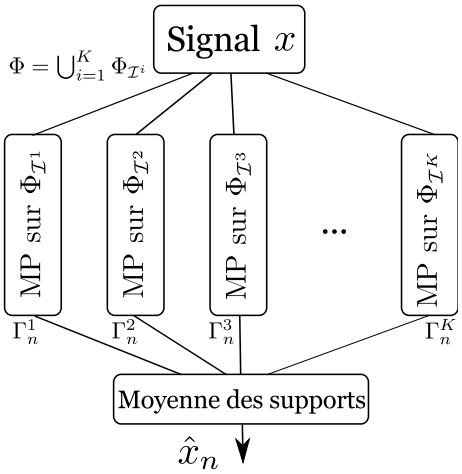

FIGURE 3.4.1: Schéma des approches stochastiques présentées par FERRANDO [FDBB00],DURKA [DIB01], et ELAD [EY09]. Plusieurs poursuites sous-optimales sont lancées en parallèle, une étape ultérieure permet de retrouver un support sans biais.

un vecteur aléatoire), mais de la distribution de probabilité de celui-ci. En ce sens, leur algorithme est très proche de la formulation originale, avec comme principale différence un critère de sélection statistique :

$$\mathcal{C}_{SMP}(R^n x, \Phi) = \arg\max_{\phi \in \Phi} E\left[\langle R^n x, \phi \rangle^2\right] \tag{3.4.3}$$

Il s'agit donc plutôt d'un algorithme d'apprentissage, son utilisation est d'ailleurs proposée dans le cadre d'extraction de descripteurs [WG97].

3.4.2 Poursuites Bayésiennes

Comme déja mentionné en section 2.3.2, une formulation Bayesienne du MP a été proposée [SPZ08, ZBZJ09]. Il s'agit essentiellement d'algorithmes de type *Orthogonal Matching Pursuit*, pour lesquels le critère de sélection est modifié pour prendre en compte un modèle de bruit gaussien (2.3.5). On peut distinguer néanmoins au minimum deux algorithmes différents : Le *Bayesian Matching Pursuit* proposé par SCHNITER *et al* [SPZ08] formule la construction du support de la représentation sous la forme d'une recherche MAP. Comme le calcul exhaustif des configurations n'est pas envisageable, la recherche se fait dans un sous-ensemble de supports candidats, déterminés à l'aide d'un fort *a priori* sur la distribution de la variable aléatoire :

$$\nu(s, x) = \ln p(x|s)p(s) \tag{3.4.4}$$

qui sert de mesure de sélection de modèle, où s est la variable décrivant le support binaire (modèle de type Bernoulli-Gaussien). Dans le même article, les auteurs proposent également un algorithme basé sur un critère de minimisation d'erreur quadratique moyenne (MMSE).

Dans la variante *Bayesian Pursuit Algorithm* (BPA) proposée par ZAYYANI *et al* [ZBZJ09], le même modèle Bernoulli-Gaussien est utilisé, mais l'ajout de chaque atome se fait sur la base d'un test statistique et d'une règle de décision :

$$s_j = 1 \text{ si } |\langle x, \phi_j \rangle - \sum_{i \neq j} \tilde{\alpha}_j \langle \phi_j, \phi_i \rangle| > Th_j \tag{3.4.5}$$

où $\tilde{\alpha}_j$ est l'estimation courante du coefficient de l'atome ϕ_j et Th_j un seuil dépendant des hypothèses et en particulier de la variance du bruit. BPA sélectionne donc les atomes pour lesquels la projection $\langle x, \phi_j \rangle$ est *suffisamment* plus grande que la somme des termes d'interférences $\sum_{i \neq j} \tilde{\alpha}_j \langle \phi_j, \phi_i \rangle$ pour prendre la décision d'adjoindre l'atome ϕ_j au support.

Récapitulatif

Nom	$\mathcal{A}$
Standard MP [MZ93, Hub85]	Règle simple de mise à jour : $\tilde{x}_n = \tilde{x}_{n-1} + \langle R^n x, d_{\gamma^n} \rangle \cdot d_{\gamma^n}$
Orthogonal MP [PRK93]	Meilleure approximation à n-termes par projection orthogonale : $\tilde{x}_n = P_{\mathcal{V}(\Gamma^n)}(x)$
Gradient Pursuit [BD08]	$\tilde{x}_n = \tilde{x}_{n-1} + a^n \cdot g_{\Gamma^n}$ avec g_{Γ^n} fonction du critère à minimiser
Cyclic MP [CJ07]	Raffinement des atomes, Voir (3.2.6)

TABLE 3.4.1: Récapitulatif, principales variantes de la règle de Mise à jour.

Nom	$\mathcal{C}$
Standard MP [MZ93, Hub85]	Produit scalaire maximal
HRP [GBM+96, JKMW98]	Corrélation locale maximale (ou décorrélation minimale)
Weak MP [GN01, Tem02]	Produit scalaire sous-optimal
MPA [Gri01, MDR11]	Sélection sous-optimale puis optimisation de l'atome sélectionné (p.ex réassignement temps-fréquence)
Molecular MP [GB03, Dau06, LVRD08]	Sélectionne le meilleur groupe d'atome (moyenne de produits scalaires)
Tree-Based MP [JVF06]	Recherche hiérarchique dans un dictionnaire structuré en arbre. Parcours construit des molécules
Stagewise OMP [DTDS06, BD09]	Choisit les s plus grands coefficients parmi le vecteur $y = \Phi^T x$ et mise à jour directionnelle
CoSaMP [NT10]	Choisit les s plus grands coefficients parmi le vecteur $y = \Phi^T x$ Formalisme CS
Subspace Pursuit [DM09]	Choisit les s plus grands coefficients parmi le vecteur $y = \Phi^T x$ puis parmi $x_p = \Phi_{\Gamma^s} y$ Sélectionne un sous-espace à chaque itération
Regularized OMP [NV10]	Choisit les s plus grands coefficients parmi le vecteur $y = \Phi^T x$ Garantie de recouvrement si RIP respectée

TABLE 3.4.2: Récapitulatif, principales variantes du critère de sélection.

Nom	remarques
Probabilistic MP [FDBB00]	Plusieurs poursuites aléatoires sous-optimales hasard sur choix des atomes
Bayesian MP [ZBZJ09, SPZ08]	Critère de sélection Bayésien
Probabilistic OMP [DE10]	Recherche dans des sous-espaces aléatoires
Random OMP [EY09]	Plusieurs poursuites aléatoires sous-optimales hasard sur choix des atomes
Stochastic MP (Durka) [DIB01]	Plusieurs poursuites aléatoires sous-optimales hasard sur paramètres du dictionnaire
Stochastic MP (Wang) [WG97]	Critère de sélection : $\arg\max_{\phi\in\Phi} E\left[\langle R^n x, \phi\rangle^2\right]$
Stochastic MP (Peel) [PERA12]	Séquence aléatoire de sous-dictionnaires (voir section 5.3.1)
SAS MP [MDR12a]	Séquence aléatoire de sous-dictionnaires (voir chapitre suivant)

TABLE 3.4.3: Récapitulatif : Algorithmes gloutons stochastiques.

Deuxième partie

Poursuites Aléatoires et Dynamiques

Chapitre 4

Matching Pursuit à Séquence de Sous-dictionnaires

Parmi les contributions de cette thèse, nous proposons une variante du MP permettant de réaliser des décompositions de haute résolution (car utilisant de très grands dictionnaires), avec une complexité réduite. En premier lieu section §4.1, nous montrerons que les décompositions gloutonnes doivent se plier à un compromis Gain-Complexité dont le noeud est le choix du dictionnaire (section 4.1.1). Ce compromis est clairement mis à jour sous la forme d'un problème de compression (section 4.1.2).

Nous présenterons ensuite l'algorithme général section §4.2, son principe section 4.2.1 avant de nous intéresser au cas particulier des séquences aléatoires section 4.2.2. Nous verrons ensuite son comportement sur quelques simulations simples section 4.2.3. Puis, nous nous intéresserons plus particulièrement au cas de dictionnaires structurés, et notamment des dictionnaires temps-fréquence section §4.3.

Nous verrons enfin section §4.4 un cas d'application à la compression bas-débit de scènes sonores. Ce travail a fait l'objet d'une publication dans la revue *Signal Processing* en octobre 2012 [MDR12a] reprenant et étendant un article de conférence ICASSP (mars 2012 [MDR12b]).

4.1 Contexte

4.1.1 Choix du dictionnaire

Il y a derrière la plupart des variantes mentionnées au chapitre précédent (en particulier les variantes de sélection), le souci de prendre en compte l'un (ou les deux) critères suivants :

- Limiter la complexité.
- Construire la meilleure approximation (ou reconstruction) possible.

Avant même de s'intéresser aux différentes variantes, il est possible de contrôler ces deux critères en choisissant le dictionnaire. On se trouve alors assez vite devant un compromis qui se résume ainsi :

- Utiliser un petit dictionnaire permet de limiter la complexité.
- Utiliser un grand dictionnaire permet de construire de meilleures approximations (au sens des moindres carrés).

Ce constat dépend bien évidemment des cas de figures considérés. Il est basé sur une double hypothèse : i) la complexité est fonction de la taille du dictionnaire (par exemple, le nombre de colonnes d'une matrice dictionnaire détermine le nombre de produits scalaires parmi lesquels chercher le maximum)

et ii) la vitesse de décroissance de l'erreur de reconstruction est fonction de la taille du dictionnaire. Cette seconde hypothèse est directement liée à l'expression de la borne de convergence (3.1.9). Plus précisément elle suppose qu'augmenter la taille du dictionnaire augmente du même coup la cohérence des signaux dans ce dictionnaire.

Il est possible d'interpréter certaines variantes du MP à l'aune de ce compromis sur la taille du dictionnaire :

MP Faible Limite la sélection à un sous-dictionnaire fixe au prix d'une représentation sous-optimale. Équivalent à considérer un dictionnaire de taille réduite.

MP Adaptatif Limite la sélection à un sous-dictionnaire fixe puis optimise localement l'atome sélectionné. Équivalent à considérer un dictionnaire de grande taille avec une recherche efficace d'un maximum local.

MP Stochastique (au sens de [DIB01]) Considère plusieurs poursuites sur des dictionnaires de taille réduite, puis reconstruit une approximation dans le dictionnaire de grande taille

On le voit, le but est de réussir à utiliser le plus grand dictionnaire possible, tout en optimisant l'étape de sélection d'un atome à l'aide de sous-dictionnaires. Ces variantes tentent de résoudre un problème sous-jacent qui est donc une minimisation jointe, sur toutes les stratégies $\mathcal{S}$ possibles, de la complexité $\mathcal{O}(\mathcal{S})$ et de l'erreur quadratique de reconstruction $\epsilon(\mathcal{S})$:

$$\min_{\mathcal{S}} \epsilon(\mathcal{S}) \text{ et } \min_{\mathcal{S}} \mathcal{O}(\mathcal{S}) \tag{4.1.1}$$

Ce problème est mal posé ; la définition de la complexité (*p.ex.* au sens de Kolmogorov, du nombre d'opérations en virgule flottante, etc) doit être précisée. De plus il sera difficile d'associer à une stratégie une valeur ϵ, car la qualité de l'approximation est forcément dépendante de la nature des signaux considérés. il est possible alternativement d'utiliser une borne sur cette erreur.

Il est intéressant de noter que le problème (4.1.1) formule un compromis sur deux des propriétés souhaitables des représentations pour l'archivage : fidélité et simplicité (voir 1.1.2 page 4).

4.1.2 Compression à l'aide de représentations parcimonieuses

Le problème de la compression de signaux peut se comprendre comme un problème de Fidélité-Concision. Le but est, à partir d'un signal x, de construire une approximation $\hat{x}$ (on dira aussi un code) qui obéit à un compromis taille/qualité, généralement représenté sous la forme de courbes débit-distorsion. Parmi les codeurs les plus utilisés on trouve le MPEG-1 Layer III (MP3)[MP392], plus récemment le standard AAC [MPE99], ou encore le dernier né des codeurs libres OPUS [1].

Le cadre des représentations parcimonieuses est propice à la compression de signaux (*p.ex.* pour l'audio [FVFK04, RRD08], et pour les images [FVF06]). Si l'on dispose d'une approximation parcimonieuse à m-termes $\tilde{x}_m = \sum_{i=1}^{m} \alpha_i d_{\gamma^i}$ dans un dictionnaire $\mathcal{D}$, alors en quantifiant le vecteur de coefficients $\boldsymbol{\alpha} \to \hat{\boldsymbol{\alpha}} = Q(\boldsymbol{\alpha})$ on obtient une approximation quantifiée $\hat{x}_m = \sum_{i=1}^{m} \hat{\alpha}_i d_{\gamma^i}$:

$$x \Longrightarrow \tilde{x}_m = \sum_{i=1}^{m} \alpha_i d_{\gamma^i} \Longrightarrow \hat{x}_m = \sum_{i=1}^{m} \hat{\alpha}_i d_{\gamma^i} \tag{4.1.2}$$

La fidélité s'exprime alors en *rapport signal à bruit* (SNR) :

$$SNR(\hat{x}_m) = 10 \log_{10} \frac{\|\hat{x}_m\|_2^2}{\|x - \hat{x}_m\|_2^2} \tag{4.1.3}$$

1. http ://www.opus-codec.org/

ou par une mesure de distorsion perceptive. La concision est mesurée par la taille du code $\hat{\boldsymbol{\alpha}}$ (alternativement sous forme d'un débit). Ce type de quantification *a posteriori* ne permet pas d'adapter la poursuite à un critère de codage. Un état de l'art exhaustif sur la compression par représentations parcimonieuses se trouve dans [Mal09] (essentiellement pour les images) et dans la thèse d'E. Ravelli [Rav08] pour les signaux audio.

L'efficacité d'un codeur basé sur une représentation parcimonieuse dépend de deux paramètres :

- La parcimonie de la représentation $\tilde{x}_m$ pour une erreur de reconstruction donnée.
- La distribution (et l'indépendance) des valeurs quantifiées $\hat{\boldsymbol{\alpha}}$.

L'approximation $\hat{x}_m$ est complètement définie par la donnée du couple indices-coefficients $(\Gamma^m, \hat{\boldsymbol{\alpha}})$. Le débit nécessaire à sa transmission est donc :

$$R(\hat{x}_m) = R(\Gamma^m) + R(\hat{\boldsymbol{\alpha}}) \tag{4.1.4}$$

où $R(\Gamma^m)$ est le débit nécessaire à la transmission des m indices des atomes et $R(\hat{\boldsymbol{\alpha}})$ celui de la transmission des coefficients. Le codeur le plus simple considère chaque atome indépendamment, ce qui correspond à l'hypothèse que la décomposition est très parcimonieuse et non structurée. Sous cette hypothèse et en posant un modèle uniforme sur la distribution des atomes, le coût de codage d'un indice est de $\log_2 M$, avec M la taille du dictionnaire :

$$R(\Gamma^m) = |\Gamma^m| \log_2 M \leq n \log_2 M \tag{4.1.5}$$

En utilisant un codage entropique des indices, on peut améliorer substantiellement cette borne :

$$R(\Gamma^m) = m\bar{r} \tag{4.1.6}$$

où $\bar{r}$ est le coût moyen (en bits) du codage d'un atome.

Pour les coefficients, en utilisant une quantification uniforme à Q paliers on a :

$$R_Q(\hat{\boldsymbol{\alpha}}) = m \log_2(Q) \tag{4.1.7}$$

Dans le cadre d'un MP, on sait de plus que les coefficients sont d'amplitude de plus en plus faible, en utilisant la borne de convergence (3.1.9), Frossard *et al* [FVFK04] proposent de quantifier le i-ème coefficient α_i uniformément dans un intervalle rétrécissant $[0, J_i]$ où J_i est donné par la borne (3.1.9) :

$$J_i = (1 - \mu_{inf}(\mathcal{D}))^{i/2} \|x\| \tag{4.1.8}$$

on pose alors q_i le nombre de paliers de quantification du coefficient α_i. Le débit total est donné par :

$$R(\hat{x}_m) = \sum_{i=1}^{m} \log_2 q_i + r_i \tag{4.1.9}$$

La distorsion est bornée quant à elle par la somme de deux termes ; le premier est dû à l'approximation MP, le second à l'erreur de quantification :

$$\begin{aligned} D(\hat{x}_m) &\leq \|\boldsymbol{x} - \tilde{\boldsymbol{x}}_m\|_2^2 + \|\boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}}\|_2^2 \\ &\leq \|R^m x\|_2^2 + \sum_{i=1}^{m} |\alpha_i - \hat{\alpha}_i|^2 \end{aligned} \tag{4.1.10}$$

où $R^m x$ est toujours le résiduel après m itérations. Avec ces critères de compression, il est possible d'évaluer les différentes stratégies de décomposition sur une tâche de codage audio.

Stratégies On peut reformuler (4.1.1) dans le cadre de la compression par l'optimisation conjointe :

$$\min_{\mathcal{S}} D(\hat{x}_m^{\mathcal{S}}) \text{ sujet à } R(\hat{x}_m^{\mathcal{S}}) \leq R_{budget} \tag{4.1.11}$$

où $\hat{x}_m^{\mathcal{S}}$ est l'approximation quantifiée obtenue par stratégie $\mathcal{S}$ et R_{budget} un débit nominal. Frossard *et al* [FVFK04] utilisent la méthode des multiplicateurs de Lagrange pour trouver une quantification optimale, qui se comprend ici comme un nombre maximal d'atomes N_{max} à encoder. Dans un article récent, Lovisolo *et al* [LdSD10] optimisent cette méthode en modélisant la distribution des angles des résiduels et en utilisant un quantificateur de Lloyd-Max.

Dans ce travail, notre approche est légèrement différente. On cherche à effectuer une minimisation triple :

$$\min_{\mathcal{S}} D(\hat{x}_m^{\mathcal{S}}) \text{ et } \min_{\mathcal{S}} \mathcal{O}(\mathcal{S}) \text{ sujet à } R(\hat{x}_m^{\mathcal{S}}) \leq R_{budget} \tag{4.1.12}$$

où $\mathcal{O}(\mathcal{S})$ est en quelque sorte la complexité de la stratégie $\mathcal{S}$. Cette valeur est assez difficile à définir, en effet s'il est possible d'estimer le nombre d'opérations en virgule flottante pour chaque itération dans l'une ou l'autre stratégie, la vitesse de convergence est également un critère important. En pratique, un compromis consiste à mesurer les temps nécessaires pour atteindre un certain degré d'approximation sur une même machine.

Clairement le problème (4.1.12) est mal posé. Pourtant il est possible d'interpréter certaines variantes de MP à l'aune de cette formulation :

MP Faible Effort porté sur les contraintes de débit et de complexité (taille M réduite, recherche simple) au prix d'une distorsion aggravée par la sélection sous-optimale des atomes.

MP Adaptatif Effort porté sur la contrainte de distorsion et de complexité (taille M grande, recherche simple) au prix d'un débit augmenté par le(s) paramètre(s) additionnel(s).

MP Stochastique (au sens de [DIB01]) Effort porté sur la contrainte de distorsion (recherche multiple et taille totale M grande), au prix d'un débit et une complexité alourdis par la multiplicité des décompositions.

D'une façon générale, les variantes de MP proposées dans un cadre de recouvrement parcimonieux sont celles qui portent l'effort sur la contrainte de distorsion (*p.ex.* de reconstruction) et nécessitent généralement plus de ressources. Les variantes de MP proposées pour le codage audio doivent prendre tous ces critères en compte.

Dans la suite, nous proposons un algorithme qui porte l'effort sur les trois contraintes. La distorsion est minimisée par l'emploi d'un grand dictionnaire, parcouru à l'aide d'une séquence de sous-dictionnaires de taille réduite ce qui limite la complexité. Parallèlement le débit est faible grâce à la connaissance préalable de la séquence.

4.2 Matching Pursuit à Séquence de Sous-dictionnaires (SSMP)

4.2.1 Principe

Nous proposons une modification de MP qui consiste à changer, à chaque itération, de dictionnaire. Plus précisément, on envisage de décomposer un signal $x \in \mathbb{R}^N$ dans un dictionnaire $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$ très redondant ($M \gg N$), en limitant la sélection à l'itération n à un sous-dictionnaire $\mathbf{\Phi}_{\mathcal{I}^n}$ de taille $|\mathcal{I}^n| = K^n < M$, où $\mathcal{I}^n \subset \{0..M\}$ est un ensemble d'indices de colonnes de $\mathbf{\Phi}$. Ce faisant, on

construit une séquence de sous-ensembles d'indices $\boldsymbol{I} = \{\mathcal{I}^n\}_{n=1..m}$ où m est le nombre d'itérations envisagées. Alternativement, on peut aussi considérer la séquence de sous-dictionnaire $\{\boldsymbol{\Phi}_{\mathcal{I}^n}\}_{n=1..m}$. On appelle cet algorithme : *Matching Pursuit à Séquence de Sous-dictionnaires* (SSMP). En première approximation, chaque itération voit ainsi sa complexité de l'étape de sélection limitée par la taille K^n du sous-dictionnaire.

Il est aisé de voir que cette modification de MP ne touche que l'étape de sélection, et pas celle de mise à jour du résidu. Cette modification peut donc s'appliquer à toutes les variantes sur la mise à jour de MP (OMP, GP, CMP,..). De même, on peut parfaitement envisager certaines variantes du critère de sélection, notamment le Matching Pursuit Moléculaire, à condition que les sous-dictionnaires gardent une structure adéquate.

Nous nous limitons pour l'instant à l'étude de la mise à jour du résidu standard. SSMP construit à partir d'un dictionnaire $\boldsymbol{\Phi}$ et d'une séquence de sous-ensemble d'indices $\boldsymbol{I}$, une approximation d'un signal x en m itérations de la forme :

$$\tilde{x}_m = \sum_{n=1}^{m} \alpha_n \phi_{\gamma^n}^{\mathcal{I}^n} \tag{4.2.1}$$

où

$$\phi_{\gamma^n}^{\mathcal{I}^n} \quad = \quad \mathcal{C}(\boldsymbol{\Phi}_{\mathcal{I}^n}, R^n x) \tag{4.2.2}$$

L'approximation $\tilde{x}_m$ vit dans le sous-espace engendré par l'union des m sous-dictionnaires utilisés :

$$\tilde{x}_m \in \text{span}\left(\bigcup_{n=1}^{m} \boldsymbol{\Phi}_{\mathcal{I}^n}\right) \tag{4.2.3}$$

Dès lors, une stratégie naturelle va consister à choisir des sous-dictionnaires les plus différents deux à deux possible, de manière à maximiser la dimension de ce sous-espace. Même dans le cas de sous-dictionnaires complets ($\forall n, \text{span}(\boldsymbol{\Phi}_{\mathcal{I}^n}) = \mathcal{H}$), nous verrons que cette stratégie est pertinente.

4.2.2 MP à Séquence Aléatoire de Sous-dictionnaires (SASMP)

A première vue, (4.2.2) se rapproche d'un MP Faible, le maximum n'étant cherché que dans un sous-ensemble d'éléments du dictionnaire. La particularité de notre approche est que ce sous-ensemble change durant toute la décomposition au lieu d'être fixé. En particulier nous proposons d'utiliser une séquence pseudo-aléatoire, prédéterminée et donc indépendante du signal x. Cette variante de SSMP est appelée *Matching Pursuit à Séquence Aléatoire de Sous-dictionnaires* (SASMP) par la suite.

Dans la suite et sauf mention contraire, on étudiera explicitement SASMP avec le formalisme suivant. On pose $\boldsymbol{\Phi}$ le dictionnaire complet, la séquence de sous-dictionnaires $\{\boldsymbol{\Phi}_{\mathcal{I}^n}\}_{n=1..m}$ est construite en tirant à chaque itération un ensemble de K^n indices parmi les M possibles.

Algorithme Nous l'avons vu, seule l'étape de sélection est modifiée, ce qui permet d'envisager toutes sortes de variantes de MP avec des séquences de sous-dictionnaires (le cas des poursuites à sélection de sous-espace est néanmoins un peu différent). Le pseudo-code générique est donné ci dessous et illustré Figure 4.2.1.

Algorithm 3 Poursuite à Séquence de Sous-dictionnaires (SSMP)

Entrées: x , $\mathcal{D}$, $\boldsymbol{I}$

1: $R^0x := x$, $\tilde{x}_0 := 0$, $\Gamma^0 = \emptyset$, $n = 1$
2: **Répéter**
3: **Etape 1** : Selection atome dans sous-dictionnaire $\gamma_n \in \mathbf{\Phi}_{\mathcal{I}^n}$:
$\phi^{\mathcal{I}^n}_{\gamma_n} \leftarrow \mathcal{C}(\mathbf{\Phi}_{\mathcal{I}^n}, R^{n-1}x)$
$\Gamma^n \leftarrow \Gamma^{n-1} \cup \gamma_n$
4: **Etape 2** : Mise à jour de l'approximation et du résiduel :
$\tilde{x}_n \leftarrow \mathcal{A}(x, \mathcal{D}_{\Gamma^n})$
$R^nx \leftarrow x - \tilde{x}_n$
5: **Jusqu'à** ce qu'une condition d'arrêt soit remplie

Sorties: $\tilde{x}_n$, R^nx

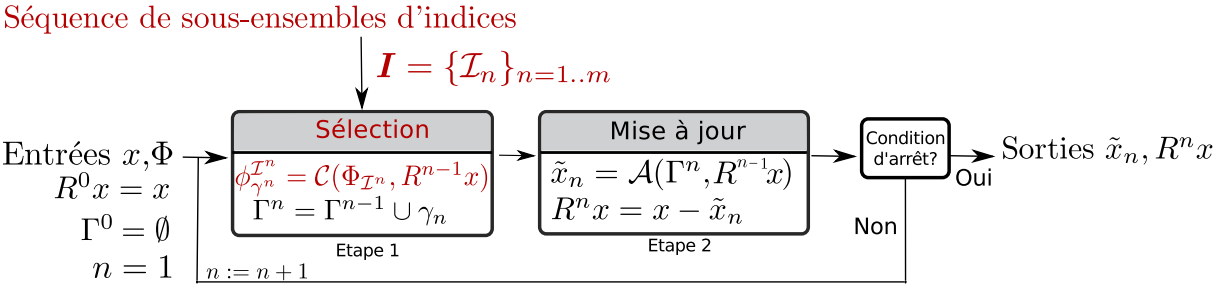

FIGURE 4.2.1: Diagramme de l'algorithme MP à séquence de sous-dictionnaires.

4.2.3 Simulations

Il est aisé de montrer l'intérêt que présente le fait de changer de sous-dictionnaire à chaque itération sur des exemples synthétiques en réalisant des expériences simples. Le code Matlab® permettant de reproduire ces expériences est disponible en ligne sur la page de l'article *Signal Processing*[2].

Les paramètres de cette simulation sont les suivants :

- Signal : $\boldsymbol{x} \in \mathbb{R}^N$ est une réalisation d'un bruit blanc gaussien centré ($\forall i \leq N, x_i \sim \mathcal{N}(0, \sigma^2)$)
- Dictionnaire : $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$ avec $M > N$, une matrice à entrées gaussiennes, dont les colonnes sont normalisées

On cherche alors à comparer trois stratégies :

MP-$\mathbf{\Phi}$: une poursuite utilisant le dictionnaire complet

MP-$\mathbf{\Phi}_{\mathcal{I}^0}$: une poursuite utilisant un sous-dictionnaire $\mathbf{\Phi}_{\mathcal{I}^0}$ fixe de taille K^0

MP-$\{\mathbf{\Phi}_{\mathcal{I}^n}\}$: une poursuite utilisant à chaque itération un sous-dictionnaire aléatoire de taille K^n

Pour simplifier les notations, on considère des sous-dictionnaires de taille fixe ($\forall n, K^n = K$). Les performances de ces approches sont mesurées en terme d'erreur de reconstruction normalisées :

$$\epsilon(n) = 10 \log_{10} \frac{\|\tilde{x}_n - x\|_2^2}{\|x\|_2^2} \tag{4.2.4}$$

la Figure 4.2.2 montre le type de résultats obtenus (en moyenne sur 1000 simulations) avec $N = 64$, $M = 256$ et $K = 64$ pour deux règles de mises à jours (MP simple ou Orthogonal). Les poursuites les plus efficaces (en termes de rapidité de décroissance de l'erreur de reconstruction) sont celles utilisant le dictionnaire $\mathbf{\Phi}$ complet. Les poursuites sur un sous-dictionnaire fixe $\mathbf{\Phi}_{\mathcal{I}^0}$ convergent le plus

2. http ://www.tsi.telecom-paristech.fr/aao/ ?p=531

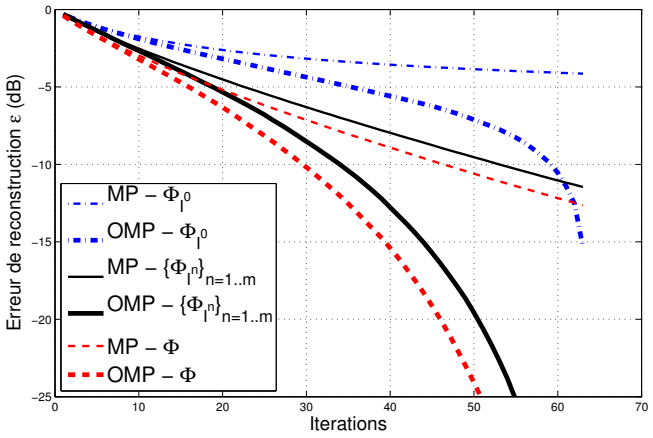

FIGURE 4.2.2: Courbe de décroissance de l'erreur d'approximation pour un cas synthétique avec trois stratégies de sélection et deux règles de mises à jour (MP, OMP).

lentement. Les poursuites utilisant une séquence de sous-dictionnaires $\{\mathbf{\Phi}_{\mathcal{I}^n}\}$ ont un profil de convergence intermédiaire. A nombre d'itération équivalent, ces poursuites permettent une décroissance plus importante qu'avec un sous-dictionnaire fixe et légèrement plus faible qu'en utilisant le dictionnaire complet. En revanche, en termes de rapidité d'éxecution, la poursuite utilisant le dictionnaire complet est beaucoup plus lente que les deux autres. Nous verrons en section 4.4.3 une évaluation plus précise de cet état de fait.

4.2.4 Convergence et stabilité

SSMP est une instance de Matching Pursuit Faible. A chaque itération n, un atome sous-optimal ϕ_{γ^n} est sélectionné :

$$|\langle R^n x, \phi_{\gamma^n}\rangle| \geq t_n \max_{\phi\in\mathbf{\Phi}} |\langle R^n x, \phi\rangle| \tag{4.2.5}$$

et le facteur de sous-optimalité t_n dépend du sous-dictionnaire choisi :

$$t_n = \frac{\max_{\phi\in\mathbf{\Phi}_{\mathcal{I}^n}} |\langle R^n x, \phi\rangle|}{\max_{\phi\in\mathbf{\Phi}} |\langle R^n x, \phi\rangle|} \leq 1 \tag{4.2.6}$$

TEMLYAKOV [Tem02] a montré qu'une condition suffisante de convergence dans ce cas de figure est :

$$\sum_{n=1}^{\infty} \frac{t_n}{n} = \infty$$

GRIBONVAL et NIELSEN [GN01] étendent ce résultat au cas de calculs approchés et montrent que cette condition est nécessaire. Nous voyons qu'il est possible de contrôler le paramètre t_n en choisissant judicieusement la taille et/ou la structure des sous-dictionnaires de façon à maintenir :

$$\sum_{n=1}^{\infty} \frac{1}{n} \frac{\max_{\phi\in\mathbf{\Phi}_{\mathcal{I}^n}} |\langle R^n x, \phi\rangle|}{\max_{\phi\in\mathbf{\Phi}} |\langle R^n x, \phi\rangle|} = \infty \tag{4.2.7}$$

On peut récrire :

$$\begin{aligned} t_n &= \frac{\max_{\phi\in\mathbf{\Phi}_{\mathcal{I}^n}} |\langle R^n x, \phi\rangle|}{\max_{\phi\in\mathbf{\Phi}} |\langle R^n x, \phi\rangle|} \\ &= \frac{\mu(R^n x, \mathbf{\Phi}_{\mathcal{I}^n})}{\mu(R^n x, \mathbf{\Phi})} \end{aligned}$$

où $\mu(x, \mathbf{\Phi})$ est définit comme en Section 3.1.2 page 42 par (3.1.6). Le pire des cas de figure serait que le résiduel soit parfaitement cohérent dans $\mathbf{\Phi}$ (*c-à-d.* $\mu(R^n x, \mathbf{\Phi}) = 1$) et le plus incohérent possible dans $\mathbf{\Phi}_{\mathcal{I}^n}$ (*c-à-d.* $\mu(R^n x, \mathbf{\Phi}_{\mathcal{I}^n}) = \mu_{inf}(\mathbf{\Phi}_{\mathcal{I}^n})$). Dès lors on peut borner t_n :

$$t_n \geq \mu_{inf}(\mathbf{\Phi}_{\mathcal{I}^n}) \tag{4.2.8}$$

Une condition suffisante évidente de convergence du SSMP est donc, en dimension N finie, de ne prendre que des sous-dictionnaires contenant une base orthonormale de $\mathcal{H}$, ce qui garantirait $\mu_{inf}(\mathbf{\Phi}_{\mathcal{I}^n}) \geq \frac{1}{\sqrt{N}}$. En revanche, sans plus d'hypothèses sur la séquence de sous-dictionnaires envisagée, il est malaisé de chercher à définir une condition nécessaire.

Proposition. *Soit* $\{\mathbf{\Phi}_{\mathcal{I}^n}\}_{n=1..+\infty}$ *une séquence de dictionnaires. SSMP converge si :*

$$\sum_{n=1}^{\infty} \frac{\mu_{inf}(\mathbf{\Phi}_{\mathcal{I}^n})}{n} = \infty$$

Dans le cas du SASMP, le cadre privilégié d'étude de la convergence est probabiliste. Nous proposons une modélisation originale utilisant les statistiques d'ordre en annexe A de ce travail.

Stabilité La stabilité de SSMP est difficile à étudier. A première vue, SSMP (et plus encore SASMP) ne semble pas être un algorithme pertinent pour des problèmes de recouvrement parcimonieux. En effet une condition nécessaire pour garantir que SSMP choisisse à chaque itération un atome appartenant effectivement au support idéal est que le sous-dictionnaire contienne au moins l'un de ces atomes.

Le sous-échantillonnage aléatoire du dictionnaire signifie qu'à chaque itération la probabilité que le sous-dictionnaire choisi ne contienne aucun des atomes appartenant au support (ou à la meilleure approximation possible) n'est pas nulle. En particulier, trouver un équivalent de la condition de recouvrement exact proposée par Tropp [Tro04] s'avère problématique dans le cas général.

Pour s'en convaincre, on peut tracer les diagrammes de transition de phase pour le cas suivant (Figure 4.2.3). On opère comme en [DT10] une recherche de solution parcimonieuse au problème $y = Ax$, $y \in \mathbb{R}^N$, $x \in \mathbb{R}^M$ et $A \in \mathbb{R}^{N \times M}$, pour lequel on sait qu'il existe une solution x_0 avec $\|x_0\|_0 = k \leq M$. A est appelée matrice de mesure, elle est construite en tirant aléatoirement N lignes dans une matrice de Fourier de taille $M \times M$. Pour différents ratio de sous-échantillonnage $\delta = N/M$ et de parcimonie $\rho = k/M$ on mesure la proportion de succès – le recouvrement du support exact en k itérations – sur 100 tirages.

4.3 SSMP dans des dictionnaires temps-fréquence

Dans le cas particulier des dictionnaires structurés, il est intéressant de construire des sous-dictionnaires eux-mêmes structurés. On peut alors comprendre chaque sous-dictionnaire de la séquence comme un sous-ensemble de vecteurs $\{\phi_i^{\mathcal{I}^n}\}_{i=1..K^n}$ issus d'un repère redondant $\{\phi_j\}_{j=1..M}$. Une façon simple de construire une séquence est alors de choisir un sous-ensemble $\{\phi_i^{\mathcal{I}^0}\}_{i=1..K^0}$ et de construire les autres par une transformation géométrique simple, analogue à une translation ou une rotation.

4.3.1 MP dans des sous-dictionnaires rotatifs

Il est aisé de visualiser l'effet du sous-échantillonnage de dictionnaire dans un plan 2D. La Figure 4.3.1 illustre les différents sous-échantillonnages possibles d'un dictionnaire très redondant et structuré.

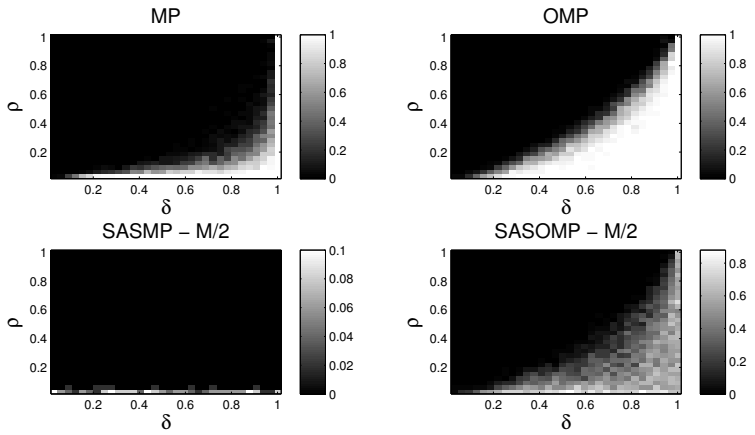

FIGURE 4.2.3: Diagrammes de transition de phase pour un problème de recouvrement parcimonieux pour différents algorithmes. On voit qu'ici le sous-échantillonnage des colonnes de la matrice de mesure réduit fortement les performances.

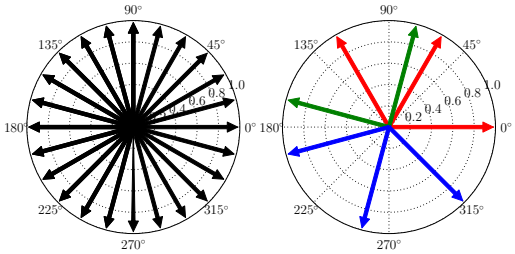

FIGURE 4.3.1: Sous-échantillonnage de dictionnaires structurés : exemple 2D. Gauche : dictionnaire redondant, Droite : en vert exemple de sous-échantillonnage minimal (repère orthonormal). Rouge et Bleu : exemples de sous-dictionnaires redondants.

Une repère ajusté du plan est figuré en vert : il s'agit du cas particulier d'une base orthonormale. En rouge et bleu, deux autres sous-échantillonnages du dictionnaire. Dans ce cadre, il est possible de paramétrer SSMP à l'aide d'une séquence d'angles $\{\theta_i\}_{i=1..m}$ et d'un sous-dictionnaire initial $\mathbf{\Phi}_{\mathcal{I}^0} \subset \mathbf{\Phi}$. A l'itération n $(0 \leq n < m)$, le sous-dictionnaire $\mathbf{\Phi}_{\mathcal{I}^n}$ est une rotation de $\mathbf{\Phi}_{\mathcal{I}^0}$ d'angle θ_n.

Bien sûr en 2D cette technique n'a pas grand intérêt. Lorsque la dimension N devient grande, en revanche, cette rotation va permettre de retirer une partie du biais lié au choix arbitraire de $\mathbf{\Phi}_{\mathcal{I}^0}$. En particulier, la décomposition obtenue vit dans un dictionnaire beaucoup plus grand $\bigcup_{n=1}^{m} \mathbf{\Phi}_{\mathcal{I}^n}$.

4.3.2 Sous-échantillonnage de dictionnaires de Gabor

Chaque atome d'un dictionnaire de Gabor discret peut être identifié par la donnée du triplet (L, u, k) où L est la longueur (échelle) de l'atome, u sa localisation temporelle et k sa localisation fréquentielle. Nous avons vu en 2.1.4 page 25, comment construire un repère de Gabor à partir d'une base de Fourier discrète $\{e^{j2\pi nk/L}\}_{0\leq k\leq L} \in \mathbb{C}^L$. Reprenons la définition des atomes (2.2.14), posons $\mathbf{\Phi}_{L,\Delta_u,\Delta_k}$ le dictionnaire construit à partir des atomes d'échelle L, localisés temporellement tous les Δ_u et fréquentiellement tous les $\frac{2\pi\Delta_k}{L}$. $\mathbf{\Phi}_{L,\Delta_u,\Delta_k}$ définit un pavage du plan temps-fréquence dont la finesse est paramétrée par le couple (Δ_u, Δ_k). Cette structure particulière permet d'envisager des

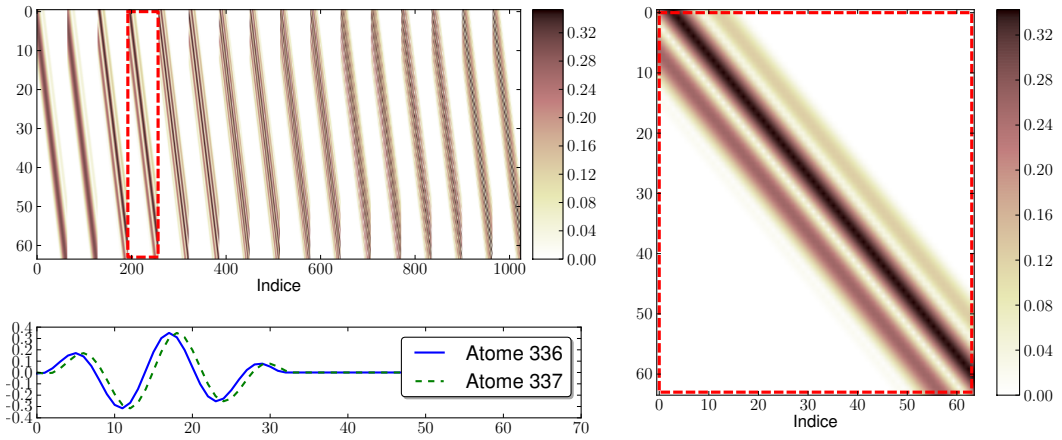

FIGURE 4.3.2: Haut : Visualisation d'un dictionnaire de Gabor mono-échelle $\Phi_{L,1,1}$ ($L = 32$) construisant un repère de $\mathbb{R}^N$ avec $N = 64$ (On n'affiche ici que les atomes des $L/2$ premiers bins fréquentiels, les $L/2$ suivants étant les conjugués de ceux-ci). Bas : Formes d'ondes de deux atomes (c-à-d. colonnes) successifs de $\Phi_{L,1,1}$. Droite : zoom sur la 4e sous matrice, la structure de Toeplitz apparaît clairement.

calculs rapides de projections.

Pavage le plus fin En théorie, le pavage peut être rendu arbitrairement fin (et le dictionnaire arbitrairement grand) en fixant (Δ_u, Δ_k) – notamment en fréquence avec du *zero-padding*. Néanmoins, pour des raisons pratiques (accélération des calculs de projections à l'aide de transformées rapides) il est intéressant de se limiter au cas discret $(\Delta_u, \Delta_k) \in (\mathbb{N}^*)^2$. Dès lors il existe un pavage le plus fin, obtenu pour le couple $(\Delta_u, \Delta_k) = (1, 1)$. On note $\mathbf{\Phi}_{L,1,1}$ le dictionnaire réalisant ce pavage. Il peut se comprendre comme la collection de tous les atomes du type considérés (*p.ex.* Gabor, MDCT) localisés en temps et en fréquence sur des valeurs entières $(u, k) \in [0..N-1] \times [0..L-1]$. Sous forme matricielle, $\mathbf{\Phi}_{L,1,1}$ est composé de NL colonnes et de N lignes.

La projection d'un signal $x \in \mathbb{R}^N$ dans un tel dictionnaire a une complexité de $\mathcal{O}(N^2 L)$. En utilisant la structure du dictionnaire (p.ex. dans le cas de dictionnaires de Gabor, ou MDCT) le calcul se ramène à celui de N transformées de Fourier de taille L, soit en utilisant des transformées rapides une complexité de $\mathcal{O}(NL \log L)$. En rangeant les colonnes de façon adéquate, on peut montrer que $\mathbf{\Phi}_{L,1,1}$ est une concaténation de $L+1$ matrices de Toeplitz :

$$\mathbf{\Phi}_{L,1,1} = \left[\mathbf{\Phi}_L^0 \mathbf{\Phi}_L^1 \dots \mathbf{\Phi}_L^L\right] \tag{4.3.1}$$

où $\mathbf{\Phi}_L^k = [\phi_{1,k}, \phi_{2,k}, \dots, \phi_{u,k}]$ est une matrice dont la u-ième colonne ($0 \leq u < N$) est l'atome de Gabor :

$$\phi_{L,u,k}^{Gabor}[n] = w\left[\frac{n-u}{L}\right] \cdot \exp\left(\frac{2i\pi kn}{L}\right) \tag{4.3.2}$$

La Figure 4.3.2 présente un tel dictionnaire pour un cas de dimensions réduites ($L = 32, N = 64$).

Une décomposition dans $\mathbf{\Phi}_{L,1,1}$ permet de sélectionner des composants très bien localisés dans le plan temps-fréquence. Mais pour des scènes sonores de seulement quelques secondes, la dimension implique une taille très importante de dictionnaire et rend le calcul de projections sur $\Phi_{L,1,1}$ fastidieux. Typiquement le choix est donc fait de sous-échantillonner le dictionnaire $\mathbf{\Phi}_{L,1,1}$.

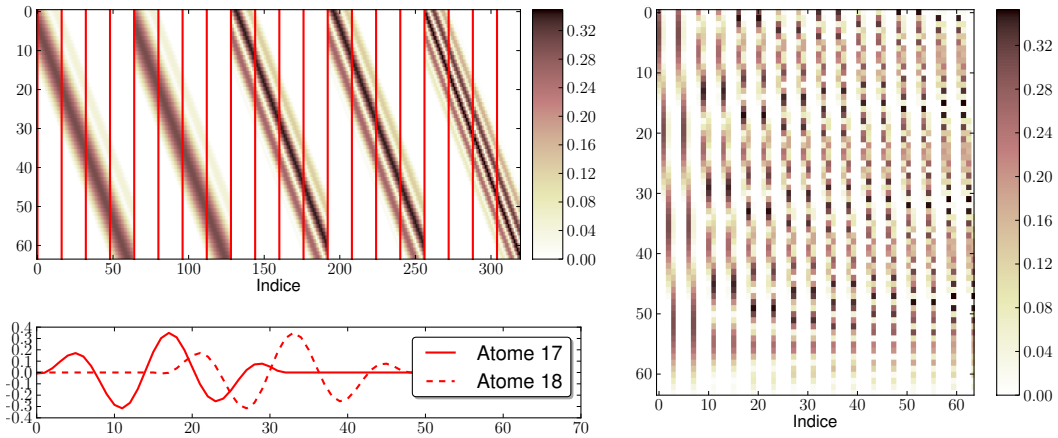

FIGURE 4.3.3: Haut : Dictionnaire de Gabor mono-échelle $\Phi_{L,1,1}$ ($L = 32$, zoom sur les 320 premières colonnes) construisant un repère de $\mathbb{R}^N$ avec $N = 64$, les barres verticales figure le pas de sous-échantillonnage de $\Delta_u = L/2$. Bas : Formes d'ondes de deux atomes (c-à-d. colonnes) successifs de $\Phi_{L,\frac{L}{2},1}$. Droite : sous-dictionnaire $\Phi^1_{L,\frac{L}{2},1}$ obtenu par sous-échantillonnage en commençant sur la sur la première colonne.

Sous-échantillonnage en fréquence En posant $\Delta_k > 1$, on crée un dictionnaire $\mathbf{\Phi}_{L,1,\Delta_k} \subset \mathbf{\Phi}_{L,1,1}$ de taille $N \times N\frac{L}{\Delta_k}$. On peut espérer ainsi accélérer les calculs des projections au prix d'une perte de précision sur la localisation fréquentielle des composantes d'une scène sonore. En pratique, ce choix s'avère contre-productif. Dans toutes les expériences que nous avons menées, sous-échantillonner en fréquence détériore les performances d'approximation et n'accélère que très peu les calculs. Cela s'explique par deux considérations :

1. La dimension temporelle des signaux (N) est souvent grande devant l'échelle L des atomes. La complexité des projections est donc dominée par la dépendance linéaire en N. Le gain en complexité du sous-échantillonnage en fréquence est donc faible.

2. Les scènes sonores en général et la musique en particulier sont composées en grande partie d'éléments harmoniques. La qualité de la reconstruction globale dépend fortement de la qualité de reconstruction de ces composantes, elle-même fonction de la résolution fréquentielle du dictionnaire.

En conséquence, nous poserons dans toute la suite $\Delta_k = 1$. Pour simplifier les notations, on écrira $\mathbf{\Phi}_{L,\Delta_u,1} \to \mathbf{\Phi}_{L,\Delta_u}$.

Sous-échantillonnage en temps Un choix classique d'avancement (par exemple pour la TFCT et la MDCT) est une fraction de la taille de la fenêtre d'analyse, soit $\Delta_u = \frac{L}{Q}$ (typiquement $Q = 2$ ou 4). En supposant le nombre de trames $P = \frac{QN}{L}$ entier, la taille du sous-dictionnaire résultant est alors de $N \times PL$ et la complexité du calcul des projections tombe à $\mathcal{O}(PL \log L)$.

Nous pouvons ainsi construire un sous-dictionnaire $\mathbf{\Phi}_{L,\frac{L}{Q}} \subset \mathbf{\Phi}_{L,1}$ en sélectionnant un sous ensemble des colonnes. En considérant $\mathbf{\Phi}_{L,1}$ sous la forme (4.3.1), cette sélection peut être comprise comme l'application d'un peigne de largeur Δ_u sur les colonnes, comme illustré sur la figure 4.3.3.

Les figures 4.3.3 et 4.3.4 présentent deux sous-échantillonnages différents du dictionnaire Φ. Les deux sous-dictionnaires construits sont des repères de $\mathbb{R}^N$, équivalents à une rotation près. Alter-

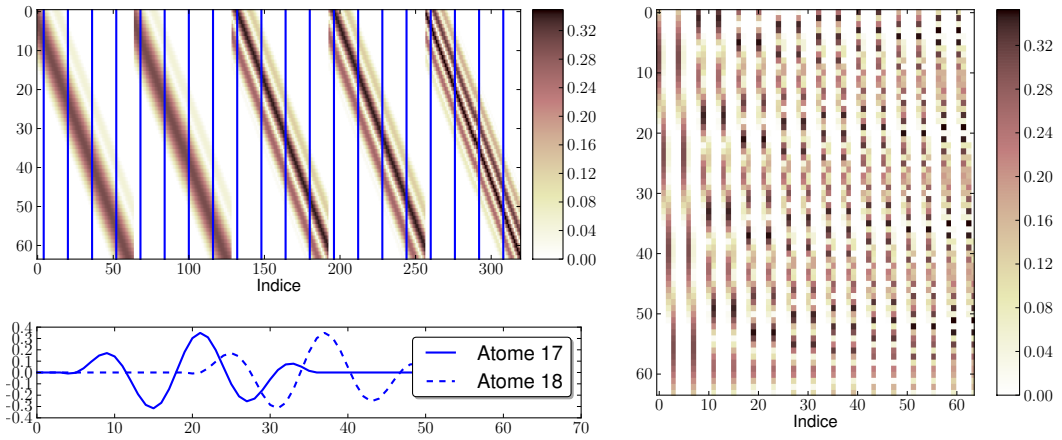

FIGURE 4.3.4: Haut : Dictionnaire de Gabor mono-échelle $\Phi_{L,1,1}$ ($L = 32$, zoom sur les 320 premières colonnes) construisant un repère de $\mathbb{R}^N$ avec $N = 64$, les barres verticales figure le pas de sous-échantillonnage de $\Delta_u = L/2$. Bas : Formes d'ondes de deux atomes (c-à-d. colonnes) successifs de $\Phi_{L,\frac{L}{2},1}$.. Droite : sous-dictionnaire $\Phi^4_{L,\frac{L}{2},1}$ obtenu par sous-échantillonnage en commençant sur la quatrième colonne.

nativement, ces deux repères correspondent à deux découpages (ou fenêtrages) possible d'un signal, décalés circulairement d'un paramètre τ.

Paramétrisation Un sous-dictionnaire de $\mathbf{\Phi}_{L,1}$ est donc défini par la donnée d'un pas de sous-échantillonnage Δ_u et d'un décalage initial τ. Posons $\theta = (L, \Delta_u, \tau)$ le jeu de paramètres définissant un sous-dictionnaire. Une séquence de sous-dictionnaires $\{\mathbf{\Phi}_{\theta^n}\}$ est définie par une séquence de paramètres $\{\theta^n\}$. On peut par exemple construire une séquence de sous-dictionnaires de taille K fixe en définissant une séquence de décalages $\Theta = \{\tau_n\}_{n=1..m}$ et en posant :

$$\forall i, \mathbf{\Phi}_{\theta^n} = \mathbf{\Phi}_{(L,\Delta_u,\tau_n)} \tag{4.3.3}$$

On peut alors considérer chaque sous-dictionnaire en fonction du dictionnaire $\mathbf{\Phi}_{(L,\Delta_u,0)} = \{\phi^0_i\}_{i=1..K}$ en remarquant :

$$\forall n, \mathbf{\Phi}_{(L,\Delta_u,\tau_n)} = \{\phi^n_i\}_{i=1..K} = \{\phi^0_i \otimes \delta_{\tau_n}\}_{i=1..K} \tag{4.3.4}$$

où $\otimes$ dénote la convolution circulaire.

Dans le cas d'une union de dictionnaires, on peut définir le dictionnaire fin multi-échelle :

$$\mathbf{\Phi} = \bigcup_{s=1}^{S} \mathbf{\Phi}_{L_s,1} \tag{4.3.5}$$

qui contient pour chaque échelle s tous les atomes (*p.ex.* de Gabor, ou MDCT) de taille L_s localisés sur des *bins* temps-fréquence (u, k) entiers. $\mathbf{\Phi}$ contient donc $N \cdot \left(\sum_{s=1}^{S} L_s/2\right)$ atomes, et définit un pavage très fin et très redondant du plan temps-fréquence. Un sous-dictionnaire de $\mathbf{\Phi}$ peut être construit comme une union de sous-dictionnaires :

$$\mathbf{\Phi}_{\mathcal{I}^n} = \bigcup_{s=1}^{S} \mathbf{\Phi}_{(L_s,\Delta^s_u,\tau^s_n)} \tag{4.3.6}$$

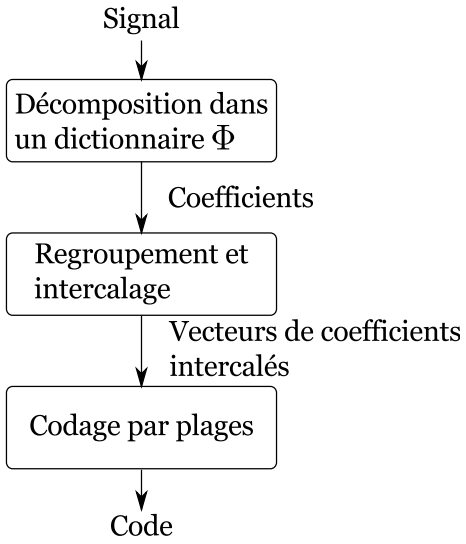

FIGURE 4.4.1: Schéma-bloc d'un codeur audio basé sur une représentation parcimonieuse [RRD08]

où un pas de sous-échantillonnage temporel Δ_u^s est défini pour chaque échelle (*p.ex.* en posant $\Delta_u^s = \frac{L_s}{Q}$). La paramétrisation complète de la séquence de sous-dictionnaires revient donc à définir la séquence multiple :

$$\mathbf{\Theta} = \{\tau_n^s\}_{n=1..m, s=1..S} \tag{4.3.7}$$

Le choix de la séquence $\mathbf{\Theta}$ détermine en grande partie les performances du SSMP. On pourrait par exemple imposer des décalages communs pour les sous-dictionnaires d'échelles différentes ($\forall s, \tau_n^s = \tau_n$), mais l'expérience montre que la décomposition gagne à ce que les décalages soient asynchrones. Sur chaque échelle, on modélise les décalages par une variable aléatoire discrète uniforme τ^s.

$$\forall s, \tau^s \sim \mathcal{U}(0, \Delta_u^s - 1) \tag{4.3.8}$$

A première vue, $\mathbf{\Theta}$ est un paramètre supplémentaire à transmettre pour pouvoir effectuer la reconstruction d'une approximation. L'idée, dans un contexte de compression, est d'envisager des séquences pseudo-aléatoires, connues à l'avance au codeur et au décodeur, ce qui épargne le coût de transmission, tout en améliorant de façon significative la qualité de l'approximation.

4.4 Application à la compression de scènes sonores

4.4.1 Codeur naïf

La compression de scènes sonores par approximations parcimonieuses dans une union de bases MDCT a été étudiée notamment par RAVELLI [RRD08]. Contrairement à lui, nous n'avons pas dans ce travail proposé d'architecture de codage complète (comme présentée Figure 4.4.1), notre travail se concentre sur la première de ces étapes : la décomposition elle-même.

Le schéma de l'architecture considérée est plus naïf, il est présenté Figure 4.4.2. En couleur la partie qui nous intéresse plus spécialement, celle de l'algorithme de décomposition. Nous avons comparé pour ce bloc, trois stratégies gloutonnes :

MP $\mathbf{\Phi} - \Delta_u^s = L_s/2$ Un Matching Pursuit sur un dictionnaire fixe de pavage grossier (sous-dictionnaire de $\mathbf{\Phi}$).

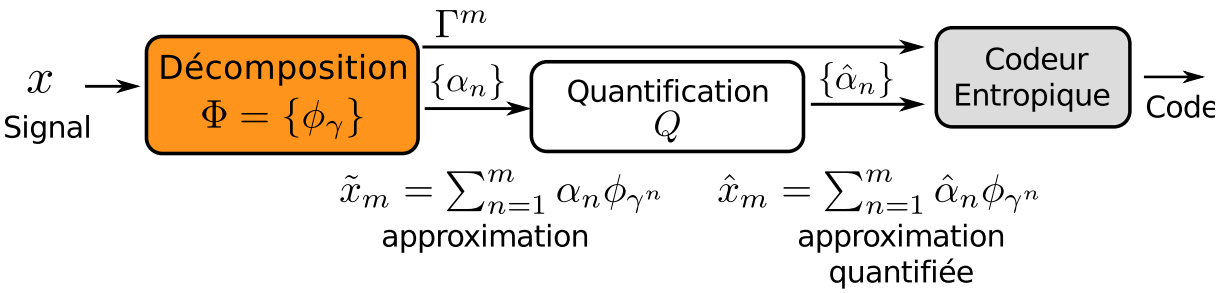

FIGURE 4.4.2: Diagramme par blocs de l'architecture naïve de codage audio.

MPA $\mathbf{\Phi} - \Delta_u^s = 1$ Un Matching Pursuit Adaptatif sur le dictionnaire de pavage le plus fin avec recherche préalable dans un sous dictionnaire fixe de pavage grossier ($\Delta_u^s = L_s/2$, le même que pour le cas précédent) puis optimisation locale dans un voisinage. Cette optimisation se traduit par une variable supplémentaire (un décalage) nécessaire à la définition de chaque atome, soit au final, le besoin de transmettre une séquence $\mathbf{\Theta}$ de paramètres supplémentaires.

SASMP $\mathbf{\Phi} - \Delta_u^s = L_s/2$ Un Matching Pursuit sur une séquence aléatoire de sous-dictionnaires de pavage grossier. La séquence de décalages $\mathbf{\Theta}$ qui paramétrise les sous-dictionnaires est indépendante du signal et supposée connue à l'avance tant au codeur qu'au décodeur.

Ces trois stratégies, résumées Figure 4.4.3 ont une complexité, par itération, équivalente. Il est difficile de considérer un MP sur le grand dictionnaire de pavage fin lorsque l'on travaille sur des scènes sonores de plusieurs secondes. Le MP Adaptatif est néanmoins une bonne approximation de ce cas de figure. Dans ce cas, une information complémentaire (par exemple un raffinement de la localisation des atomes) doit être transmise en plus des indices et poids quantifiés. Le pendant de cette information pour la stratégie proposée est la séquence de paramètres des sous-dictionnaires. Dans ce cas néanmoins, cette information peut être connue à l'avance au niveau du codeur et du décodeur, et n'a donc pas besoin d'être transmise.

4.4.2 Résultats

L'implémentation de tous les algorithmes a été entièrement réalisée en Python. Lorsque c'est possible, les calculs de projections sont optimisés à l'aide de transformées rapides. Dans ce cas de figure, nous utilisons la boite à outil FFTW [3] en langage C et une interface python spécifique. Le code source de toutes ces expériences fait l'objet d'une documentation et sera proposé en ligne bientôt. Nous avons déposé une implémentation naïve de SASMP et SASOMP en Matlab sur le serveur central *file exchange* [4]. Une implémentation libre de droits permettant de reproduire une partie de ces résultats sur des données réelles est également disponible en language Python [5].

Codage naïf sur une union de bases MDCT En premier lieu, nous montrons par une expérience simple l'intérêt que présente l'utilisation d'une union de bases MDCT par rapport à une simple base orthonormale, reprenant ainsi les observations déjà formulées dans [RRD08]. La Figure 4.4.4 donne un exemple de comparaison entre des MP sur ces deux dictionnaires, pour un signal de glockenspiel.

3. www.fftw.org
4. http ://www.mathworks.com/matlabcentral/fileexchange/37537-matching-pursuit-with-random-sequential-subdictionaries
5. https ://github.com/mmoussallam/PyMP

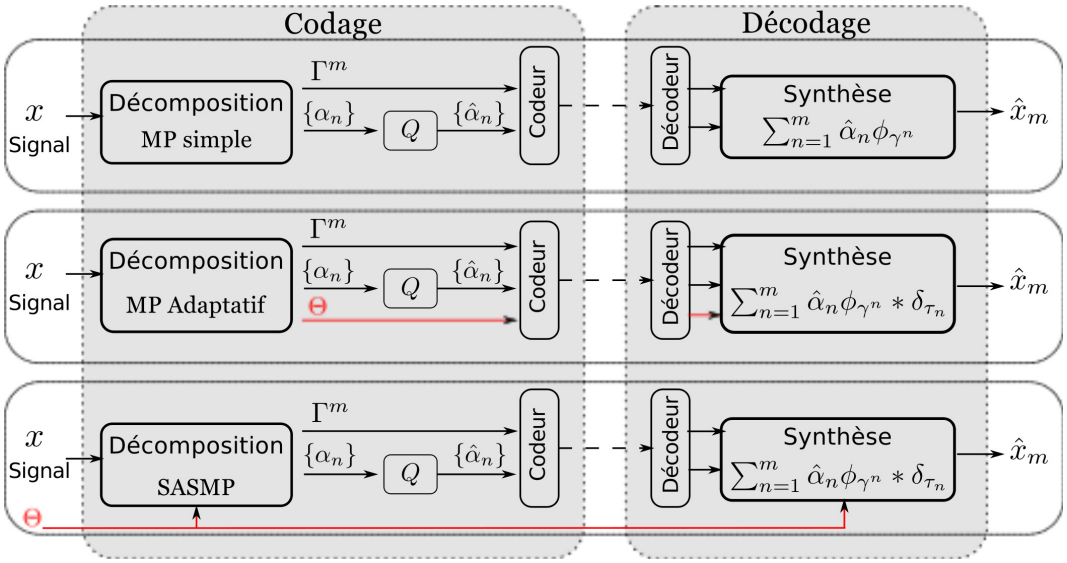

FIGURE 4.4.3: Diagramme de trois stratégies de codage. Décomposition simple avec transmission des indices Γ^m et des coefficients quantifiés $\{\hat{\alpha}_n\}_{n=1..m}$. Décomposition adaptative avec transmission en plus d'une information complémentaire $\boldsymbol{\Theta}$ d'optimisation locales des atomes. Enfin décomposition proposée, sans transmission d'information complémentaire si la séquence pseudo-aléatoire de sous-dictionnaires paramétrée par $\boldsymbol{\Theta}$ est connue au codeur et au décodeur.

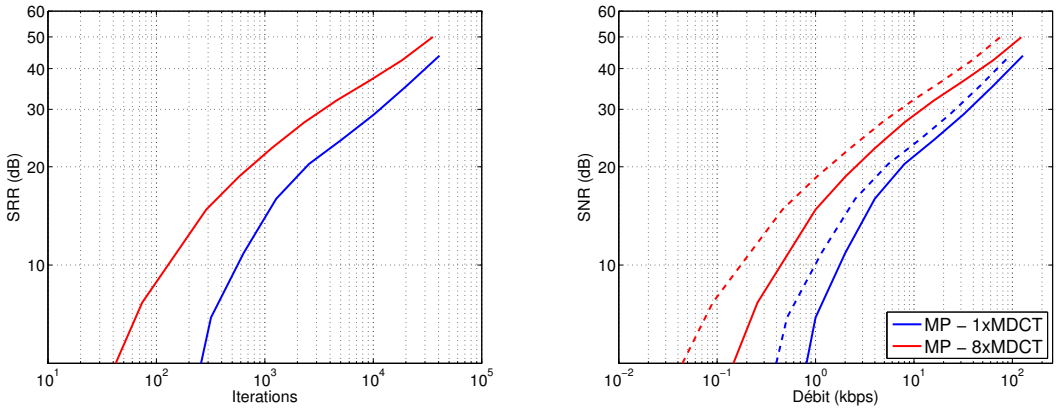

FIGURE 4.4.4: Gauche : évolution du critère de qualité de reconstruction (SRR) en fonction du nombre d'itérations pour un signal de 10 secondes de glockenspiel. Droite : Courbes débit-distorsion obtenues avec une quantification uniforme et un codeur uniforme (trait plein) ou un codeur entropique idéal (tirets).

Dans cet exemple, nous pouvons voir que non seulement l'union de bases MDCT permet d'assurer une décroissance du résiduel (*c-à-d.* une montée du SRR) plus rapide, mais également de meilleures performances en codage à bas débit. En haut-débit l'écart tend à se réduire et les courbes finissent par se croiser. Rappelons ici que le codeur utilisé dans [RRD08] est plus performant que le notre (en particulier en ce qui concerne le codage de source) ce qui explique que les auteurs observent ce croisement à un débit inférieur au notre (environ 128 Kbps).

RAVELLI *et al* ont observé ce phénomène et proposent un codeur adaptatif qui passe d'une union de 8 bases à une seule base MDCT en haut débit. Dans ce travail nous ne proposons pas un tel dispositif

et nous concentrerons sur l'étude des performances à bas-débit (*c-à-d.* <128 Kbps). La Figure 4.4.4 permet en outre de voir que l'utilisation d'un codeur entropique idéal (courbes en tirets) ne change pas fondamentalement la nature de la comparaison.

Comparaison des trois stratégies sur une union de bases MDCT On s'attache désormais à la comparaison des trois stratégies décrites plus haut. Pour cela on utilise un corpus de 4 signaux issus de la base de test *Sound Quality Assessment Material* (SQAM) [MPE03] régulièrement utilisée pour la mesure des performances d'un codeur audio. Il s'agit ici de glockenspiel, de voix d'homme, d'orchestre et de musique populaire. La Figure 4.4.5 montre que pour tous les exemples (et nous n'avons trouvé aucune scène sonore réelle pouvant servir de contre-exemple) SASMP permet, à un débit donné, d'obtenir une qualité significativement meilleure.

Nous voyons également que SASMP sélectionne un nombre d'atomes équivalent au MP Adaptatif et dont la corrélation fine avec le signal est similaire (à l'exception peut-être du premier exemple : le glockenspiel, pour lequel le MP Adaptatif réalise une meilleure décomposition en raison sans doute de la localisation très précise de l'énergie dans le plan temps-fréquence sur cet exemple).

Comparaison des types de séquences Dans ce qui précède, nous avons paramétrisé les séquences de sous-dictionnaires à l'aide d'une séquence de décalage :

$$\mathbf{\Theta} = \{\tau_n^s\}_{n=1..m, s=1..S}$$

On a posé que pour le SASMP, cette séquence était pseudo-aléatoire, tirée selon une loi uniforme discrète :

$$\tau_n^s = \mathcal{U}(0, \Delta_u^S - 1)$$

Mais nous n'avons pas justifié que cette méthode était la plus intéressante. L'intuition derrière ce modèle est que plus les sous-dictionnaires sont variés, plus l'algorithme pourra en tirer profit. Pour confirmer cette intuition, nous avons comparé l'approche ci-dessus avec des SSMP déterministes. Deux séquences déterministes sont construites avec les règles suivantes :

Unitaire les décalages suivent une progression très simple et sont les mêmes pour toutes échelles :

$$\forall s, \begin{cases} \tau_0^s = 0 \\ \tau_n^s = \tau_{n-1}^s + 1 \quad \text{si } n > 0 \end{cases}$$

Sauts les décalages suivent une progression par sauts, différente selon les échelles :

$$\forall s, \begin{cases} \tau_0^s = 0 \\ \tau_n^s = \tau_{n-1}^s + \frac{\Delta_u^s}{2} + 1 \quad \mathrm{mod}\Delta_u^s \text{ si } n > 0 \end{cases}$$

La Figure 4.4.6 présente les résultats obtenus. Nous voyons que la stratégie de décalages unitaires synchrones est la moins performante. La stratégie déterministe par sauts donne des scores assez proches de SASMP, mais toujours inférieurs. Il est important de noter que, même si les performances sont assez proches, SASMP est toujours légèrement meilleur. Nous avons lancé plusieurs exécutions de SASMP, la variance des résultats est très faible et en tout état de cause inférieure à l'écart avec le meilleur SSMP déterministe, et ce pour tous les signaux réels rencontrés.

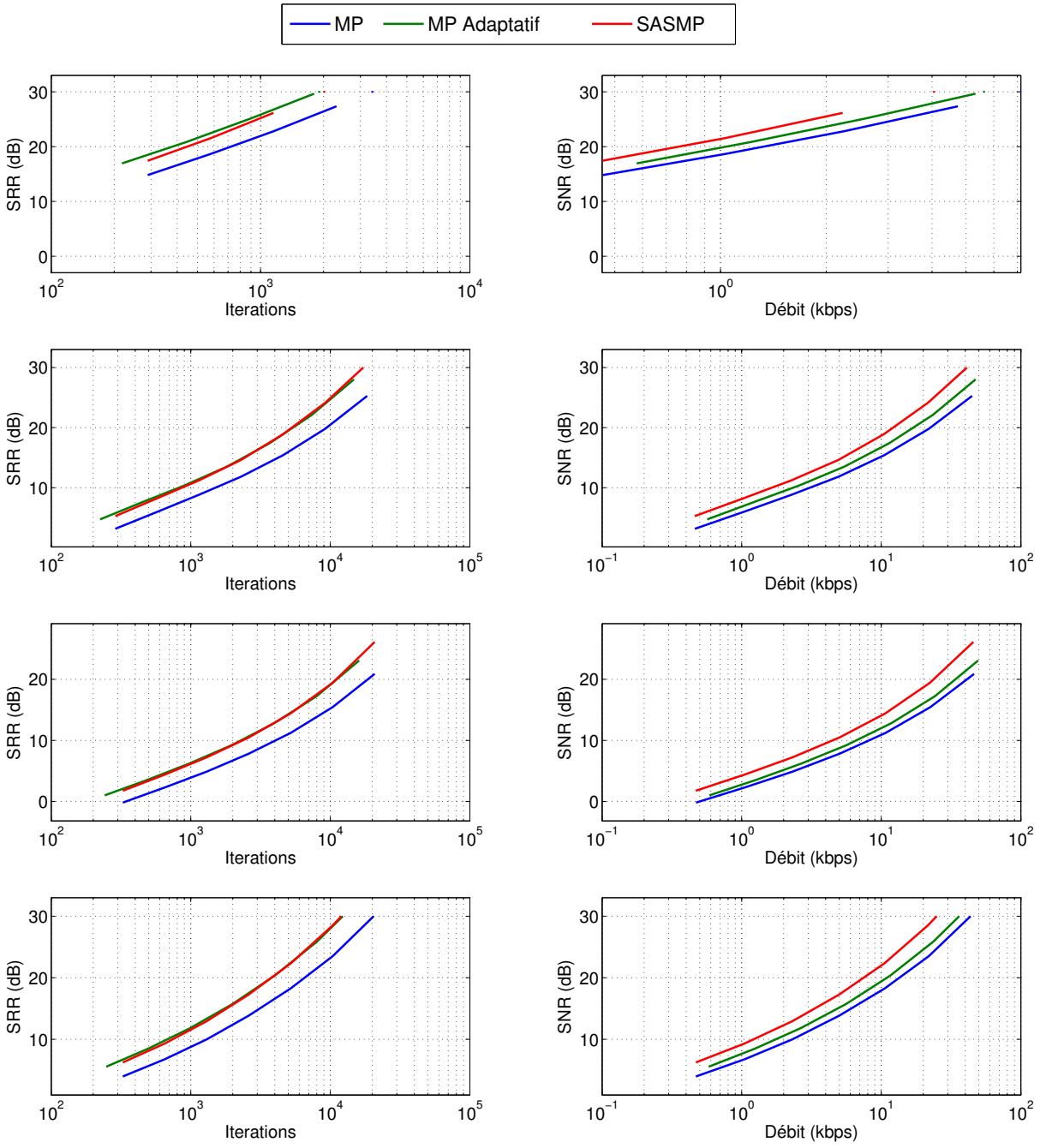

FIGURE 4.4.5: Comparaison de trois stratégies gloutonnes pour l'approximation et le codage de scènes sonores. Gauche : erreur de reconstruction (SRR) en fonction du nombre d'itérations. Droite : courbes débit (théorique)-distorsion correspondantes.. De haut en bas : Glockenspiel, Voix d'homme, Orchestre et Musique populaire.

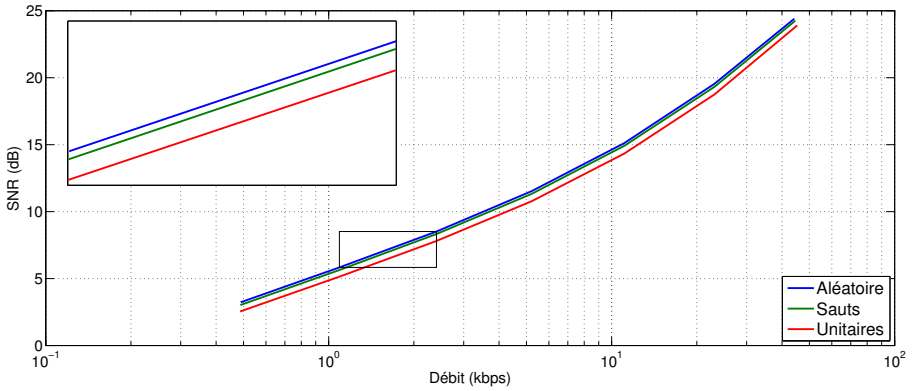

FIGURE 4.4.6: Courbes débit-distorsion obtenues pour SASMP et deux SSMP déterministes. Courbes moyennes sur 5 scènes sonores .

Etape		MP $\mathbf{\Phi}$	MP $\mathbf{\Phi}_{\mathcal{I}^0}$	OMP $\mathbf{\Phi}$	OMP $\mathbf{\Phi}_{\mathcal{I}^0}$	SSMP $\{\mathbf{\Phi}_{\mathcal{I}^n}\}$	SSOMP $\{\mathbf{\Phi}_{\mathcal{I}^n}\}$
1	Projection	$\mathcal{O}(MN)$	$\mathcal{O}(KN)$	$\mathcal{O}(MN)$	$\mathcal{O}(KN)$	$\mathcal{O}(KN)$	$\mathcal{O}(KN)$
	Sélection	$\mathcal{O}(M)$	$\mathcal{O}(K)$	$\mathcal{O}(M)$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	$\mathcal{O}(K)$
2	Gram Matrix	0	0	$\mathcal{O}(nN)$	$\mathcal{O}(nN)$	0	$\mathcal{O}(nN)$
	Coefficients	0	0	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$	0	$\mathcal{O}(n^2)$
	Résiduel	$\mathcal{O}(N)$	$\mathcal{O}(N)$	$\mathcal{O}(nN)$	$\mathcal{O}(nN)$	$\mathcal{O}(N)$	$\mathcal{O}(nN)$
	Total	$\mathcal{O}(MN)$	$\mathcal{O}(KN)$	$\mathcal{O}(n^2+MN)$	$\mathcal{O}(n^2+KN)$	$\mathcal{O}(KN)$	$\mathcal{O}(n^2+KN)$

TABLE 4.4.1: Complexités par étape pour différents algorithmes (sans optimisation autre que la factorisation QR).

4.4.3 Complexités et temps de calcul

Complexité dans le cas général On se limite dans ce paragraphe aux complexités théoriques correspondant aux cas de dictionnaires non structurés, pour lesquels aucune astuce d'optimisation n'est disponible. La complexité se divise en deux composantes, correspondant chacune à une étape : l'étape de sélection et celle de mise à jour.

Soit x dans $\mathbb{R}^N$ et $\mathbf{\Phi}$ un dictionnaire redondant de M atomes. Soit $\{\mathbf{\Phi}_{\mathcal{I}^n}\}$ une séquence de sous-dictionnaires de $\mathbf{\Phi}$ de taille K. A l'itération n , les complexités dans le cas général sont données Table 4.4.1. L'étape de sélection pour un MP standard nécessite le calcul des M projections atomiques dans l'espace de dimension N, suivi d'une recherche de maximum. L'étape de mise à jour pour le MP standard ne requiert qu'une soustraction. Pour des poursuites orthogonales en revanche, la mise à jour nécessite une pseudo-inversion qui implique le calcul d'une matrice de GRAM et le calcul des coefficients. Ces deux sous-étapes additionnelles ont une complexité croissante au fil des itérations. Pour pallier ce problème, des techniques d'accélération ont été proposées [BD08], qui utilisent une factorisation QR dont la mise à jour est un ordre de grandeur plus simple. La Table 4.4.1 tient compte de cette optimisation.

Projection et mise à jour rapide Dans le cas général, SSMP a une complexité équivalente à celle du MP sur un sous-dictionnaire fixe. Mais lorsqu'on utilise des dictionnaires structurés (*p. ex.* temps-fréquence), certaines astuces permettent d'accélérer très significativement les étapes de projection et de mise à jour. La première est proposée dans l'article [MZ93], les projections sont calculées à partir

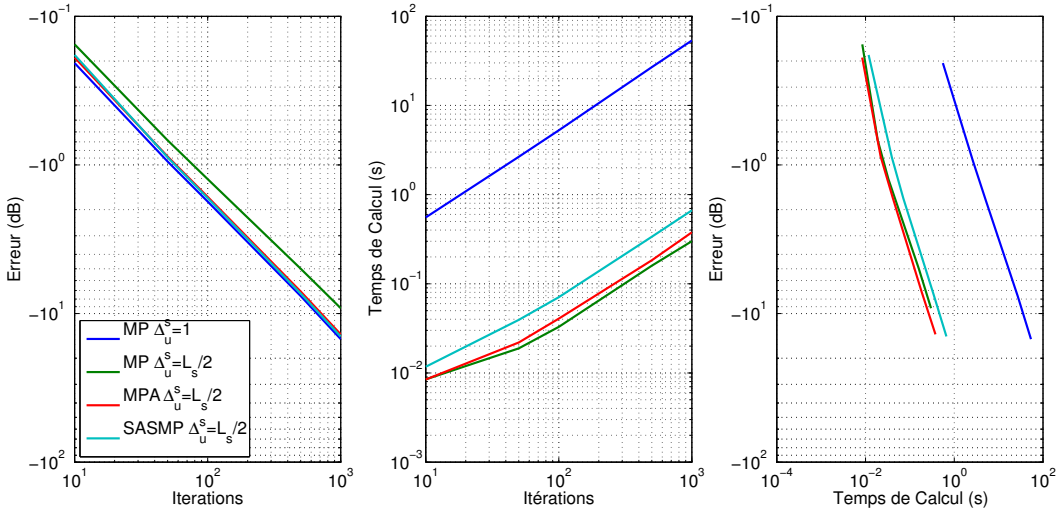

FIGURE 4.4.7: Gauche : erreur quadratique de reconstruction normalisée. Milieu : temps de calcul. Droite : erreur quadratique fonction du temps de calcul. Résultats moyens sur 100 exécutions réalisées sur un Dual Core Intel E8500 à 3.16GHz et 8 Go de RAM.

de celles de l'étape précédente et des produits scalaires entre atomes du dictionnaires :

$$\langle R^n x, \phi_\gamma \rangle = \langle R^{n-1} x, \phi_\gamma \rangle - \langle R^n x, \phi_{\gamma^{n-1}} \rangle \langle \phi_{\gamma^{n-1}}, \phi_\gamma \rangle \tag{4.4.1}$$

où ϕ_{γ^n} est l'atome sélectionné à l'itération n. Les produits $\langle \phi_{\gamma^{n-1}}, \phi_\gamma \rangle$ pouvant être pré-calculés, la complexité totale est grandement réduite. En particulier dans le cas de dictionnaires temps-fréquence, une grande partie de ces produits sont nuls. Cette optimisation nécessite que la projection $\langle R^{n-1} x, \phi_\gamma \rangle$ ait été calculée à l'itération précédente. Pour une poursuite utilisant une séquence de sous-dictionnaires cela implique que $\gamma \in \mathcal{I}^{n-1} \bigcap \mathcal{I}^n$ (*c-à-d.* que l'atome ϕ_γ appartienne aux deux sous-dictionnaires successifs). La plupart du temps cette propriété ne sera pas vérifiée (*p. ex.* le SASMP proposé plus haut sur des dictionnaires temps-fréquence).

Une optimisation importante est l'utilisation de transformées rapides. Si l'on préserve la structure des sous-dictionnaires – comme on l'a fait plus haut – cette optimisation est possible quelque soit l'algorithme. Pour un atome $\phi_{L_s,\gamma}$ de longueur L_s le calcul de $\langle R^n x, \phi_{L_s,\gamma} \rangle$ ne nécessite plus que $\mathcal{O}(L_s \log L_s)$.

Un autre facteur d'accélération est proposé dans l'implémentation MPTK [KG06] et également par MAILHÉ *et al*[MGBV09] dans le cas orthogonal. Les atomes ayant un support temporel limité (*p. ex.* $L_s < N$), à chaque itération seul un sous-ensemble de projections sont affectées par la sélection d'un atome, et de même pour la mise à jour. Il serait là aussi difficile d'adapter les optimisations proposées dans [MGBV09] au cas du SSOMP.

Pour illustrer cet accroissement de complexité, nous pouvons comparer différentes stratégies sur des exemples synthétiques. Nous générons ainsi un bruit blanc gaussien x de dimension N, et nous le décomposons dans un dictionnaire structuré (3 échelles) avec les stratégies suivantes :

MP $\Delta_u^s = 1$ Un MP standard utilisant le dictionnaire multi-échelles de pavage le plus fin.

MP $\Delta_u^s = \frac{L_s}{2}$ Un MP standard utilisant un sous-dictionnaire multi-échelles fixe de pavage déterminé par $\Delta_u^s = \frac{L_s}{2}$.

MPA $\Delta_u^s = \frac{L_s}{2}$ Un MP Adaptatif réalisant une pré-sélection dans un sous-dictionnaire multi-échelles de pavage déterminé par $\Delta_u^s = \frac{L_s}{2}$, puis une optimisation locale.

SASMP $\Delta_u^s = \frac{L_s}{2}$ Un MP avec séquence aléatoire de sous-dictionnaire multi-échelles de pavage déterminé par $\Delta_u^s = \frac{L_s}{2}$

La Figure 4.4.7 résume les résultats obtenus. Ici la dimension est $N = 8192$. On utilise des atomes de Gabor de taille 32,128 et 512 échantillons. On peut faire plusieurs remarques :

- Sur ce cas de figure (signal bruité, incohérent dans le dictionnaire choisi), le SASMP a un profil de convergence moyen meilleur que le MP adaptatif, ce qui est une surprise en soi.
- En revanche, la mise en place de toutes les stratégies d'optimisations listées ci-dessus rend le MP adaptatif plus rapide. On remarque néanmoins que la complexité de SASMP reste deux ordres de grandeur plus faible que celle du MP sur le dictionnaire de pavage fin.

Pour limiter ce surplus de complexité, il est possible de définir des stratégies hybrides, dans lesquels le sous-dictionnaire n'est changé qu'au bout d'un nombre J d'itérations (ce nombre peut d'ailleurs varier au cours de la décomposition). Ainsi durant J itérations les astuces d'accélération décrites plus haut peuvent être mises en place. Nous avons expérimenté ce type de stratégies avec des résultats conformes à l'intuition que l'on peut en avoir : la décomposition est d'autant plus accélérée qu'elle perd en qualité. L'allure de ce compromis est, de plus, très dépendante du signal.

Compromis Compression / Complexité

Une autre manière de réduire la complexité de SSMP est de réduire la taille des sous-dictionnaires utilisés. Cette idée présente en outre l'intérêt de réduire encore plus le coût de codage des atomes. Bien sûr, ce gain en temps de calcul et en codage des indices va être contre-balancé par une distorsion plus importante.

L'allure de ce compromis est présenté Figure 4.4.8. Tous les points de cette figure sont obtenus pour une même valeur de distorsion (10 dB de SNR), le temps de calcul et le débit nécessaire pour atteindre cette valeur sont présentés relativement aux valeurs obtenues avec un MP standard sur un sous-dictionnaire de taille fixe ($\Delta_u^s = L_s/2$). Là encore, on voit que les profils varient grandement entre les signaux. On remarque que, pour les signaux d'orchestre et de voix, il est possible de réduire très fortement la taille du sous-dictionnaire, et donc d'accélérer les calculs tout en maintenant un compromis débit-distorsion relativement stable. En revanche pour le signal de glockenspiel, cette accélération a pour effet une détérioration nette des performances en compression.

Dans cette expérience, nous utilisons des dictionnaires MDCT de 8 échelles différentes (de 4 à 512 ms). Les résultats sont moyennés sur 100 exécutions avec un Dual Core (Intel E8500 3.16 GHz, 8GB RAM). On peut conclure de cette expérience que l'algorithme SASMP présente un gain de performance d'autant plus important que le signal est riche, c'est-à-dire dont l'énergie n'est pas concentrée que sur quelques zones temps-fréquence. Ce dernier type de signaux (dont le signal de glockenspiel offre une bonne illustration) regroupe ceux dont le codage par décomposition sur un dictionnaire redondant d'atomes temps-fréquence est le plus efficace. Par conséquent, on peut souligner le fait que SASMP est surtout intéressant en codage pour les cas de figure les plus difficiles pour les codeurs paramétriques usuels.

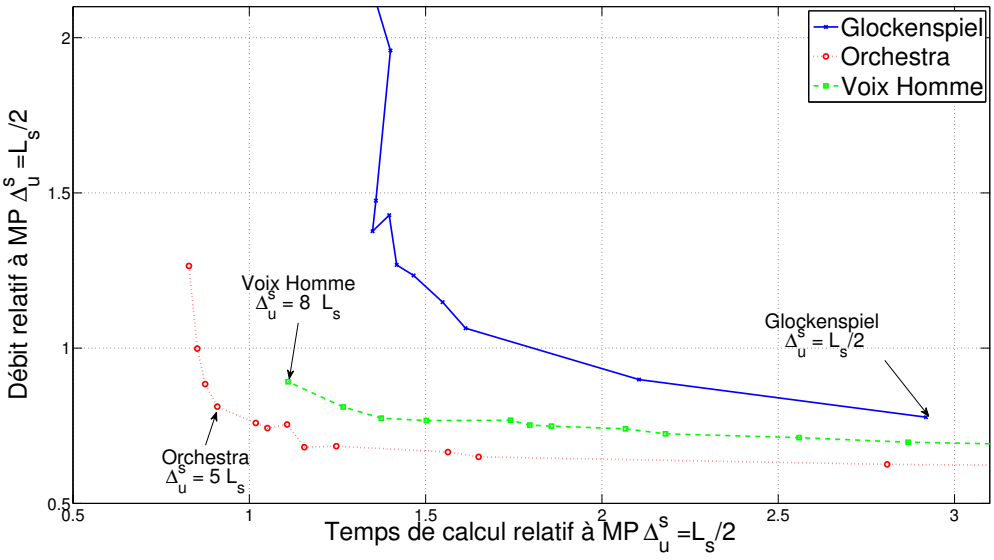

FIGURE 4.4.8: Illustration du compromis entre compression et temps de calcul pour un SASMP sur différentes tailles de sous-dictionnaires. Le taux de compression et le temps de calcul sont donnés relativement au cas de figure MP simple sur un sous-dictionnaire fixe.

Conclusion

Nous proposons dans ce chapitre une variante de l'algorithme *Matching Pursuit* dans laquelle la sélection des atomes s'opère dans des sous-dictionnaires différents à chaque itération. En construisant la séquence de sous-dictionnaires de façon pseudo-aléatoire, de bonnes performances de reconstruction peuvent être atteinte, avec une complexité théorique réduite.

Dans le cas de dictionnaires structurés de type union de bases orthonormales, il est possible de contrôler la séquence de sous-dictionnaires avec un petit nombre de paramètres, par exemple de décalage temporel aléatoire sur les atomes.

En supposant connue à l'avance cette séquence de paramètres, il est possible d'améliorer significativement les performances d'un codeur basé sur la représentation parcimonieuse d'un signal. Nous avons vu, avec un modèle de codeur très simple, que le compromis débit-distorsion est meilleur, quelle que soit la scène sonore considérée.

Chapitre 5

Matching Pursuit Dynamiques

En soi, MP est un processus dynamique puisque la construction de l'approximation se fait itérativement. En revanche, ses paramètres (dictionnaire, critère de sélection, règle de mise à jour) sont généralement fixés durant tout le processus. La variante de MP développée dans le chapitre précédent tire profit du changement régulier du dictionnaire. On peut envisager plus largement la famille d'algorithmes MP dont les paramètres changent au cours d'une exécution sous le nom de *Matching Pursuit Dynamiques*.

Dans ce chapitre, nous essaierons dans un premier temps (section §5.1) de justifier le recours à ce type d'algorithme, en mettant en évidence les variations importantes de nature que subissent les projections des résiduels successifs dans des dictionnaires. Il nous faudra alors proposer des métriques pertinentes pour rendre compte de ces évolutions. Pour cela, nous nous appuierons sur les statistiques d'ordre pour estimer les paramètres de cette évolution.

Dans un deuxième temps (section §5.2) nous verrons comment adapter dynamiquement le sous-échantillonnage pour tirer profit de cette évolution. On peut en effet proposer un algorithme de type SASMP avec une séquence de dictionnaires de taille variable qui permet d'accélérer la convergence tout en maintenant une décomposition efficace, et nous verrons en particulier l'application au codage de scènes sonores.

Dans un dernier temps (section §5.3), nous étendrons le principe du MP dynamique au critère de sélection, et plus spécifiquement au calcul des projections. Nous verrons qu'il est possible de modéliser cet algorithme sous la forme d'un SSMP pour lequel le sous-échantillonnage des colonnes du dictionnaire s'accompagne d'un sous-échantillonnage des lignes. Nous profiterons de l'occasion pour mentionner les travaux de Peel *et al* [PERA12] et les pistes envisagées lors d'une collaboration. Nous ferons également le lien avec des techniques récentes de factorisation matricielle utilisant des sous-échantillonnages aléatoires.

5.1 Évolution des distributions de projections

5.1.1 Mise en évidence

Posons, tout d'abord quelques notations. Le résiduel de la décomposition d'un signal $x \in \mathbb{R}^N$ après de n itérations de MP est noté $R^n x$. Soit $\mathbf{\Phi}$ le dictionnaire composé de M atomes $\phi_i \in \mathbb{R}^N$. La projection de $R^n x$ dans Φ donne M valeurs $\{\langle R^n x, \phi_i \rangle\}_{i=1..M}$. Soit Z une variable aléatoire modélisant ces projections, normalisées par l'énergie du résiduel. On considère alors $\{z_i\}_{i=1..M}$ comme un échantillon

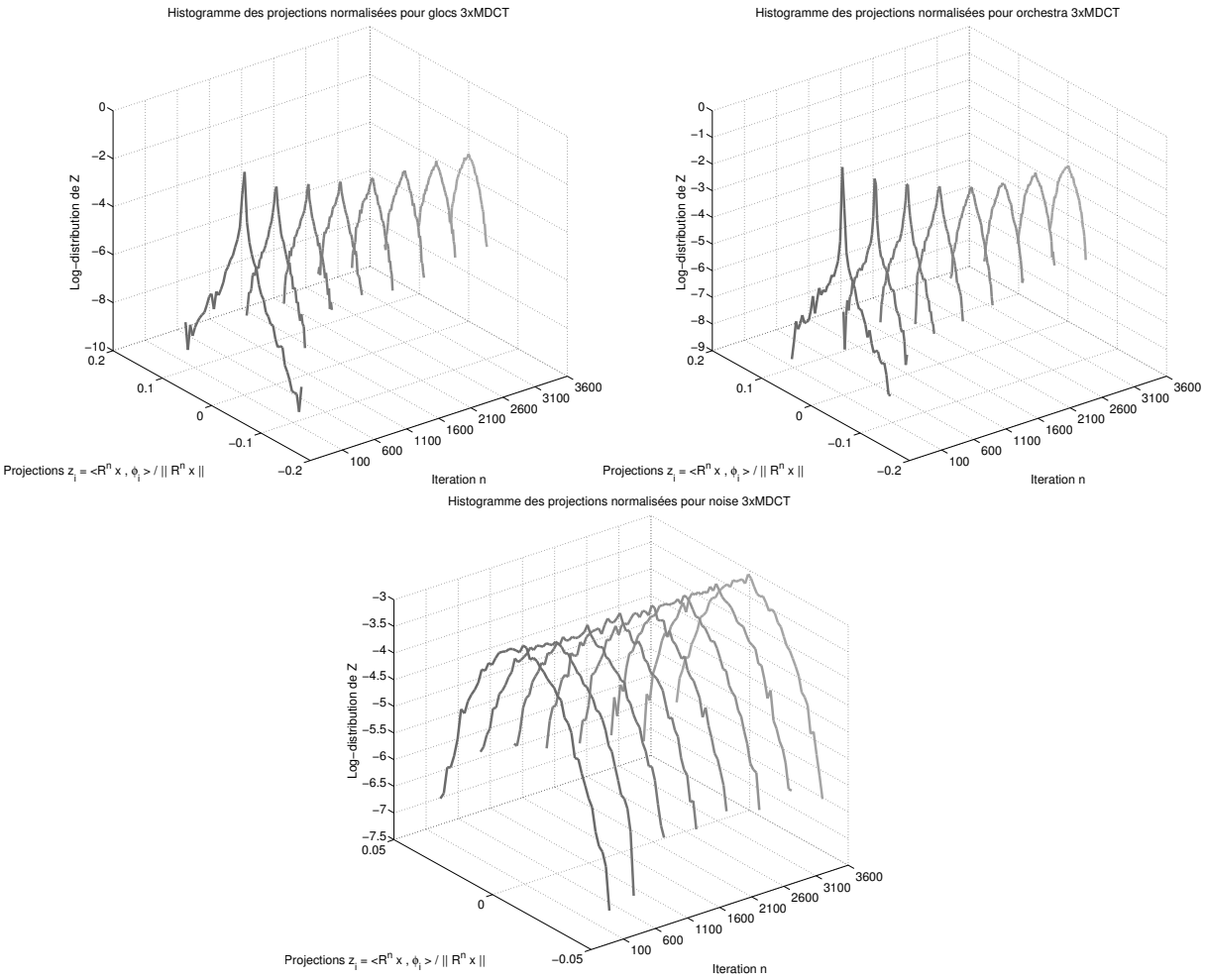

FIGURE 5.1.1: Évolution de la distribution des projections dans un dictionnaire 3xMDCT pour différents signaux : Glockenspiel (en haut à gauche) Orchestra (en haut à droite) et du bruit blanc gaussien (en bas)

de taille M de la variable Z :

$$\forall i \in [1..M], z_i = \frac{\langle R^n x, \phi_i \rangle}{\|R^n x\|_2} \tag{5.1.1}$$

Au fil des itérations de MP, le profil des projections du résiduel sur le dictionnaire change, ce qui va se traduire par une variation de la distribution de la variable aléatoire Z. Pour illustrer ce phénomène, on montre Figure 5.1.1 ces profils à différentes itérations pour trois exemples : deux scènes sonores réelles (glockenspiel et orchestre) et du bruit blanc gaussien. Pour permettre l'analogie avec des densités empiriques de probabilités, nous présentons les histogrammes normalisés de ces projections. Sur cette figure, on peut faire les remarques suivantes :

—∘ Pour les scènes sonores musicales, le profil original des projections ressemble visuellement à celui d'une distribution Bernoulli-Laplacienne. Cette distribution correspond à des signaux dont l'énergie est très localisée dans le plan temps-fréquence ; tandis qu'un petit nombre d'atomes capte une grande partie de l'énergie du signal (ce qui s'illustre par la queue longue de la distribution), le pic autour de zéro témoigne du fait qu'une proportion importante d'atomes sont très

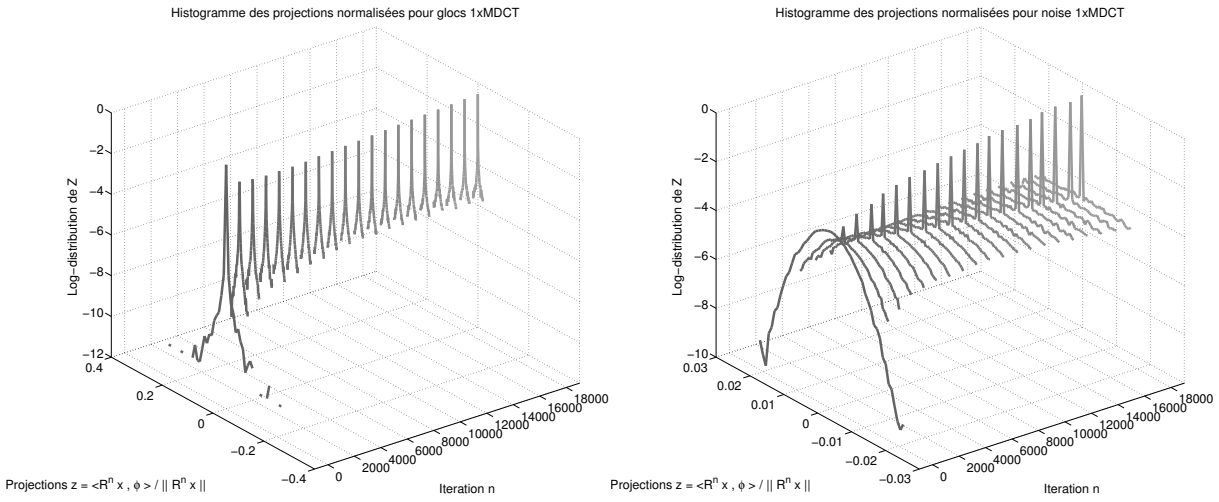

FIGURE 5.1.2: Évolution de la densité de probabilité empirique des projections au cours d'un MP dans une base orthonormale (MDCT fenêtre de 64 ms) pour une scène sonore de glockenspiel et un bruit blanc gaussien (1 seconde de signal soit $N = 32000$ échantillons)

peu ou pas du tout corrélés avec le signal. Ces atomes sont ceux qui pavent les zones du plan temps-fréquence dans lesquelles le signal ne présente presque pas d'énergie.

- Au cours de la décomposition, ce profil change. Rapidement, la queue de la distribution empirique se raccourcit et l'allure se rapproche d'une distribution Bernoulli-Gaussienne. A cet instant les caractéristiques les plus saillantes ont déjà été extraites, le résiduel se compose en partie de zones bruitées et de zones à très faible énergie. Enfin dans un dernier temps, cette différence entre zones s'estompe et un profil gaussien apparaît.
- De fait, pour le bruit blanc gaussien, les histogrammes normalisés sont cohérents avec un modèle gaussien centré, de variance proportionnelle à l'énergie du signal et ce dès le début de la décomposition (en dehors des toutes premières itérations), il n'y a pas ensuite d'évolution.

Profil dans des bases orthonormales Comme mentionné par LOVISOLO *et al* [LdSD10], le cas d'une décomposition dans une base orthonormale est particulier car la probabilité d'annuler le résiduel augmente avec le nombre n d'itérations jusqu'à valoir 1 pour $n = N$ itérations. Cette observation est également valable pour une poursuite orthogonale. La Figure 5.1.2 montre l'évolution des distributions empiriques lors de la décomposition de signaux (musique populaire et bruit blanc) dans une base orthonormale. Cette évolution est d'une nature différente : les projections tendent vers une Bernoulli-Gaussienne et l'amplitude du pic en zéro croît très rapidement.

Profils en fonction de la taille du dictionnaire Pour des unions de bases, augmenter le nombre de bases accélère le processus de transformation des profils de distribution. La Figure 5.1.3 montre pour les mêmes signaux que précédemment une évolution accélérée vers la distribution Gaussienne dans le cas d'une union de 8 bases par rapport à une union de 3 bases MDCT. Ceci n'est pas surprenant car on sait qu'augmenter la taille du dictionnaire accélère généralement le processus de convergence.

Si l'évolution du profil de la distribution des projections est acquise, quantifier cette évolution est un problème complexe. Dans une optique de poursuite gloutonne, on peut arguer que, plus que la distribution des projections elles-mêmes, c'est la localisation des maxima au fil des itérations qui nous

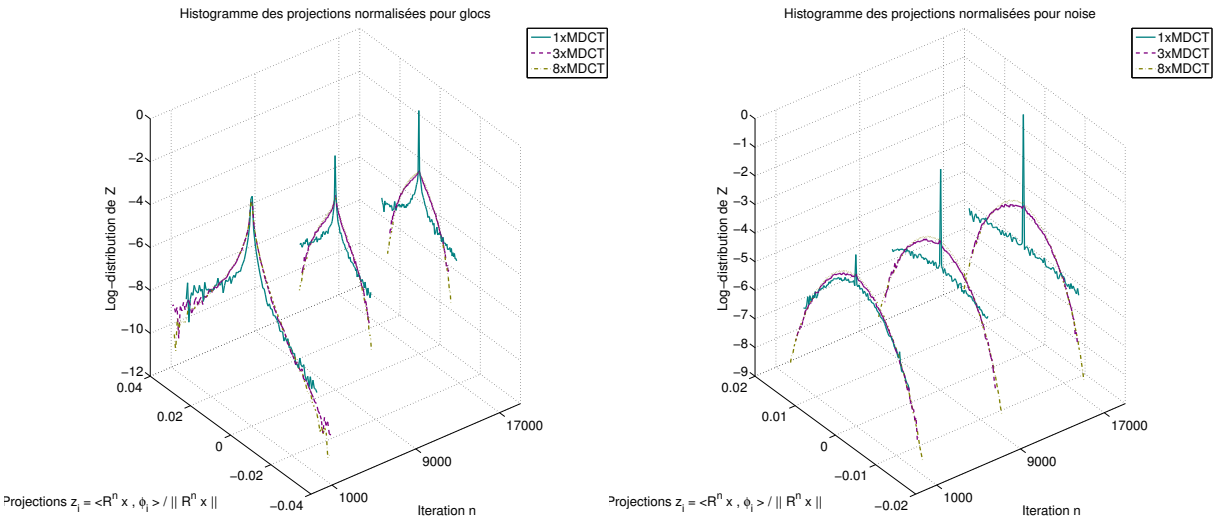

FIGURE 5.1.3: Évolution de la densité de probabilité empirique des projections d'un signal de glockenspiel (gauche) et d'un bruit blanc gaussien (droite) au cours d'un MP dans des unions de bases MDCT.

intéresse. En particulier, des indicateurs de l'étalement – la longueur de la queue – de cette distribution (tel que l'indice de Gini [HR08], la kurtosis ou encore la L-Kurtosis [Hos90, Haz12]) semblent pertinents et suffisants à première vue. Nous allons voir néanmoins qu'envisager cette distribution dans son ensemble peut nous permettre de proposer des métriques efficaces.

5.1.2 Métriques

Que les propriétés statistiques du résiduel changent au cours d'une exécution d'un algorithme MP est un phénomène connu, déjà étudié dans l'article de MALLAT et ZHANG [MZ93] à travers la variable :

$$\lambda_\Phi(R^n x) = \sup_{\phi\in\Phi} \frac{|\langle R^n x, \phi\rangle|}{\|R^n x\|} \tag{5.1.2}$$

dénommée cohérence normalisée (en relation à la cohérence (3.1.6)). Une autre manière de considérer cette même grandeur est sous la forme d'un angle :

$$\theta_\Phi(R^n x) = \arccos\left(\lambda_\Phi(R^n x)\right) \tag{5.1.3}$$

θ_Φ est alors le plus petit angle existant entre le résiduel normalisé et un vecteur du repère défini par Φ. En supposant que l'algorithme sélectionne effectivement le meilleur atome au sens de ces métriques, on peut lier la décroissance de l'énergie du résiduel à ces deux grandeurs :

$$\|R^{n+1}x\|^2 = \|R^n x\|^2 - \sup_{\phi\in\Phi} |\langle R^n x, \phi\rangle|^2 \tag{5.1.4}$$

$$\frac{\|R^{n+1}x\|^2}{\|R^n x\|^2} = 1 - \lambda_\Phi^2(R^n x) \tag{5.1.5}$$

$$= \sin^2\left(\theta_\Phi(R^n x)\right) \tag{5.1.6}$$

Dès lors, deux grandeurs caractéristiques du dictionnaire vont nous intéresser. La première est celle qui permet de borner la convergence de MP. C'est, au choix, la plus petite corrélation entre un signal

x non nul et un élément du dictionnaire, ou l'angle le plus grand existant entre un tel x et les vecteurs de Φ :

$$\Lambda(\Phi) = \inf_{x \in \mathcal{H}} (\lambda_\Phi(x)) \tag{5.1.7}$$

$$\Theta(\Phi) = \arccos(\Lambda(\Phi)) \tag{5.1.8}$$

On retrouve cette métrique sous le nom de redondance structurelle [FV01, FVFK04] et plus récemment la formulation angulaire dans les travaux de Lovisolo *et al* [LdSD10]. De façon évidente :

$$\frac{\|R^{n+1}x\|^2}{\|R^n x\|^2} \leq 1 - \Lambda^2(\Phi) = \sin^2(\Theta(\Phi)) \tag{5.1.9}$$

La relation (5.1.5) permet, par récurrence d'obtenir une borne supérieure sur l'énergie du signal résiduel à l'étape n :

$$\|R^n x\|^2 = \|x\|^2 \prod_{i=0}^{n-1} \left(1 - \lambda_\Phi^2(R^i x)\right) \tag{5.1.10}$$

$$\|R^n x\|^2 \leq \|x\|^2 \left(1 - \Lambda^2(\Phi)\right)^n \tag{5.1.11}$$

Pour autant cette borne est très lâche et ne permet pas en pratique de modéliser la convergence. La Figure 5.1.4 montre l'évolution de la cohérence normalisée au fil des itérations d'un MP standard sur un dictionnaire Φ_K redondant (de taille $K = 3N$) pour des scènes sonores et du bruit blanc. On voit que les valeurs tendent vers une asymptote. Le processus MP se comporte après les premières itérations comme un processus chaotique et le résiduel tend vers un attracteur nommé *bruit de dictionnaire* [MZ93]. La valeur asymptotique de la cohérence normalisée est celle obtenue pour un bruit blanc gaussien W. On observe qu'elle est caractéristique du dictionnaire et on note :

$$\Lambda_W(\Phi) = \mathbb{E}\left[\lambda_\Phi(W)\right] \tag{5.1.12}$$

Cette grandeur permet de borner la convergence de l'erreur de reconstruction, ou plus précisément le taux de convergence. Pour les scènes sonores réelles, ce taux est à l'origine beaucoup plus important que pour du bruit et dépendant du signal. Plus loin dans la décomposition, les résiduels sont *blanchis* et toutes les courbes de décroissance présentent le même profil, celui du bruit blanc, dominé par la valeur $\Lambda_W(\Phi)$.

Dans une optique de débruitage, cette valeur permet de déterminer un critère d'arrêt pour le MP. Seuls les atomes dont la cohérence normalisée est significativement plus importante que $\Lambda_W(\Phi)$ peuvent être conservés [MZ93, DB95]. Dans une optique de codage, Ravelli [RRD08] utilise une estimation empirique de cette valeur pour proposer un basculement de dictionnaire vers une base orthonormale.

Estimation de $\Lambda_W(\Phi)$ Nous présentons en annexe un calcul basé sur les statistiques d'ordre. Le principe consiste à considérer les projections $|\langle R^n x, \phi \rangle|$ normalisées par l'énergie du résidu comme les réalisations d'une variable aléatoire Z et d'estimer ensuite la cohérence structurelle à l'aide de statistiques d'ordre :

$$\Lambda_W(\Phi_M) = \mathbb{E}\left[\lambda_\Phi(W)\right] = \mu_{M:M} \tag{5.1.13}$$

ou M est le nombre de tirages c'est à dire le nombre d'atomes du dictionnaire Φ_M. $\mu_{M:M}$ est l'espérance de $Z_{M:M}$ la variable aléatoire décrivant le maximum parmi M tirages de Z. Soit en utilisant le moment

d'ordre 2 :

$$\mathbb{E}\left[\frac{\|R^{n+1}x\|^2}{\|R^n x\|^2}\right] = 1 - \mu^{(2)}_{M:M} \tag{5.1.14}$$

Ce qui signifie qu'étant donné la distribution de Z on peut modéliser la décroissance du résiduel par MP étape par étape. Les profils de distribution du maximum d'une variable aléatoire demi-normale et exponentielle sont donnés Figure 5.1.5. Pour une variable Z distribuée exponentiellement, une expression analytique de l'espérance de $Z_{M:M}$ est disponible (voir Annexe A). Ce modèle correspond aux premières itérations d'un MP. La valeur asymptotique $\Lambda_W(\Phi_M)$ suggère plutôt l'adoption d'un modèle gaussien pour les projections de W.

Partons de l'hypothèse que les projections normalisées de W dans Φ_M suivent une loi gaussienne. On pose Y la variable aléatoire modélisant ces projections. Soit M la taille du dictionnaire, l'ensemble $\{y_i\}_{i=1..M}$ des projections peut se voir comme un tirage de M échantillons de Y, tels que :

$$y_i = \frac{\langle W, \phi_i \rangle}{\|W\|} \tag{5.1.15}$$

avec :

$$Y \sim \mathcal{N}(0, \sigma_Y^2)$$

En posant un modèle normal sur les projections $\langle W, \phi_i \rangle \sim \mathcal{N}(0,1)$, il est raisonnable de poser comme variance : $\sigma_Y^2 = \frac{1}{N}$ où N est la dimension de W. Sous ces hypothèses, Z qui modélise la valeur absolue de Y suit une loi demi-normale de paramètres :

$$f^Z(z) = \begin{cases} \frac{\sqrt{2}}{\sigma_Z\sqrt{\pi}} & e^{-\frac{z^2}{2\sigma_Z^2}} \text{ si } z \geq 0 \\ 0 & \text{sinon} \end{cases}$$

avec :

$$\sigma_Z^2 = \sigma_Y^2\left(1 - \frac{2}{\pi}\right) \tag{5.1.16}$$

Pour une variable demi-normale, le calcul de l'espérance $\mu_{M:M}$ du maximum parmi M échantillons est complexe. En revanche, ces distributions sont uni-modales et on peut voir (Figure 5.1.5) que la médiane $\nu_{M:M}$ est proche de l'espérance. Or l'estimation de la médiane est beaucoup plus simple (le calcul est présenté en Annexe A) :

$$\nu_{M:M} = \sigma_Z\sqrt{2}\mathrm{erf}^{-1}\left(0.5^{\frac{1}{M}}\right) \approx \mu_{M:M} \tag{5.1.17}$$

En utilisant l'estimation donnée par (5.1.17) on trouve une estimation $\tilde{\Lambda}_W(\Phi_M) = \nu_{M:M}$ de $\Lambda_W(\Phi_M)$. Nous présentons Figure 5.1.4 en tirets noirs la borne donnée par cette estimée et la convergence de l'erreur de reconstruction associée :

$$\epsilon_{W,\Phi}(n) \quad = \quad 10\log_{10}\left[\left(1 - \Lambda_W^2(\Phi)\right)^n\right] \tag{5.1.18}$$

On a ainsi trouvé une nouvelle borne de convergence basée sur une modélisation du comportement du bruit de dictionnaire :

$$\mathbb{E}(\|R^n x\|^2) \leq \|x\|^2(1 - \nu_{M:M}^2)^n \tag{5.1.19}$$

Cette convergence correspond en quelque sorte au pire cas possible. Pour des dictionnaires de type Gabor ou union de bases MDCT, le bruit blanc gaussien est le signal dont la décomposition est la plus fastidieuse et sa vitesse de convergence est la plus lente que l'on puisse observer. On peut noter (bien que cela n'a pas été effectué durant cette thèse), que cette borne peut sans doute avantageusement remplacer celle proposée par Frossard *et al* [FVFK04] pour la quantification progressive des coefficients.

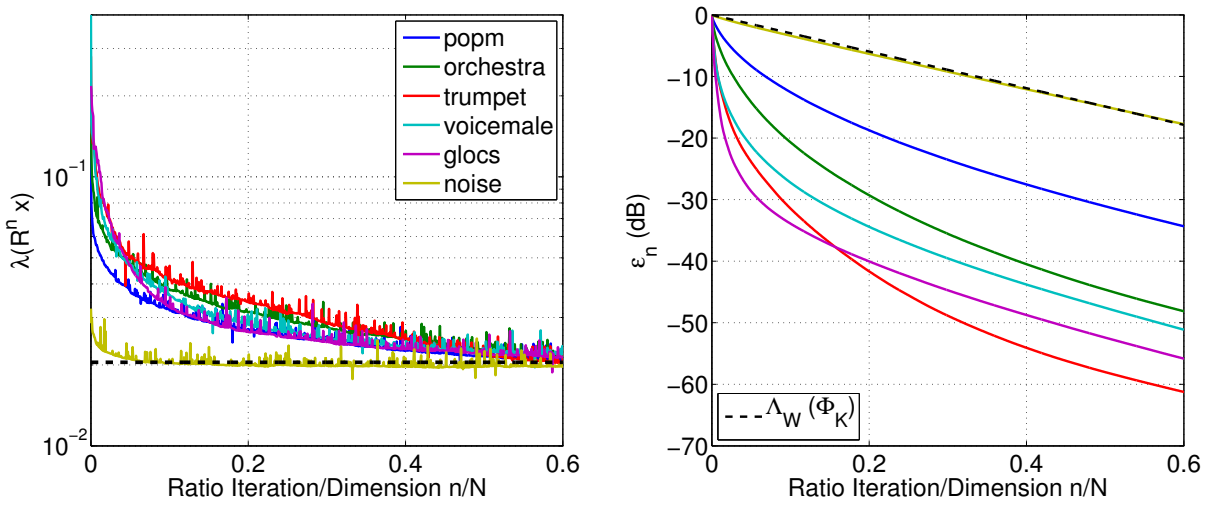

FIGURE 5.1.4: Cohérence normalisée et erreur de reconstruction pour différents signaux (1 seconde) au fil des itérations de MP (dictionnaire 3xMDCT). En tirets, estimation de $\Lambda_W(\Phi_K)$ et de la convergence obtenu par (5.1.17).

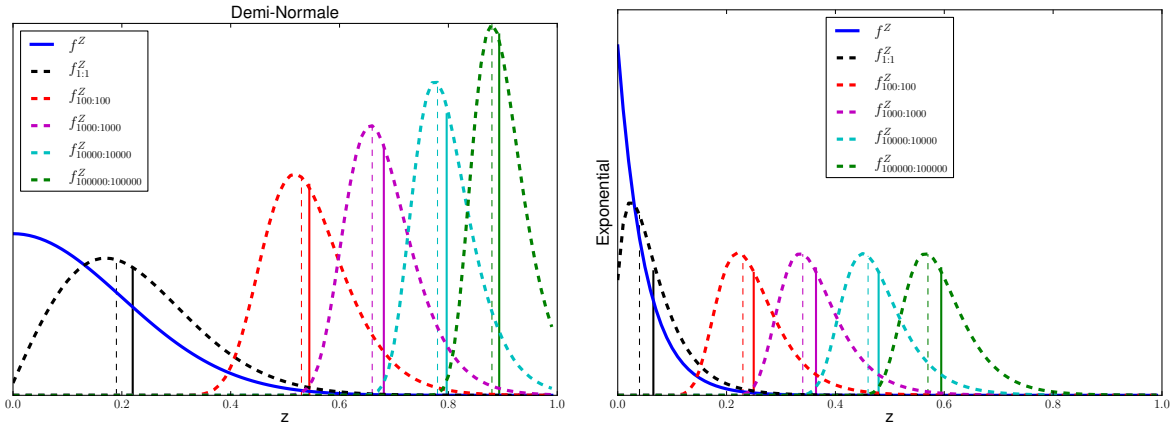

FIGURE 5.1.5: Densités de probabilité des maxima pour différents nombres d'échantillons. Gauche : modèle demi-normal de variance $\sigma^2 = 0.2$. Droite : loi exponentielle de paramètre $\lambda = 0.05$. Trait plein : espérance du maximum. Tirets : médiane.

Cas du SASMP Dans le cas du SASMP sur des sous-dictionnaires de taille fixe K, on peut comparer le comportement chaotique de la poursuite avec celui du MP sur un sous-dictionnaire fixe de taille K. La Figure 5.1.6 présente les profils de décroissance de $\lambda(R^n x)$ et de $\|R^n x\|$ pour différents signaux avec un SASMP sur une union de 3 bases MDCT. La borne théorique $\tilde{\Lambda}_W(\Phi_K) = \nu_{K:K}$ pour un sous-dictionnaire de taille $K = 3N$ est figurée en tirets noirs. En pratique SASMP tend en régime asymptotique vers la borne rouge $\tilde{\Lambda}_W(\Phi_M) = \nu_{M:M}$, donnée par (5.1.18) pour le grand dictionnaire de taille $M = \sum_s L_s N$. En d'autres termes, en régime asymptotique (*c-à-d* dès que le résiduel tend vers un bruit blanc), SASMP se comporte comme une poursuite dans le dictionnaire complet.

Il est également très intéressant de constater que la valeur de la borne ne dépend pas de la taille du sous-dictionnaire utilisé. En revanche, on peut estimer que le régime asymptotique sera atteint d'autant plus tardivement que cette valeur est faible.

Cette propriété est très intéressante comme nous allons le voir dans une optique de codage haut-débit. En effet, ce comportement semble indiquer que l'on peut fortement réduire la taille des sous-dictionnaires tout en garantissant la sélection d'atomes dont la corrélation avec le signal est approximativement aussi bonne qu'en utilisant le dictionnaire complet.

5.1.3 Vers un modèle complet

Pour modéliser complètement l'évolution des projections du résiduel – en dehors du cas de la décomposition d'un bruit blanc – la connaissance de l'asymptote Λ_W n'est pas suffisante. On voit en effet que la convergence du résiduel de scènes sonores réelles est bien plus rapide dans les premières itérations. A ce stade on peut donc se poser la question de savoir s'il est possible de modéliser plus finement cette convergence. Pour cela, il faut d'abord proposer un modèle pour le résiduel lui-même, évoluant au fil des itérations. Soit $R^n x \in \mathbb{R}^N$ le n-ième résiduel de la décomposition de x dans $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$, il est courant de proposer un modèle mixte support-amplitudes, c'est à dire faire l'hypothèse que $R^n x$ a une expression parcimonieuse dans Φ :

$$R^n x = \sum_{i=1}^{M} \left(w_i^n . s_i^n . \phi_{\gamma^i} + e_i^n \right) \tag{5.1.20}$$

où $\boldsymbol{e^n}$ est un bruit blanc centré de variance $\sigma^2_{e^n}$, $\boldsymbol{s^n}$ est un vecteur modélisant le support, à valeur dans $\{0,1\}^M$ et $\boldsymbol{w^n}$ est un vecteur de poids. Dès lors le vecteur $\boldsymbol{y^n}$ des projections dans le dictionnaire s'écrit :

$$\begin{aligned} y_i^n &= \langle R^n x, \phi_i \rangle \\ &= \sum_{j=1}^{M} w_j^n . s_j^n . \langle \phi_j, \phi_i \rangle + \langle e_i^n, \phi_i \rangle \\ &= w_i^n s_i^n + \sum_{j \neq i} w_j^n . s_j^n . \langle \phi_j, \phi_i \rangle + \sum_{j=1}^{M} \langle e_j^n, \phi_j \rangle \end{aligned}$$

on peut voir que pour des dictionnaires quasi-incohérents (ce qui est le cas par exemple d'une union de bases MDCT, mais pas si l'on considère un dictionnaire de Gabor discret de pavage fin comme en 4.3) les produits $\langle \phi_j, \phi_i \rangle$ sont nuls ou négligeables si $i \neq j$. On peut alors récrire y^n en négligeant le terme d'interférences :

$$y_i \simeq w_i^n s_i^n + v_i^n$$

où $\boldsymbol{v^n}$ est à son tour gaussien centré. Autrement dit les projections suivent elles aussi une loi mixte support-amplitude avec bruit. En supposant l'indépendance des composants de $\boldsymbol{s^n}$ on peut supposer que les s_i^n suivent une loi de Bernoulli de paramètre p^n.

Pour des scènes sonores réelles la distribution originale des projections dans les dictionnaires temps-fréquence de type union de bases MDCT présente un profil Bernoulli-Laplacien :

$$\mathcal{BL}(y^0;\beta^0) \rightarrow \begin{cases} p(s_i^0=1)=p^0 \\ p(w_i^0|s_i^0=1) \propto \frac{1}{\beta^0} e^{-\beta^0 w_i^0} \\ p(w_i^0|s_i^0=0)=\delta_0 \end{cases} \tag{5.1.21}$$

où β^0 est un paramètre et δ_0 un dirac.

Plus loin dans la décomposition, le profil se rapproche d'un Bernoulli-Gaussien, soit :

$$\mathcal{BG}(y^n;\sigma_y) \rightarrow \begin{cases} p(s_i=1)=p^n \\ p(w_i|s_i=1) \propto \frac{1}{\sigma_y^2} e^{-\frac{w_i^2}{2\sigma_y^2}} \\ p(w_i|s_i=0)=\delta_0 \end{cases}$$

Enfin dans les dictionnaires redondants, y tend en distribution vers y^∞, gaussien centré. Un modèle complet doit proposer une telle mutation de y^0 vers y^∞. Nous avons essayé plusieurs distributions classiques (Cauchy, Student-t, etc..), toutes ne sont susceptibles d'expliquer les profils de y^n que dans une plage de valeur réduite. Prédire l'évolution de y^n (c'est à dire prédire l'évolution de p^n, w^n et v^n) demeure un problème ouvert dans le cas du MP sur un dictionnaire fixe, et *a fortiori* pour le SSMP.

5.2 Sous-échantillonnage dynamique

En poussant la logique du SSMP plus avant, on peut envisager une poursuite sur des sous-dictionnaires de taille variable. Les observations qui précèdent sur l'évolution de la distribution des projections peuvent alors nous aiguiller sur le type de stratégie à adopter.

5.2.1 Principe

Plages d'optimalité des sous-échantillonnages En fixant une taille de sous-dictionnaire fixe K, on borne les performances de SASMP. En particulier dans les premières itérations, SASMP va sélectionner des atomes d'autant moins bons que l'étendue de son choix est réduite. Pourtant, on a vu que SASMP tend vers une poursuite dans le grand dictionnaire, avec une valeur de la borne indépendante de la taille K. Dans un tel mode asymptotique, les atomes sélectionnés sont équivalents (en terme de cohérence normalisée) quel que soit K. En revanche, plus K est faible, moins cet atome est coûteux à coder.

La conjonction de ces deux phénomènes a pour conséquence l'existence d'une plage d'optimalité du compromis débit-distorsion pour chaque sous-échantillonnage. Les valeurs de cette plage sont très dépendantes du signal. Nous présentons Figures 5.2.1 et 5.2.2 les courbes obtenues pour différentes scènes sonores et du bruit blanc avec SASMP sur des sous-dictionnaires de différentes tailles, ainsi que le SNR relatif au cas de figure $\Delta_u^s = L_s/2$ (sous-dictionnaires de taille fixe $K^0 = 3N$) ce cas de figure correspond au plus petit dictionnaire complet. Ce qui signifie que pour $K < K^0$ les sous-dictionnaires utilisés ne sont plus complets ! On voit qu'en pratique cela ne pose pas de problème car SASMP se comporte asymptotiquement comme une poursuite sur un dictionnaire très redondant.

Nous pouvons voir que sur le signal de glockenspiel, le sous-échantillonnage est très dommageable à bas débit, un sous-échantillonnage de $K = 0.5K^0$ ne devient intéressant qu'au delà de 32 Kbps de débit. Pour le signal de musique populaire, cette limite n'est que de 16 Kbps (et même 12 Kbps pour un signal de voix, non montré sur ces figures).

Dans le cas du bruit blanc, on observe un compromis intéressant, conformément à ce qu'on pouvait attendre, on peut très largement sous-échantillonner même à bas-débit. Une valeur optimale semble se trouver autour de $K = K^0/6$. Il faut néanmoins garder à l'esprit que même avec SASMP et la meilleure configuration, ces performances sont très inférieures à celles observées pour des scènes sonores réelles.

Dans cette expérience nous avons arrêté les décompositions lorsqu'un débit théorique de 256 Kbps était atteint (sans codage entropique des indices des atomes). L'allure des courbes indique que plus le débit augmente plus il est intéressant de sous-échantillonner car on gagne ainsi sur les deux tableaux :

- Plus K est faible plus le coût de codage des indices est faible et l'on peut à débit constant augmenter le nombre d'atomes, ce qui réduit la distorsion (augmente le SNR)
- Plus K est faible plus SASMP est rapide, car le nombre de projections calculées est fortement réduit.

En revanche, à bas-débit, la distorsion additionnelle introduite par le sous-échantillonnage n'est pas compensée par le gain en débit. Cette observation est à rapprocher de ce que l'on sait de la nature des distributions des projections dans les premières itérations. Rappelons en effet que dans ce régime, ces distributions ont une queue plus longue et un sous-échantillonnage aura donc pour conséquence la sélection d'un atome significativement sous-optimal.

Compromis débit-distorsion Avec l'évolution du profil de la distribution des projections dans le dictionnaire, c'est le profil du compromis débit-distorsion qui est modifié. Pour s'en convaincre, adoptons le modèle simpliste suivant :

- Le débit est dominé par le coût de codage des indices et le gain obtenu par sous-échantillonnage K parmi K^0 est :

$$G(K) = \log_2 \left(\frac{K^0}{K} \right) \tag{5.2.1}$$

- La distorsion relative induite par sous-échantillonnage K parmi K^0 est, toujours selon un modèle simpliste :

$$r_\epsilon(z, K) = 10 \log_{10} \left(\frac{Z^2_{K^0:K^0}}{Z^2_{K:K}} \right) \tag{5.2.2}$$

La figure 5.2.3 montre des simulations pour $K^0 = 128$ et différentes distributions. Comme mentionné plus haut, si l'on disposait d'une estimation de l'évolution des distributions, on pourrait en prenant en compte les profils du compromis débit-distorsion associé proposer un sous-échantillonnage optimal.

Malheureusement, il est difficile d'estimer en avance cette évolution en l'absence de modèle complet (voir section 5.1.3). On peut en revanche mettre en place des stratégies dynamiques non-adaptatives de sous-échantillonnage progressif, car même si l'on ignore leur nature précise, l'expérience nous montre que ces distributions évoluent, pour des scènes sonores réelles, d'un profil type Bernoulli-Laplacien vers un profil gaussien.

Sachant que le compromis est plus avantageux à mesure que le résiduel se blanchit, nous pouvons envisager une poursuite dynamique profitant de cette connaissance.

5.2.2 Variation dynamique de la taille du sous-dictionnaire

Connaître les plages d'optimalité précises des facteurs de sous-échantillonnage (*c-à-d.* les plages de débit pour lesquels le sous-échantillonnage de facteur K présente le meilleur SNR possible) pour une scène sonore particulière est un problème difficile. Il n'est pas envisageable de tester tous les sous-échantillonnages non plus pour trouver empiriquement ces plages. De plus, même en supposant que l'on connaisse ces valeurs, un problème subsiste : faut-il adapter la taille du sous-dictionnaire à chaque itération ou par plages d'itérations ? Auquel cas il faut considérer dans une application de codage que ces plages seront des paramètres supplémentaires à transmettre.

Si l'on veut s'épargner la transmission de ces paramètres, il faut fixer à l'avance la taille des sous-dictionnaires. Pour illustrer l'intérêt d'un MP dynamique, nous avons implémenté un exemple non adaptatif, c'est à dire sans connaissance de ces plages d'optimalité. Au cours de la décomposition d'un signal, nous réduisons successivement la taille du sous-dictionnaire. Ce faisant nous proposons un codeur sans garantie d'optimalité, mais à l'implementation simple et qui présente un comportement cohérent sur l'ensemble des signaux considérés.

L'algorithme proposé est donc un SASMP avec une séquence de sous-dictionnaires $\{\Phi_n\}$ de taille K^n variable. On choisit pour cet exemple une règle simple basée sur l'observation moyenne des plages d'optimalité :

$$K^n = \begin{cases} K^0 & \text{si } \frac{n}{N} \leq 0.02 \\ \frac{K^0}{2} & \text{si } 0.02 < \frac{n}{N} \leq 0.05 \\ \frac{K^0}{4} & \text{si } 0.05 < \frac{n}{N} \leq 0.1 \\ \frac{K^0}{6} & \text{si } 0.1 < \frac{n}{N} \end{cases}$$

ces limites sont parfaitement arbitraires mais offrent l'avantage d'être non adaptatives – donc transparentes en termes de coûts de transmission – et très simples à mettre en place. En particulier on notera ici les faibles valeurs des bornes de ratio Itération/dimensions considérées, au delà de 10% de la dimension originale, nous considérons le résiduel comme une réalisation d'un bruit blanc et adoptons en conséquence le sous-échantillonnage $K^0/6$ qui apparaît expérimentalement comme proche de l'optimal.

5.2.3 Performance en codage

La figure 5.2.4 montre que la stratégie proposée permet en moyenne des performances toujours supérieures ou égales aux approches fixes, et ce sur toutes les plages de débit. Ceci semble valider la pertinence de cette approche, surtout si l'on considère le fait que le SASMP dynamique proposé est de plus en plus rapide (il y a de moins en moins de projections à calculer). Cette application semble montrer qu'il est possible, en adoptant une stratégie aléatoire dynamique de gagner sur à peu près tous les tableaux : vitesse, précision et concision.

5.3 Sous-échantillonnage des lignes et des colonnes

En conclusion de ce chapitre, nous pouvons faire le lien entre ces travaux et un algorithme récent proposé par Peel *et al* [PERA12] ainsi qu'avec les techniques de factorisation matricielle utilisant des sous-échantillonnages aléatoires.

5.3.1 Matching Pursuit à sélection Stochastique

Le MP stochastique proposée dans [PERA12] reprend l'idée de l'utilisation d'une séquence de sous-dictionnaire et l'étend à la sélection aléatoire de lignes en plus des colonnes. En d'autres termes, à chaque itération, seul un sous-ensemble des projections est considéré pour la recherche du maximum mais de plus, c'est une approximation des produits scalaires qui sert à déterminer ce maximum. En utilisant une formulation matricielle : on pose $y^n = \Phi^T R^n x$ avec $\Phi \in \mathbb{R}^{N\times M}$, le vecteur des M projections du résiduel à l'étape n. MP cherche l'indice γ^n de l'élément maximal de $|y^n|$ avant de soustraire sa contribution à $R^n x$ et d'itérer. Le principe est de remplacer Φ dans cette opération par un sous-dictionnaire $\Phi_{\kappa,\mu}$ où κ est le ratio de sous-échantillonnage des lignes et μ celui des colonnes. On tire donc un sous-ensemble $\mathcal{I}^n$ de colonnes et un sous-ensemble $\mathcal{J}^n$ de lignes, et en lieu et place de y^n on calcule donc :

$$y^n_{\kappa,\mu} = \Phi^T_{\kappa,\mu} R^n x \tag{5.3.1}$$

qui ne contient donc que μM éléments ($0 < \mu \leq 1$) qui s'écrivent chacun :

$$\forall i \in \mathcal{I}^n, y^n_{\kappa,\mu}(i) = \sum_{j\in\mathcal{J}^n} \phi_i[j] \cdot R^n x[j] \tag{5.3.2}$$

L'intérêt est là aussi l'accélération de l'algorithme, avec au final la possibilité de considérer des dictionnaires beaucoup plus grands. Il faut néanmoins préciser que le sous-échantillonnage des lignes et des colonnes relèvent de deux stratégies bien distinctes :

Le sous-échantillonnage des colonnes a un impact direct, presque irréversible, sur la sélection des atomes en limitant de fait le choix possible. En effet un atome ϕ_γ tel que $\gamma \notin \mathcal{I}^n$ ne pourra pas être sélectionné. En particulier on a vu que cette limitation est fortement dommageable dans les cas de figure ou un support exact est recherché.

Le sous-échantillonnage des lignes introduit quant à lui un terme d'erreur sur le calcul des projections car :

$$y^n_{\kappa,\mu}(i) = \sum_{j\in\mathcal{J}^n} \phi_i[j] \cdot R^n x[j] = \langle \phi_i, R^n x\rangle - \sum_{j\notin\mathcal{J}^n} \phi_i[j] \cdot R^n x[j] \tag{5.3.3}$$

dès lors, le sous-échantillonnage des lignes n'a pas pour effet d'interdire la sélection d'un atome. En ce sens, on est plus proche de la philosophie du *compressed sensing*. Plus généralement, le sous-échantillonnage des lignes et des colonnes peut se voir comme une forme *d'Approximate Weak Greedy Algorithm* [GN01].

Dans leur travail (mené indépendamment du nôtre) Peel *et al* [PERA12] font des observations proches, à savoir que la variation aléatoire du sous-dictionnaire s'avère dans la plupart des cas bénéfique. Les preuves théoriques de cette supériorité sont néanmoins difficiles à fournir, une collaboration sur ce sujet entre leur équipe et la notre est d'ailleurs en cours. En particulier, nous sommes intéressés à étendre les principes décrits plus haut de MP dynamiques à leur formalisme. L'idée sous-jacente est d'adapter la stratégie de sous-échantillonnage à l'évolution de la distribution des projections y^n. Intuitivement, on peut imaginer que dans les premières itérations, le sous-échantillonnage des lignes soit moins critique que celui des colonnes, et que ce rapport s'inverse au fil des itérations.

5.3.2 MP dynamique dans le contexte

Pour finir, notons que, si nous avons présenté le SASMP dans le cadre d'une décomposition de scènes sonores et pour une application précise de codage, le fait que cette technique soit utilisée

par d'autres chercheurs sur d'autres types de signaux et dans un formalisme plus général est en soi satisfaisant. Clairement, les techniques de sous-échantillonnage aléatoire sont actuellement un domaine de recherche en plein essor.

Sans même parler des techniques de type Monte-Carlo déjà très utilisées par la communauté statistique, il est intéressant de noter l'apparition récente de techniques de factorisation matricielle basées sur un sous-échantillonnage aléatoire, et notamment les travaux de HALKO *et al* [HMT09]. Ces techniques et leurs développements très récents [NT12] proposent des algorithmes de calcul rapide de décomposition en valeur singulières (SVD) et de factorisation CUR. Dans ces travaux, les auteurs s'attachent à trouver un sous-espace de dimension réduite qui capture l'essentiel de *l'action* d'une matrice large, c'est à dire étant donné une matrice $\boldsymbol{A}$ trouver une matrice $\boldsymbol{Q}$ dont les colonnes soient orthogonales et qui vérifie :

$$\boldsymbol{A} \approx \boldsymbol{Q}\boldsymbol{Q}^*\boldsymbol{A}$$

le but étant bien sûr de trouver $\boldsymbol{Q}$ ayant le moins de colonnes possible. Et la façon de trouver ce sous-espace est d'utiliser un échantillonnage aléatoire, c'est à dire un ensemble de projections aléatoires :

$$\boldsymbol{y}^i = \boldsymbol{A}\boldsymbol{\omega}^i, i = 1..k \tag{5.3.4}$$

Le sous-espace engendré par ces projections (en supposant $\boldsymbol{A}$ de rang k), une fois orthogonalisé, fourni une matrice $\boldsymbol{Q}$ qui capte avec une forte probabilité l'essentiel de l'action de la matrice $\boldsymbol{A}$.

Ces méthodes commencent à faire leur apparition dans le domaine des représentations audio (p.ex. pour la factorisation en matrices non-négatives de spectrogrammes [ASCA12]). Bien que fondamentalement différentes, ces approches mettent en évidence le même type de propriétés de l'échantillonnage aléatoire, à savoir :

- Une accélération des calculs
- Une convergence rapide vers une solution pertinente.

A l'avenir, il sera intéressant de comparer les algorithmes de poursuites type SASMP avec ces procédures d'échantillonnage aléatoire. Une interprétation intuitive de SASMP se fait par analogie avec les techniques d'étalement de spectre *(spread spectrum* [Dix94]) dans le domaine des télécommunications. Récemment ce type de méthodes a été adapté au *Compressed Sensing* par PUY *et al* [PVGW12]. Dans ces techniques, un générateur pseudo-aléatoire connu est utilisé pour transformer un signal bande étroite en un autre, qui sans la connaissance de la séquence pseudo-aléatoire ressemble à du bruit large bande.

Par analogie, SASMP décompose un signal en une somme d'éléments vivant dans un espace beaucoup plus grand, et seule la connaissance préalable de la séquence pseudo-aléatoire de parcours de ce dictionnaire permet de reconstruire le signal de départ.

Il est également possible de faire des analogies avec le phénomène de *dither* utilisé en quantification [ZF92] par grille aléatoire. En reprenant le formalisme de MP faible, le fait que l'algorithme choisisse un atome sous-optimal dans un sous-dictionnaire peut se comprendre comme l'addition d'un bruit de quantification lors de la reconstruction. C'est ce bruit que DURKA [DIB01] présente comme un biais lié au choix d'un sous-dictionnaire. Le sous-échantillonnage aléatoire permet de réduire ce bruit de quantification par une méthode analogue au *dithering,* à savoir introduire de l'aléatoire dans la définition des grilles de quantification (ici dans le choix du sous-dictionnaires).

Conclusion

Nous avons étudié les propriétés asymptotiques des décompositions par MP. En utilisant les statistiques d'ordre, nous avons pu proposer un modèle statistique pour la décomposition d'un signal gaussien dans un dictionnaire. Une estimation de la vitesse de convergence assez précise a pu être dérivée. Et nous avons pu vérifier que le comportement asymptotique d'une poursuite sur une séquence de sous-dictionnaires aléatoires était équivalent à celui d'une poursuite dans le dictionnaire non sous-échantillonné.

Dans une optique de codage, ce comportement justifie l'utilisation de dictionnaire de plus en plus petit à mesure que le débit augmente (*c-à-d.* au fil de la décomposition). Une modélisation complète de la convergence des algorithmes dynamiques reste encore à proposer. De plus, il est possible d'envisager de faire varier d'autres paramètres de l'algorithme de façon dynamique, notamment la règle de mise à jour et le critère de sélection des atomes. Cette plasticité nous conforte dans l'idée d'une utilisation dans un contexte d'archivage, pour traiter hiérarchiquement les différentes contraintes.

Si nous nous sommes concentrés pour l'instant sur des tâches de codage, la partie suivante s'intéresse aux tâches d'indexation que nécessite l'archivage de scènes sonores.

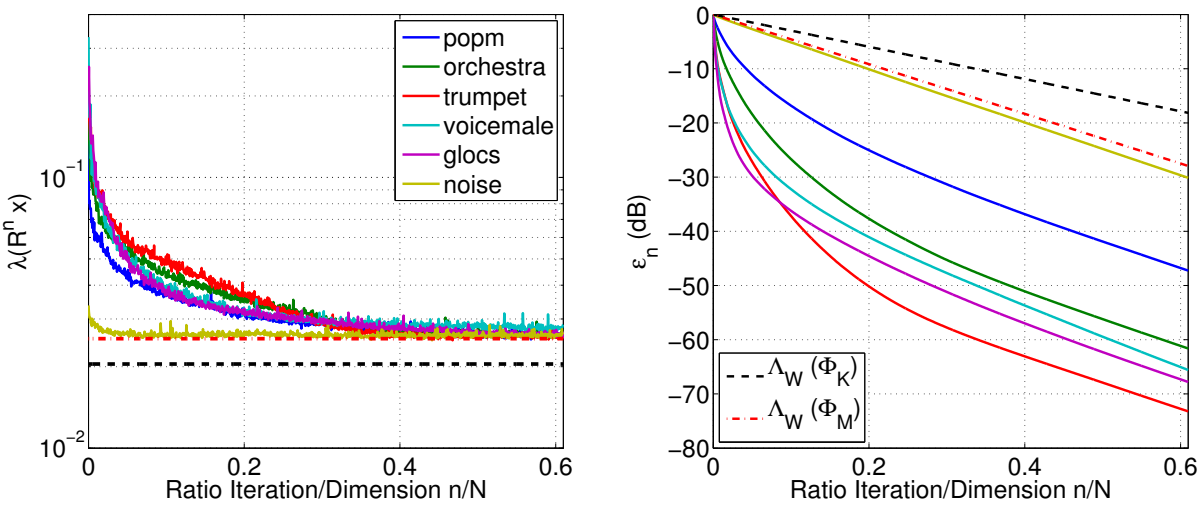

FIGURE 5.1.6: Cohérence normalisée et erreur de reconstruction pour différents signaux (1 seconde) au fil des itération de SASMP (dictionnaires 3xMDCT). En tirets, estimation de $\Lambda_W(\Phi_M)$ et $\Lambda_W(\Phi_K)$ et de la convergence obtenu par (5.1.17).

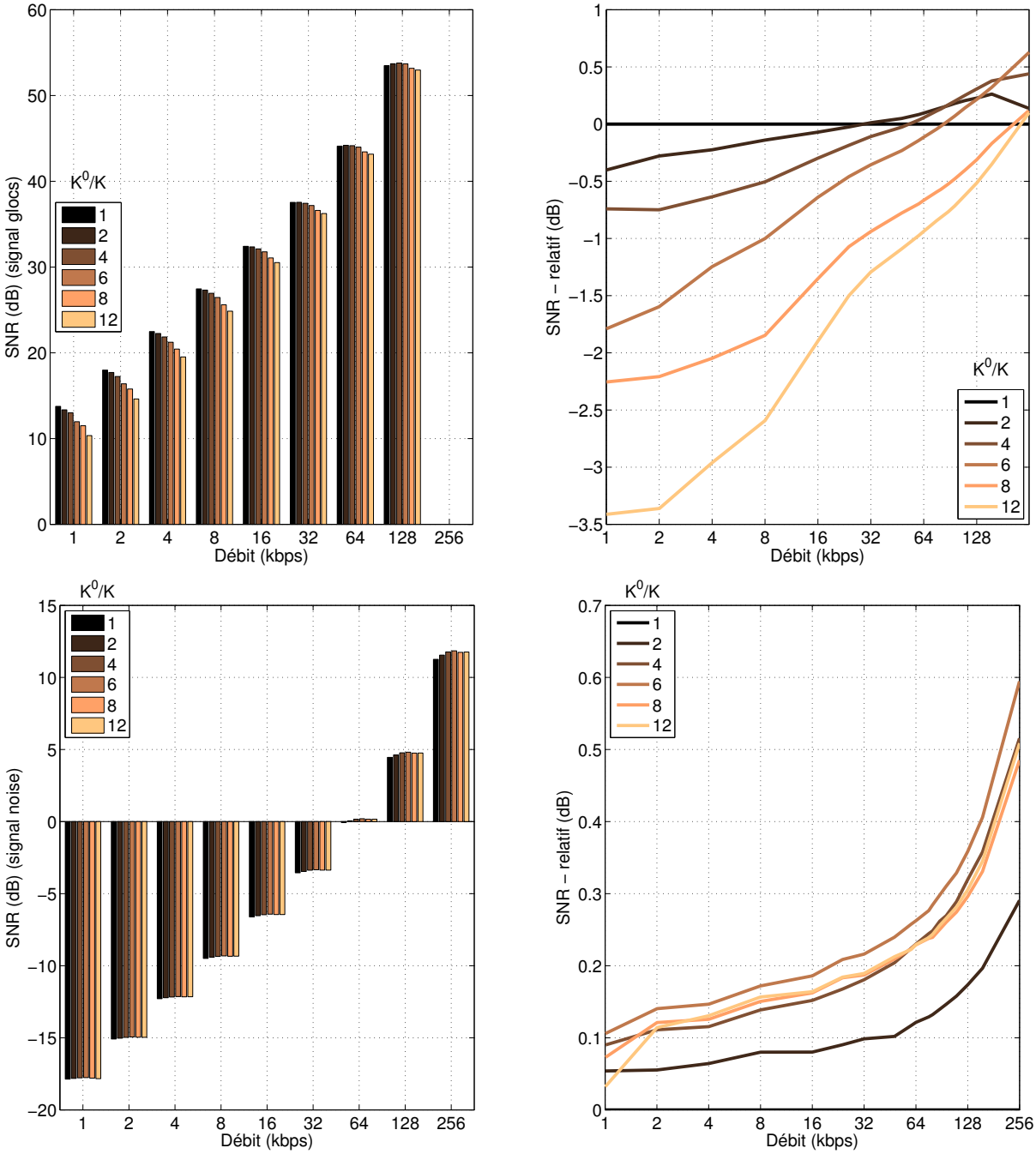

FIGURE 5.2.1: Gauche : Courbes débit-distorsion pour SASMP pour différents facteurs de sous-échantillonnage K^0/K. Droite : Courbes relative. K^0 correspond à une union de 3 bases MDCT. Haut : signal de glockenspiel. Bas : bruit blanc gaussien.

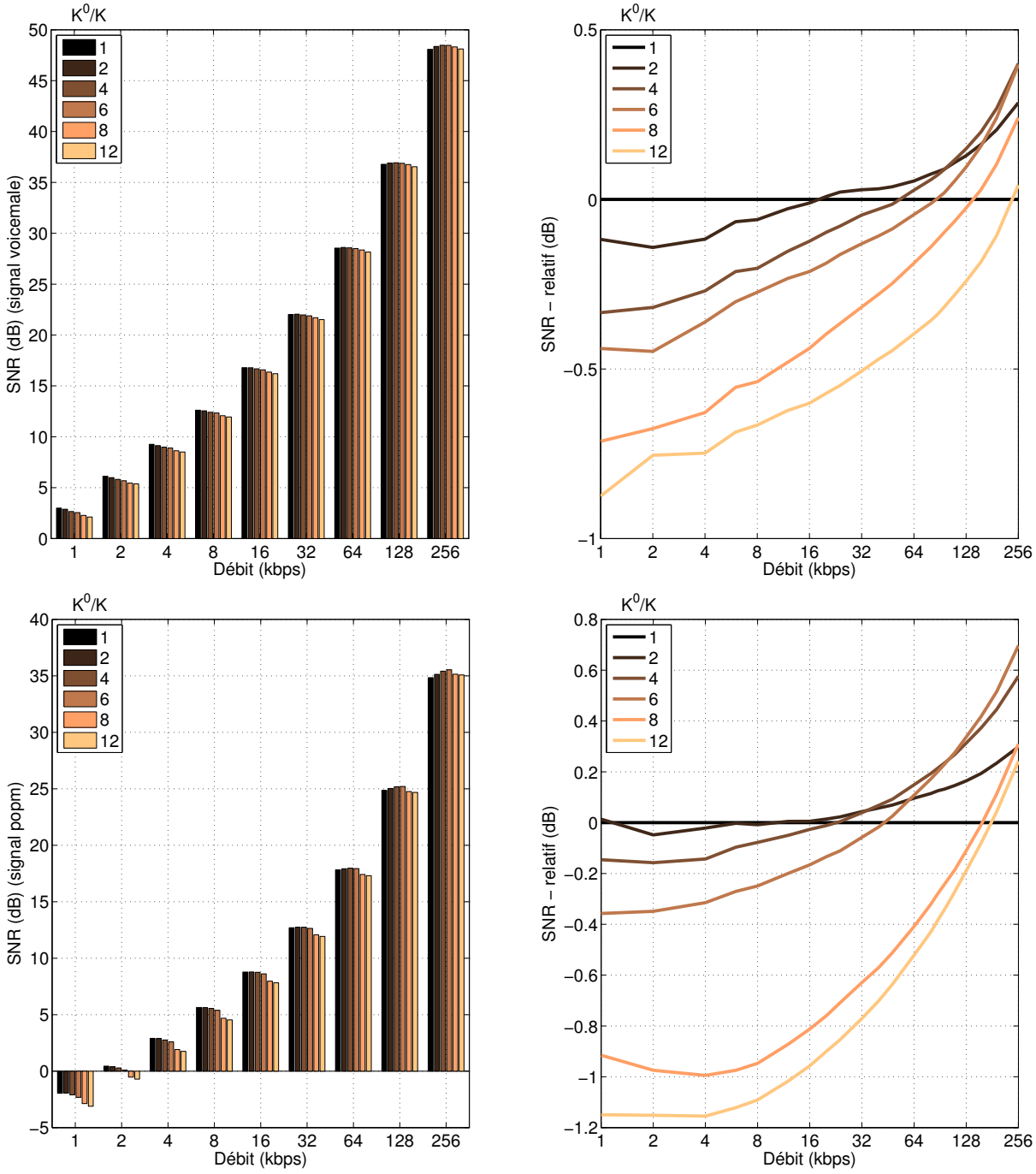

FIGURE 5.2.2: Gauche : Courbes débit-distorsion pour SASMP pour différents facteurs de sous-échantillonnage K^0/K. Droite : Courbes relative. K^0 correspond à une union de 3 bases MDCT. Haut : signal d'orchestre. Bas : Signal de musique pop.

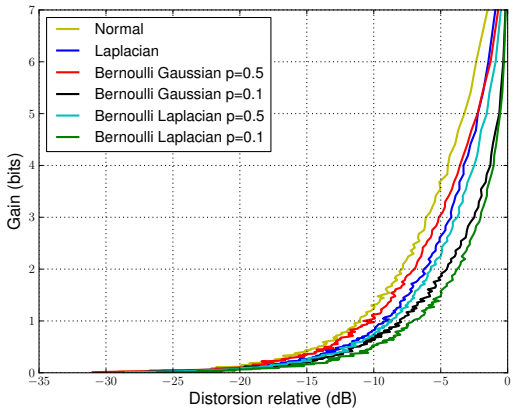

FIGURE 5.2.3: Profil simpliste des compromis débit-distorsion pour différentes distributions des projections z (valeurs moyennes pour 1000 simulations).

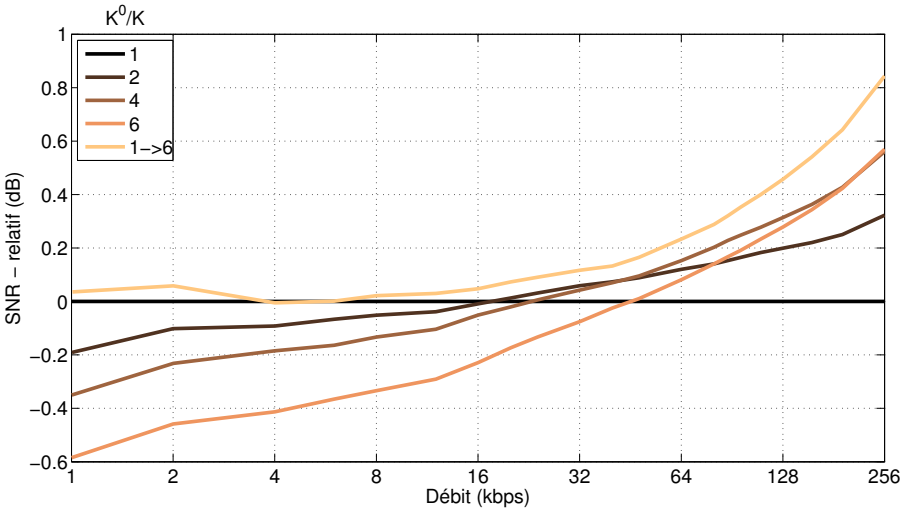

FIGURE 5.2.4: Comparaison d'une stratégie dynamique et de sous-échantillonnages fixes sur une tâche de codage. Le sous-échantillonnage dynamique offre les meilleures performances quelque soit le débit. Moyenne sur 10 exécutions de SASMP sur 4 scènes sonores réelles (glockenspiel, voix d'homme, orchestre et musique populaire).

Troisième partie

Redondances et Structures

Chapitre 6

Structures des scènes sonores et de leurs représentations

Nous avons jusqu'ici présenté des méthodes de représentations de signaux en général et de scènes sonores en particulier entièrement basées sur une hypothèse simple de parcimonie dans un repère donné. Dans cette partie, nous les considèrerons comme des outils pour résoudre des problématiques plus larges liées au problème de l'archivage. Dans ce cadre élargi, il n'est plus possible d'envisager les données indépendamment de leur contexte. En particulier lorsque le volume de ces données devient grand, il faut pouvoir appréhender leur organisation interne, leurs dépendances éventuelles les unes aux autres. Ce faisant, on cherche à mettre à jour les différents niveaux de *structures* qui les constituent.

Nous verrons que les structures doivent s'envisager à la fois comme une propriété *naturelle* des données considérées et comme propriété *souhaitable* de leurs représentations. La traduction des structures de signaux dans le domaine de la représentation n'allant pas de soi, nous verrons quelles contraintes s'appliquent à quels types de structures.

En premier lieu, nous décrirons dans la section §6.1 les niveaux de structures pertinents dans un contexte d'archivage de scènes sonores. Nous verrons que des échelles très variées sont à considérer ce qui induira des contraintes fortes. Une fois défini un niveau d'organisation intéressant, il faut encore trouver des moyens de le mettre à jour et des tâches dans lesquelles son exploitation peut s'avérer utile.

Ces premiers niveaux de structures feront apparaître la nécessité d'étendre les méthodes de représentations parcimonieuses aux données structurées. Nous verrons donc section §6.2 les modèles existants et les algorithmes disponibles. Dès lors, il nous sera possible d'étudier les conditions de représentations des différents types de structures.

Pour finir, nous introduirons les applications qui ont été proposées dans cette thèse et font l'objet des deux prochains chapitres, à savoir la détection de motifs récurrents par comparaison d'empreintes acoustiques, et la séparation de sources répétitives.

6.1 Structures, échelles et contraintes

6.1.1 Différents niveaux de structure

Redondances La redondance est la forme la plus répandue de structure de données. Sur une séquence de symboles – un message – $X = \{x_1, x_2, .., x_k\}$, on recherche les redondances sous la forme

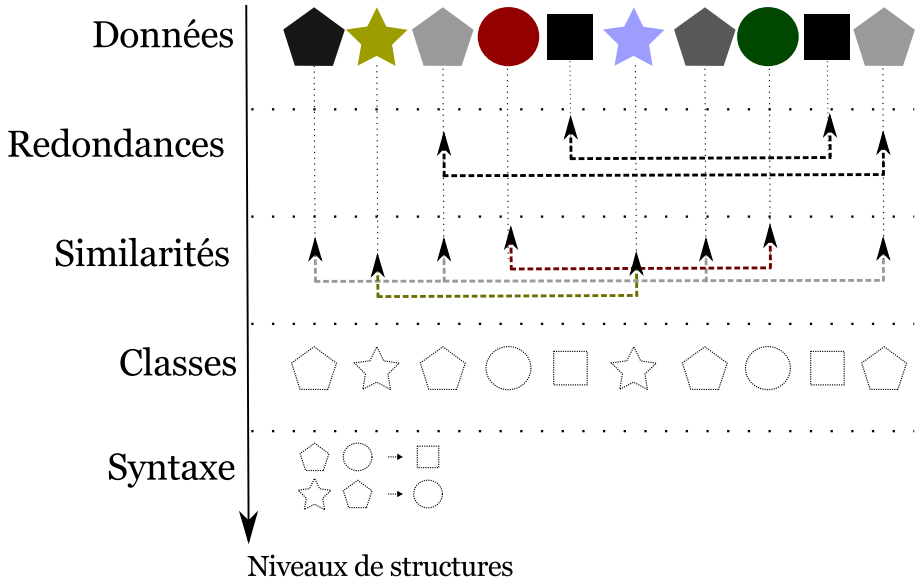

FIGURE 6.1.1: Schéma des niveaux de structuration d'un ensemble de données (segmenté en une séquence d'objets). Au premier niveau on trouve les redondances, puis les répétitions ou similarités, puis le regroupement en classes, enfin les structures génératives ou de syntaxe.

de sous-séquences de X récurrentes. Par exemple, le message binaire 0101011 est redondant puisqu'il contient plusieurs fois la séquence 01. De la même façon mais à une autre échelle, le flux radiophonique de la plupart des stations de radio françaises est redondant, car des scènes sonores d'une durée de plusieurs secondes, voire minutes (jingles, publicités, chansons, reportages, rediffusion d'émissions) sont répétées à différents moments du flux. A une échelle encore supérieure, il est facile de voir que la base de données des sites de diffusion de vidéos en ligne est également redondante, le même objet audiovisuel étant stocké en différents endroits.

Caractériser une redondance revient à une simple comparaison binaire entre deux objets. Pourtant, lorsque le nombre et/ou la taille de ces objets devient grand, cette simple tâche va déjà nécessiter la mise en oeuvre de stratégies efficaces et le recours à des simplifications supplémentaires.

Répétitions Souvent les termes répétitions et redondances sont interchangeables. Dans ce travail, nous définissons une répétition comme une généralisation de la redondance au cas non exact. Cette notation peut paraître contre intuitive, elle correspond à l'utilisation du terme répétition au sens musical du terme. Soit une séquence de symboles $X = \{x_1, x_2, .., x_k\}$, on dit que X est répétitif si deux sous-séquences de X sont, sinon égales, du moins similaires. Par exemple, la reprise d'un motif musical, d'un refrain ou d'un mouvement est une répétition. Le discours parlé est également très souvent répétitif. La notion de répétition est plus proche de celle de similarité. En effet, deux interprétations différentes d'une même pièce musicale peuvent se comprendre comme une répétition, même si les signaux résultants de ces deux enregistrements sont alors très différents. Dès lors, les répétitions sont des structures plus complexes que les redondances qui nécessitent de définir des mesures de distance et de similarité.

Dans un article récent, BELLO [Bel11] donne une vue d'ensemble des méthodes de mise en évidence des similarités musicales à l'aide de descripteurs. Il s'appuie sur une grande quantité de travaux au sein de la communauté MIR [PLR02, AFLM06, BCTL07, MKC05, WCL09, SD09], tous basés sur

l'hypothèse que la musique est intrinsèquement structurée par la répétition. C'est un point de vue également développé dans la thèse de Cont sur l'anticipation musicale [Con08].

Ce niveau de structure pose déjà un certain nombre de problèmes de détection, mais reste relativement intuitif. Les redondances et répétitions sont la conséquence d'actes conscients, artistiques (musicaux) ou professionnels (diffusion radiophonique). Ce niveau est essentiel pour l'archivage, il conditionne en effet les deux facettes d'un système d'archivage :

- Indexation : les éléments redondants ou répétés doivent être localisés et étiquetés conjointement.
- Compression : les éléments redondants ou répétés doivent être codés conjointement.

Classes et regroupements A un niveau supérieur, une forme de structure très intéressante pour l'indexation (donc pour l'archivage) est le regroupement en classes. Pour les scènes sonores, par exemple, il pourra s'agir d'une classification en voix / musique [Sau96, SS97, AeqGC06, EMKPK00, PRAO02, PRAO03, RRVCMn+08, DIFM10], ou encore en genres musicaux [RBPU08, PK09, SN12]. Une collection de signaux est organisée en classes sur la base de similarités et de distances entre les descripteurs. Les systèmes mettant en oeuvre cette idée ont souvent recours à des classifieurs génériques de type Machines à Vecteurs Support (ou Séparateurs Vastes Marges) [RRE07, RR09, RRD10, MFRD10, AM11], ou sont basés sur des modèles de Markov caché [JE09], des algorithmes de k-moyennes, de k-plus proches voisins [JV08, ZKG10, JFF11] ou encore des algorithmes de Boosting [FELR11]. Pour améliorer les performances de ces approches, on utilise souvent une base de données annotée sur laquelle on peut entraîner les classifieurs.

Deux types de problèmes se posent à ce niveau structurel :

- Quelles distances sont pertinentes ? Les recherches vont du champ théorique de la géométrie de l'information [BL98] aux distances spécifiques temps-fréquence [MBF94, BF01] en passant par des modèles statistiques et des β-divergences [LOX06, FBD09].
- Comment définir les classes ? Sont-elles réellement disjointes ? et le cas échéant comment construire l'ensemble d'apprentissage ? Ces questions sont cruciales, notamment pour la séparation en genre musicaux [SN12].

Ces problématiques sont rappelées dans la thèse de Essid [Ess05]. Elles sont constitutives du champ de recherche de l'apprentissage automatique. Le lecteur intéressé pourra également se référer au livre de Duda, Hart et Storck [DHS96].

Autres structures Dans la littérature MIR, on recherche l'information structurelle sous différentes formes : la structure rythmique (p. ex. le tempo [BDA+05, RRD10]), la structure mélodique [AFLM06]. Notons que, si les exemples pris jusqu'ici sont liés à l'audio, la communauté de traitement des images rencontre les mêmes problématiques [BM95, Pau07, Gun10].

La recherche de structures, au delà des classes, notamment dans un flux de données/symboles, est un champ plus large qui rejoint celui de l'analyse du langage naturel. A ce niveau, il faut envisager un lexique, une syntaxe et une ontologie [RA07].

Plus loin on ira dans la prise en compte des couches de structures, meilleure sera l'indexation proposée. Aussi un système d'archivage idéal doit être capable de détecter tous ces niveaux. Malheureusement, on voit bien que plus une structure est profonde, plus sa détection – et même sa caractérisation – est complexe, voire s'apparente à un problème mal posé. Aussi dans le contexte de cette étude, nous n'envisagerons pas de structures au delà des classes. Il est de toute façon illusoire de

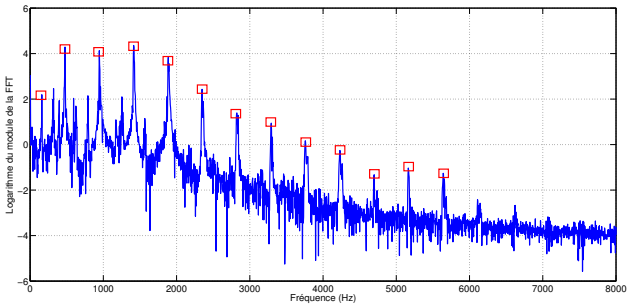

FIGURE 6.1.2: Spectre (logarithme du module de la transformée de Fourier) d'une scène sonore composée d'une note de trompette. Une représentation parcimonieuse de ce spectre peut être obtenue par la donnée des couples (fréquence,amplitude) des pics signifiés en carrés rouges. La structure harmonique de ces pics permet de réduire encore la quantité d'information nécessaire à la donnée de la fondamentale f_0 et des amplitudes.

chercher à utiliser un modèle de langage sur des données aussi variées que les scènes sonores peuvent l'être.

C'est d'ailleurs cette impossibilité de l'utilisation de ces modèles grammaticaux sur des données sonores de grandes tailles qui va nous dissuader d'utiliser une stratégie dite "top-down", de haut en bas pour la mise à jour des différents niveaux de structure. La logique inverse, dite "bottom-up" est la suivante : la détection des redondances et similarités est nécessaire à la mise en place d'un système de classification. De même, l'analyse syntaxique ne peut se faire que sur un lexique de dimension réduite, c'est à dire *a posteriori* d'une phase de segmentation/classification.

Mais cette vision n'est pas la seule possible. On peut en effet arguer du fait que la détection des similarités passe forcément par le calcul d'une distance et constitue en soi une forme de classification. Les structures peuvent ainsi s'envisager sous une forme imbriquée, en "poupées russes", sous une forme séquentielle (p.ex. un décodage acoustico-phonétique suivi d'un modèle de language pour le traitement de la parole), ou encore sous des formes plus complexes et imbriquées.

6.1.2 Exploitation des structures

Au vu de la variété de structures existantes, on ne sera pas surpris de l'étendue des applications possibles en traitement des signaux. Au niveau le plus bas, les similarités et autres redondances facilitent le codage des signaux.

Compression de signaux structurés Une bonne intuition de l'avantage qu'offre la structure pour la compression se trouve si l'on considère un signal musical, par exemple un signal $x \in \mathbb{R}^N$ dont le logarithme du module de la transformée de Fourier discrète $\hat{X} \in \mathbb{C}^N$ est donné en Figure 6.1.2. L'énergie de ce signal est concentrée en un nombre réduit de pics en rapport harmonique les uns aux autres. Plutôt que d'approcher x par la sélection des m plus grands coefficients de $\hat{X}$ – ce que ferait un codeur parcimonieux classique – une compression efficace tirera profit de la structure harmonique qui fixe les fréquences des partiels à être multiples d'une fréquence fondamentale. C'est le cas du codage paramétrique utilisé pour coder la parole ou la musique à bas débit (*p.ex.* le codeur MPEG-4 HILN[PM00] ou des codeurs basés sur une représentation parcimonieuse [EC03, LDCR07, PNZTL11, RV10]).

En reprenant le principe du codage parcimonieux décrit au chapitre 4, une approximation à m-termes de x dans un dictionnaire $\mathcal{D}$ de taille M donne lieu, après quantification des coefficients, à un couple $(\Gamma^m, \hat{\boldsymbol{\alpha}}_m)$ de vecteurs d'indices et d'amplitude. Toute connaissance additionnelle sur la structure de x diminue l'entropie de Γ^m et peut donc permettre de réduire le débit nécessaire à sa transmission.

Codage distribué Le paradigme du codage distribué pousse plus loin encore l'utilisation des structures (en particulier des similarités) pour la compression de données. Il se base pour cela sur l'information mutuelle $I(X,Y)$ de deux variables X et Y discrètes, dans un alphabet fini, et se résume ainsi :

Si l'on a une connaissance de la corrélation entre X et Y alors on peut réduire le débit nécessaire à leur codage. En particulier un résultat important de Slepian et Wolf [SW73] stipule que :

$$R_X \geq H(X|Y) = H(X) - I(X,Y) \tag{6.1.1}$$

$$R_Y \geq H(Y|X) = H(Y) - I(X,Y) \tag{6.1.2}$$

$$R_X + R_Y \geq H(X,Y) = H(X) + H(Y) - I(X,Y) \tag{6.1.3}$$

où $H(X)$ est l'entropie de X, R_X est le débit nécessaire pour transmettre X, $H(X|Y)$ l'entropie conditionnelle de X sachant Y et $H(X,Y)$ l'entropie jointe de X et Y. Les équations (6.1.1) et (6.1.2) expriment le fait que le codage de l'une des variables sachant l'autre est moins cher que son codage sans cette connaissance, et l'équation (6.1.3), que le codage des deux variables simultanément nécessite aussi un débit amoindri.

Dans une application de codage/décodage de X en présence d'information complémentaire Y, corrélée à X connue au codeur et au décodeur, l'intérêt est immédiat, le codeur peut coder X sans pertes avec un débit $H(X|Y)$. Un cas de figure moins intuitif est le codage de X sans l'information Y, connue uniquement du décodeur. Le théorème de Slepian-Wolf prouve que le débit nécessaire pour coder X sans pertes est, là aussi, $H(X|Y)$. La simple connaissance de la distribution jointe de X et Y, sans connaissance explicite de Y autorise un débit égal au cas de figure précédent.

Ce résultat très important de la théorie de l'information a été étendu au cas de sources continues, notamment pour deux vecteurs aléatoires Gaussiens et au codage avec pertes par Wyner et Ziv [WZ76]. Csiszar et Körner [CK80] ont prolongé ce résultat au cas d'une collection de signaux peu après mais son utilisation pratique n'est apparue que récemment. A cela deux explications :

- Les bornes théoriques de Wyner et Ziv n'ont été atteintes en pratique que vers la fin des années 90 [ZS98] et début des années 2000 [PR03]
- Les cas pratiques où la compression de données présentant de fortes corrélations est nécessaire sont également apparus assez récemment, avec l'explosion de la quantité de données numériques.

Parmi les premiers champs d'application, on trouve le codage vidéo [GaARRM05], l'imagerie hyperspectrale et satellitaire [Yeu99, MBAG07] et même la compression des signaux audio multicapteurs [MC09]. Dans les 10 dernières années, ce paradigme a été étendu au Compressed Sensing [BDW+05, ZMW+10] avec des résultats prometteurs.

Dans le cas d'une collection d'objets redondants, le codage distribué permet de réduire considérablement les coûts de stockage. Si cela se comprend facilement pour les cas de redondances exactes (nul besoin de stocker deux fois un objet répété à l'identique, il suffit de le coder une fois et de lui attribuer un indice dans une table pour répertorier ses occurrences), le codage distribué offre également

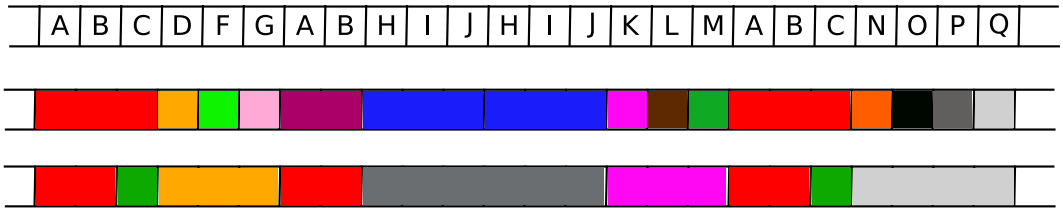

FIGURE 6.1.3: Problème de la segmentation de flux. La séquence de symboles contenant des redondances peut être segmentée de plusieurs façons différentes. Deux exemples sont représentés en couleurs.

un cadre théorique intéressant pour la compression de scènes sonores répétitives. Le débit théorique du codage joint d'une collection d'objets *similaires* peut être significativement réduit.

Codage prédictif et différentiel L'idée d'utiliser la corrélation à court terme d'un flux audio pour diminuer la quantité d'information nécessaire à son codage est au coeur du domaine du codage prédictif. De très nombreux codeurs audio utilisent ce principe, en particulier pour le codage de la parole. Etant donnée une connaissance de la valeur d'un signal à un instant donné et de ses propriétés (p.ex. ses régularités), on peut réaliser une prédiction sur sa valeur à l'instant suivant. Cette prédiction pouvant également être réalisée au niveau du décodeur, seule l'information de la différence entre la valeur réelle et la prédiction nécessite d'être transmise. Or celle-ci peut être réduite.

Toutes ses techniques sont basés sur les régularités et les corrélations à court terme des signaux audio. Dans ce travail, nous nous intéressons à des échelles de corrélation beaucoup plus importante.

Segmentation de flux Nous nous donnons la définition suivante : un flux est une séquence ininterrompue de données, constituée d'objets dont les contours ne sont généralement pas connus à l'avance. Dans une optique d'archivage, tant pour la compression que pour l'indexation, le découpage du flux en motifs délimités est une étape primordiale. Les structures internes des flux de données fournissent l'information nécessaire à ce découpage.

Malheureusement, ce problème est mal posé. En l'absence de contraintes supplémentaires (*p.ex.* parcimonie sur le nombre de motifs), il existe en effet un grand nombre de solutions au problème, comme illustré en Figure 6.1.3.

Avant de présenter comment régulariser ce problème, nous devons présenter plus en détails le concept et les outils de parcimonie structurée.

6.2 Représentations Parcimonieuses Structurées

6.2.1 Formalisation

On peut distinguer deux types de formalisation de la structuration des signaux, le premier basé sur une redéfinition de la contrainte de la parcimonie et le second basé sur une modélisation du signal. La première approche va privilégier une reformulation convexe et les méthodes d'optimisation appropriées tandis que la seconde s'attache à définir des algorithmes gloutons.

Contraintes de parcimonie structurée Rappelons le problème de représentation parcimonieuse. Soient un signal $x \in \mathbb{R}^N$ et un dictionnaire $\mathcal{D}$ de M atomes d_i. Le problème de reconstruction sous

contrainte de parcimonie s'écrit :

$$\min_{\alpha \in \mathbb{R}^M} \|x - \sum_{i=0}^{M-1} \alpha_i \cdot d_i\|^2 \text{ soumis à } \mathcal{P}(\alpha) \leq L \tag{6.2.1}$$

où $\mathcal{P}(\alpha)$ est une fonctionnelle de parcimonie sur le vecteur α. Les normes et pseudo-normes ℓ_p, souvent utilisées comme contrainte $\mathcal{P}(\alpha)$, sont transparentes à la notion de structure. Prenons par exemple, les vecteurs $x = [0, a, b, 0, c, d, 0]$ et $y = [0, 0, 0, a, b, c, d]$ ou a, b, c et d sont des réels non nuls. Alors quelle que soit la valeur de p choisie, on aura $\|x\|_p = \|y\|_p$. La position des éléments nuls (ou indifféremment des éléments non nuls) n'a pas d'influence sur la mesure ℓ_p. Pourtant, la prise en compte de cette information supplémentaire peut permettre de réduire l'entropie de y et du même coup son débit théorique.

Si l'on veut intégrer une telle connaissance *a priori*, il faut définir des fonctionnelles adaptées à une contrainte de parcimonie structurée. En particulier, comme illustré par le vecteur y, il peut être intéressant de favoriser l'émergence de solutions dans lesquelles les éléments non nuls sont regroupés. Pour cela, il faut pouvoir exprimer des contraintes par groupes de coefficients.

Une extension du LASSO, le Group-LASSO [YL06] s'attache à la résolution de ce problème. Il est initialement proposé pour résoudre un problème de régression à J facteurs :

$$x = \sum_{j=1}^{J} Y_j \alpha_j + \epsilon$$

où $x \in \mathbb{R}^N$, $Y_j \in \mathbb{R}^{N \times n_j}$ est une matrice correspondant au j-ième facteur, $\alpha_j \in \mathbb{R}^{n_j}$ un vecteur de coefficients de taille n_j et $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$. Le découpage des coefficients en plusieurs vecteurs définit les groupes et permet de poser des contraintes différentes sur chacun. Ces dernières s'appuient sur une norme définie pour toute matrice carrée K symétrique, définie positive par :

$$\|x\|_K = \left(x^T K x\right)^{1/2} \tag{6.2.2}$$

En utilisant un jeu de J matrices K_j, le Group-LASSO propose une solution parcimonieuse structurée :

$$\boldsymbol{\alpha}^{G-LASSO}(\lambda) = \min_{\boldsymbol{\alpha}=\{\alpha_1..\alpha_J\}} \left(\frac{1}{2} \|x - \sum_{j=1}^{J} Y_j \alpha_j\| + \lambda \sum_{j=1}^{J} \|\alpha_j\|_{K_j} \right) \tag{6.2.3}$$

où $\lambda \geq 0$ est un paramètre d'optimisation. Dans [YL06], Les auteurs proposent d'utiliser des matrices identités comme noyaux K_j. Cette approche revient à utiliser une norme mixte ℓ_1/ℓ_2. Une généralisation de cette approche consiste à utiliser une norme mixte ℓ_1/ℓ_q comme fonctionnelle de parcimonie structurée soit :

$$\mathcal{P}(\alpha) = \sum_{j=1}^{J} \|\alpha_j\|_q \tag{6.2.4}$$

Le cas de figure particulier de groupes de dimensions égales $n_j = n$ offre l'avantage de s'exprimer facilement sous une forme matricielle comme nous le verrons au chapitre 8. Il se comprend alors comme un problème d'approximations parcimonieuses simultanées. Rakotomamonjy [Rak11] propose une étude comparative des différents algorithmes traitant ce cas particulier. La formulation du Group-LASSO et ses garanties théoriques [RF08] supposent une partition du vecteur de coefficients α en groupes disjoints. Une extension au cas de groupes superposés est proposée par Jenatton *et al* [JAB09]. Le lecteur intéressé pourra également se référer à [BJM11] pour une revue de ces méthodes. Dans ses travaux de thèse, Lefevre [LBF11] utilise ce type de modèle pour la factorisation de spectrogrammes de scènes sonores.

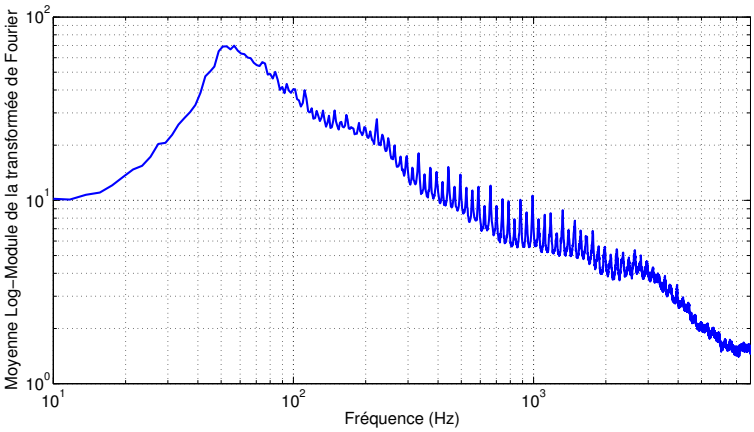

FIGURE 6.2.1: Moyenne du logarithme du module de la transformée de Fourier de 1000 extraits radiophoniques de 512 ms.

Modèles de parcimonie structurée Dans la formulation *classique* des représentations parcimonieuses, le vecteur de coefficient α porte à lui-seul une double information :

- La localisation des éléments non nuls.
- Le poids des éléments non nuls.

Les structures telles que nous les définissons dans ce travail, jouent un rôle uniquement sur la localisation des coefficients. Nous avons vu qu'une première approche pour définir des fonctionnelles de parcimonie pertinentes, consiste à partitionner l'ensemble des éléments. Afin de les mettre en valeur et *in fine,* de mieux les prendre en compte, nous pouvons envisager une formulation plus générale des représentations parcimonieuses :

$$x = \sum_{i=0}^{M-1} s_i \cdot w_i \cdot d_i + \epsilon \tag{6.2.5}$$

où l'information de structure est désormais séparée dans un vecteur de booléens $s \in \{0,1\}^M$ des pondération $w \in \mathbb{R}^M$. La formulation matricielle est :

$$\boldsymbol{x} \approx \boldsymbol{\Phi} \cdot (\boldsymbol{w} \odot \boldsymbol{s}) \tag{6.2.6}$$

ou $\odot$ dénote le produit terme à terme. On peut dès lors modéliser la structure de façon indépendante, en travaillant sur $\boldsymbol{s}$. Un tel modèle de signal se retrouve fréquemment dans les travaux récents, il apparaît par exemple dans les travaux de KOWALSKI et TORRÉSANI [KT08] et plus récemment dans [DHD12].

Étant donnée la nature booléenne de $\boldsymbol{s}$, il est naturel de lui associer un modèle de Bernoulli en l'absence de structure :

$$p(\boldsymbol{s}) = \prod_{i=0}^{M-1} p(s_i) \text{ avec } s_i \sim Ber(p_i) \tag{6.2.7}$$

où les probabilités p_i peuvent être choisies égales ou traduire un biais connu de la décomposition. Par exemple, la plupart des scènes sonores, musicales ou autres, présentent un profil énergétique avec un biais sur les basses fréquences. Il suffit pour s'en convaincre d'observer la moyenne des modules des spectres d'une collection de scènes sonores comme présentée Figure 6.2.1.

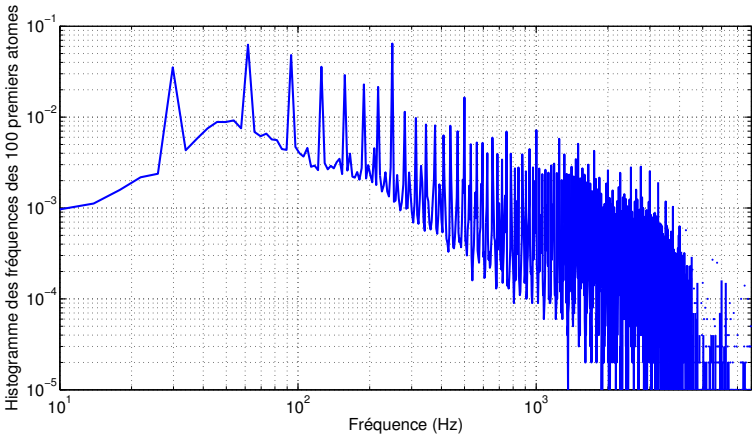

FIGURE 6.2.2: Histogramme de la répartition fréquentielle des 100 premiers atomes d'une décomposition par MP sur un dictionnaire 3xMDCT de 1000 segments de scènes sonores radiophoniques de 512 ms. Les pics sont dus à la mauvaise résolution fréquentielle des atomes de la base la plus courte.

Il est possible de fixer ou d'apprendre la distribution des probabilités p_i sur un ensemble de données. Mais il s'agira d'un simple biais spécifique à chaque atome. La Figure 6.2.2 présente la répartition fréquentielle des 100 premiers atomes choisis par MP dans une union de bases MDCT. Ce profil est celui obtenu pour 1000 extraits de scènes sonores radiophoniques. Il est intéressant de voir que certaines fréquences apparaissent sur-représentées. La base d'apprentissage comporte essentiellement de la musique populaire occidentale dans laquelle on constate une prédominance de grille d'accords standard type La Mineur- Fa - Do - Sol.

Si on veut prendre en compte les structures, il faut étudier les corrélations entre les s_i. Les structures les plus fréquemment rencontrées dans la littérature sont les corrélations locales, de voisinage, entre atomes, qui donnent lieu à une modélisation sous forme de molécules. On trouve par exemple pour l'analyse de scènes musicales des molécules tonales et/ou transitoires [Dau06], avec une structuration hiérarchique [JVF06] ou encore un modèle de molécules harmoniques [GB03, LDCR07, LVRD08].

KOWALSKI et TORRESANI [KT08] proposent un modèle explicite de structure de Bernoulli hiérarchique qui introduit des dépendances entre les atomes voisins dans le plan temps-fréquence. En particulier, s_i suit toujours une loi de Bernoulli mais conditionnellement à une variable indicatrice du temps, qui suit elle même une loi de Bernoulli. Malgré la complexité additionnelle introduite par cet empilement, il est possible d'estimer les paramètres du modèle.

Cette idée de prendre en compte des dépendances au niveau local est intuitivement séduisante et correspond assez bien en pratique au type de corrélations observées. La Figure 6.2.3 présente des profils de co-occurence observés sur des données réelles (20000 extraits radiophoniques de 3 secondes, représentés chacun par 100 atomes). Les graphiques montrent pour un atome référent, centré en temps et en fréquence, la répartition des atomes qui sont sélectionnés en même temps que lui par un MP standard sur une union de 3 bases MDCT, stoppé après 100 itérations. On peut faire les remarques suivantes :

- Les corrélations les plus importantes sont observées dans un voisinage temps-fréquence immédiat, ce qui tend à valider les modèles moléculaires type [Dau06] et [KT08] .
- Sur les atomes de grandes échelles (bien localisées en fréquence), on observe des corrélations harmoniques (notamment à l'octave). Les extraits sont ici choisis aléatoirement dans une base

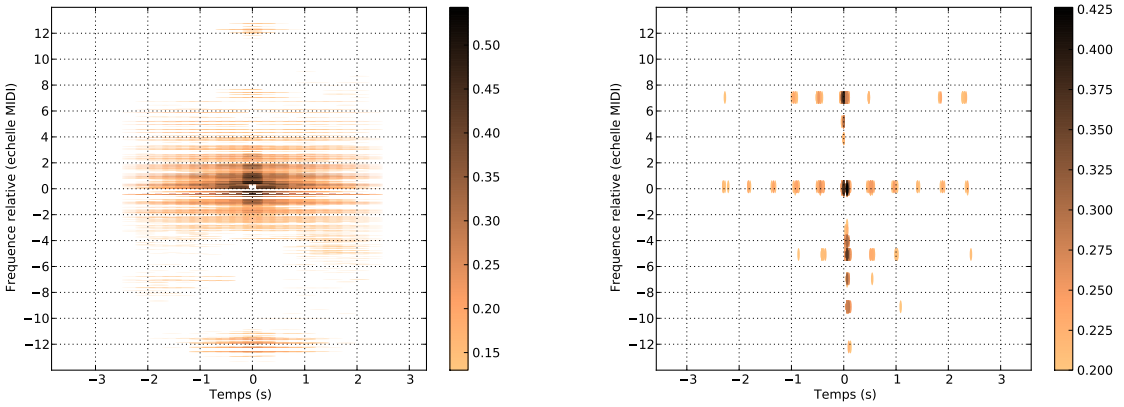

FIGURE 6.2.3: Répartition dans le plan temps-fréquence relatif (en échelle fréquentielle MIDI : une octave=12 indices) des atomes co-occurents parmi les 100 premiers sélectionnés par MP dans une union de 3 bases MDCT. Teinte : fréquence de co-occurrence normalisée, appris sur 20000 extraits radiophoniques de 3 secondes. Gauche : atomes de longueur 512 ms. Droite : atomes de longueur 128 ms.

de données radiophoniques contenant essentiellement de la musique populaire occidentale. Ces observations sont à rapprocher des modèles harmoniques [GB03, LVRD08].

—∘ Sur les atomes d'échelles plus courtes, on observe, en plus des corrélations harmoniques, des corrélations périodiques temporelles. Ces observations sont effectuées sur une collection non annotée de musiques populaires diverses, mais semblent indiquer une prédominance d'un tempo (autour de 120 bpm) dans cette collection. Ce type de structure est plus difficilement décelable dans la littérature.

Cette connaissance a priori de la distribution des atomes sélectionnés permet d'envisager d'autres modèles, comme des modèles de Markov cachés [FTDG08] ou des *Conditionnal Random Fields* [JE09]. Ces modèles peuvent être généralisés sous la forme d'une machine de Boltzmann, réseau de neurones stochastiques particulier, développé par HINTON *et al* dans les années 80[HG97]. La distribution du support est alors :

$$p(\boldsymbol{s}) \propto \exp\left(\boldsymbol{b}^{\boldsymbol{T}}\boldsymbol{s} + \boldsymbol{s}^{\boldsymbol{T}}\boldsymbol{W}\boldsymbol{s}\right) \tag{6.2.8}$$

où $\boldsymbol{W}$ est une matrice symétrique dont les éléments diagonaux sont nuls et $\boldsymbol{b}$ est un vecteur de biais (par exemple fréquentiel comme en Figure 6.2.1). Ce modèle a en particulier été considéré par PELEG *et al* [PEE12] puis DRÉMEAU *et al* [DHD12], les deux contributions se différenciant par la méthode d'optimisation choisie ensuite.La machine de Boltzmann présente l'intérêt de généraliser beaucoup de modèles probabilistes connus. Cette propriété est particulièrement utile lorsqu'on ne dispose que de peu d'information a priori sur les structures des signaux. Dans le cas contraire, la machine de Boltzmann peut se simplifier (notamment, W peut devenir parcimonieuse, par exemple elle est poly-diagonale dans le cas du modèle de Markov caché) et l'estimation de ses paramètres s'en trouve allégée (dans [DHD12], les auteurs font appel à un algorithme de Metropolis-Hastings, dans [PEE12], l'apprentissage est envisagé).

Au vu des exemples présentés Figures 6.2.1 et 6.2.3, il est tentant de vouloir apprendre $\boldsymbol{b}$ et $\boldsymbol{W}$ à partir de scènes sonores réelles. Nous avons tenté d'adapter les procédures proposées dans [DHD12] à des scènes sonores réelles et avons été confrontés à un certain nombre de difficultés, en particulier

liées à la dimension des données. Dans ce cas, en effet, l'utilisation de Metropolis-Hastings s'avère trop complexe. Comme dans tout apprentissage, si l'espoir d'obtenir une estimation pertinente pour des données homogènes (*p.ex.* une collection de sons instrumentaux, ou enregistrements d'un seul locuteur) est permis, il se réduit fortement lorsque l'on considère une base d'archives sonores plus variées. Dans ce cas, la diversité des structures rencontrées n'autorise plus une modélisation simple et malléable.

Autres formes de parcimonie structurée Dans un paradigme récent, (*Robust PCA* [CLM09]) Candès *et al* proposent une factorisation originale d'une matrice X en deux composantes :

$$X = L + S \tag{6.2.9}$$

avec L une matrice de rang faible (obtenue dans l'esprit d'une analyse en composantes principales) et S une matrice parcimonieuse. La matrice S sert à modéliser l'ensemble des *outliers*, c'est à dire des éléments mal capturés par le modèle PCA classique. Un modèle augmenté d'une modélisation du bruit et dénoté *Stable Principal Component Pursuit* est proposé dans [ZT11]. Ce modèle a récemment donné des résultats spectaculaires sur des traitements d'image et de vidéo [ZGLM12] mais également en audio, sur de la séparation de sources [HCSHJ12].

6.2.2 Invariance et robustesse

Avec le développement des représentations parcimonieuses est apparu un nouveau problème. Supposons qu'on dispose d'une représentation structurée $R(X)$ d'un signal X, la représentation d'une transformation $T(X)$ sera-t-elle également structurée et comment ? Alternativement, on peut se demander à quelles transformations du signal la structure de la représentation est robuste. L'émergence de travaux récents [ZGLM12, SM12] nous amène à considérer ces questions.

Invariance par translation Formellement, l'invariance par translation d'un opérateur Φ de $\boldsymbol{L^2}\left(\mathbb{R}^N\right)$ dans un espace de Hilbert $\mathcal{H}$ suppose que quel que soit $f \in \boldsymbol{L^2}\left(\mathbb{R}^N\right)$ et $T_c(f) = f(x-c)$ une translation de f, alors $\Phi(T_c(f)) = \Phi(f)$.

Dans le cas d'une représentation parcimonieuse, le problème se pose plutôt de la manière suivante : Soit $\tilde{f}_m$ l'approximation à m-termes de f – obtenue par exemple par MP – dans un dictionnaire $\mathcal{D}$ de M atomes d_i ($\tilde{f}_m = \sum_{i\in\Gamma^m} \alpha_i d_i$). La représentation $\tilde{f}_m(\tau)$ du signal translaté $f * \delta_\tau$ s'écrit :

$$\tilde{f}_m(\tau) = \sum_{i\in\Gamma^m_\tau} \alpha_i^\tau d_i$$

L'invariance de la représentation se vérifie si $\forall i \in \Gamma^m_\tau, \exists j \in \Gamma^m, d_i = d_j * \delta_\tau$ et $\alpha_i = \alpha_j$. Autrement dit, le support de la représentation est covariant par translation. Une condition nécessaire pour qu'une décomposition sur $\mathcal{D}$ puisse être invariante par translation est donc :

$$\forall d \in \mathcal{D} \forall \tau \in \mathbb{Z}, d * \delta_\tau \in \mathcal{D} \tag{6.2.10}$$

Cette propriété est particulièrement intéressante dans des dictionnaires structurés de type union de bases orthonormales. Ce type de dictionnaire est d'ailleurs fréquemment appelé invariant par translation (*Shift-Invariant*) et cette propriété est à la base de nombreuses optimisations [KG06, BD06, MGBV09]. Lewicki et Sejnowski [LS99] montrent qu'une manière de garantir cette invariance pour

des signaux discrets est de construire un dictionnaire très redondant dans lequel un atome est répliqué à toutes les positions temporelles. Un dictionnaire construit sur un repère de Gabor comme décrit en 4.3 vérifie ce schéma.

Une autre façon de rendre une représentation robuste aux décalages temporels est d'en extraire l'information de phase. Ainsi, le module de la transformée de Fourier est invariant par translation.

L'invariance par translation est nécessaire pour la détection de redondances dans des scènes sonores en raison de l'arbitraire fréquent du découpage en trames d'analyse. Ce découpage accélère considérablement les traitements, mais s'apparente à un sous-échantillonnage du dictionnaire complet. Les mesures de similarité dans le domaine de la représentation, qui utilisent ce type de sous-dictionnaire (voir notamment Jost *et* Vandergheynst [JV08] sur des images et Sturm *et* Daudet [SD09] sur des scènes sonores) sont en général affectées par des translations temporelles.

L'invariance par translation fréquentielle est une propriété mise en avant pour les tâches de transcription automatique de musique, et plus généralement lorsqu'un peigne harmonique est utilisé pour modéliser le spectre de différentes notes. Ainsi Hennequin *et al* [HBD11b] utilisent une représentation invariante par homothétie fréquentielle sur des spectrogrammes. En utilisant une échelle de fréquence logarithmique (p.ex. avec une transformée à Q-constant) on retrouve une propriété d'invariance par translation [SRS08, FLBR12]).

Invariance par rotation Cette propriété se retrouve essentiellement dans le traitement d'images et de vidéo. Deux approches récentes s'y attellent : celle de Sifre et Mallat [SM12] basée sur des opérateurs de dispersion (*Scattering Operators*) et celle de Zhang *et al* [ZGLM12] qui utilise des méthodes de type *Robust PCA* déjà mentionnées plus haut.

Invariance aux autres transformations Un type de déformations plus complexe se rencontre fréquemment sur des collections de scènes sonores. C'est l'effet obtenu lorsqu'on accélère ou ralentit la lecture d'un enregistrement, entraînant une modification des échelles temporelles et fréquentielles. Cette déformation, connue sous le nom de *pitch-shifting* ou modification des hauteurs, peut considérablement perturber la structure d'une représentation.

Une étude récente de Mallat [Mal12] montre que certains opérateurs – en particulier les opérateurs de dispersions, construits sur une cascade de transformées en ondelettes rectifiées – sont robustes à l'action de petites déformations et invariants par translation et rotation.

6.3 Applications

Les structures des signaux et de leurs représentations présentent un intérêt significatif dans une tâche d'archivage. Les redondances et similarités peuvent être utilisées à la fois pour la compression de scènes sonores et pour la structuration de flux audio. Le regroupement en classes est une première forme d'indexation, sur laquelle on peut en construire d'autres, de plus haut niveau encore. Aussi est-il essentiel de pouvoir tout d'abord mettre à jour ces structures pour pouvoir, dans un second temps, les exploiter pour l'indexation.

Dans cette thèse, nous avons voulu voir s'il était possible de détecter différents niveaux de structures à l'aide d'un seul type de représentation parcimonieuse. En particulier nous avons voulu étendre les travaux de Ravelli [RRD10] sur l'indexation dans le domaine de la représentation, à des cas de figures plus complexes et/ou de dimensions plus grandes.

Au chapitre suivant, nous nous intéressons au problème de détection de redondances et de similarités dans des scènes sonores musicales à l'aide de représentations parcimonieuses obtenues par algorithme glouton. Nous l'adaptons au problème de reconnaissance d'empreintes que nous évaluons sur une tâche de structuration de flux radiophonique par détection d'objets récurrents.

Dans un second temps, au chapitre 8, nous illustrons une possibilité d'exploitation de ces structures, dans le cadre d'un problème de séparation de sources répétitives, tout en utilisant le même type d'algorithme et de dictionnaire.

Chapitre 7

Détection de redondances et similarités.

Les méthodes visant à mettre au jour les redondances et similarités sont multiples et peuvent, selon l'échelle de signal considérée, varier fortement. Dans ce chapitre, nous présentons des méthodes de détection de motifs récurrents à différentes échelles, toutes basées sur une même décomposition parcimonieuse des signaux dans un dictionnaire constitué d'une union de bases MDCT.

La première étape va donc consister à passer en revue les différentes métriques, distances et autres mesures de similarités qu'il est possible de construire sur les représentations choisies. Si certaines se calculent directement à partir des vecteurs de coefficients, ou des supports parcimonieux, nous verrons que d'autres nécessitent le calcul – simple – de descripteurs. En nous basant sur des concepts issus de la théorie du codage distribué, nous proposons une nouvelle métrique et un algorithme simple qui effectue en une seule passe une mesure de similarité et une compression basée sur une factorisation des supports atomiques. Les avantages de cette approche sont mis en exergue sur des exemples simples, et les possibilités de son utilisation sur des débits plus importants sont discutées.

Dans un second temps nous verrons une application au problème de segmentation de flux radiophonique basée sur la détection de motifs récurrents par comparaison d'empreintes. Nous présentons un système développé en collaboration avec S.Fenet, dans lequel nous incluons des empreintes construites à partir de représentations parcimonieuses. Nous présentons également les résultats obtenus avec ce système lors d'une campagne européenne d'évaluation.

7.1 Détection de motifs récurrents par calcul de similarités

7.1.1 Objets et motifs récurrents

Il y a récurrence dès lors qu'un objet multimédia apparaît au moins en deux endroits d'une collection, ou à deux positions dans un flux de données. Les flux radiophoniques et télévisuels sont, en ce sens, très redondants (publicités, rediffusions). En travaillant sur des données de radio commerciale, nous avons pu constater que le nombre moyen de diffusions d'une même chanson au cours d'une journée atteint facilement la dizaine, soit une diffusion environ toutes les deux heures. La redondance d'une publicité peut excéder très largement ce chiffre.

Si l'on prend en compte les versions différentes d'un même objet : différents enregistrements audio d'un même concert, les versions *live* ou *studio*, les différentes captures du même évènement (voir par exemple les films amateurs du discours inaugural de Barack Obama[CE10]), alors la grande majorité des collections audiovisuelles contiennent des récurrences.

A plus petite échelle, il y a récurrence dès lors qu'une phrase musicale est rejouée, qu'un mot ou une phrase sont répétés par un ou plusieurs locuteurs, ou qu'un motif rythmique est repris cycliquement.

On peut ainsi trouver des similarités à des échelles de plus en plus petites. La question se pose alors du choix de la plus petite échelle de redondance à considérer. Ce choix, forcément arbitraire, est dicté par la représentation choisie. Dans la majorité des cas, pour des données audio, l'échelle de redondance la plus petite est celle d'une trame de TFCT ou la longueur utile d'un filtre auto-récursif (de l'ordre de quelque dizaines de millisecondes). Lorsqu'une représentation parcimonieuse est utilisée, l'élément de plus petite échelle se trouve être un atome du dictionnaire.

Lorsqu'on travaille avec des symboles appartenant à un alphabet de taille réduite, il existe des méthodes reconnues pour la recherche de redondances (*p. ex.* dans des séquences ADN, ou dans des textes utilisant un alphabet latin) dont l'une des plus utilisées est celle de Lempel et Ziv [ZL77]. Toute une littérature de la détection de mots dans un tel contexte est disponible. Citons, parmi d'autres, les algorithmes facteurs d'oracle [ACR99, LLA01] utilisés par Cont sur des descripteurs audio [Con08].

Souvent ces méthodes, de même que la plupart de celles développées pour l'analyse du langage (arbres de suffixes, etc..) voient leur complexité augmenter avec la taille de l'alphabet considéré. De plus, s'il est relativement simple de définir la similarité entre deux chaînes de caractères ou deux séquence de symboles, définir la similarité de deux scènes sonores est plus complexes. Le cas d'une redondance parfaite se formalise facilement, mais celui des répétitions beaucoup moins. Deux scènes sonores peuvent être jugées similaires par un auditeur, même si leurs formes d'ondes respectives ne sont pas identiques.

Pourtant cette notion de similarité doit pouvoir s'apprécier sur les représentations de ces scènes sonores. Dès lors, il nous faut étudier les métriques qu'il est possible d'utiliser pour la détection de motifs récurrents dans des représentations parcimonieuses. En premier lieu, nous décrivons quelques mesures de similarité de scènes sonores dans le domaine de la représentation.

7.1.2 Distances dans le domaine de la représentation

Nous avons montré dans la deuxième partie de ce manuscrit qu'il était possible de construire, pour un signal $\boldsymbol{x} \in \mathbb{R}^N$ une approximation à m-termes $\tilde{\boldsymbol{x}}_m$ dans un dictionnaire $\boldsymbol{\Phi} \in \mathbb{R}^{N\times M}$ telle que $\tilde{\boldsymbol{x}}_m = \boldsymbol{\Phi}\boldsymbol{\alpha}_m$ où $\boldsymbol{\alpha}_m$ est un vecteur parcimonieux vérifiant $\|\boldsymbol{\alpha}_m\|_0 \leq m$. Nous nous intéressons désormais à la comparaison de $\tilde{\boldsymbol{x}}_m$ et $\tilde{\boldsymbol{y}}_m$, approximations respectives de $\boldsymbol{x}$ et $\boldsymbol{y}$ dans $\boldsymbol{\Phi}$ **:**

$$\tilde{x}_m = \sum_{\gamma\in\Gamma_x^m} \alpha_\gamma^x \phi_\gamma = \sum_{i=0}^{M-1} s_i^x w_i^x \phi_i \tag{7.1.1}$$

$$\tilde{\boldsymbol{x}}_m = \boldsymbol{\Phi}\boldsymbol{\alpha}_m^x = \boldsymbol{\Phi}\left(\boldsymbol{s}_m^x \odot \boldsymbol{w}_m^x\right) \tag{7.1.2}$$

$$\tilde{y}_m = \sum_{\gamma\in\Gamma_y^n} \alpha_\gamma^y \phi_\gamma = \sum_{i=0}^{M-1} s_i^y w_i^y \phi_i \tag{7.1.3}$$

$$\tilde{\boldsymbol{y}}_m = \boldsymbol{\Phi}\boldsymbol{\alpha}_m^y = \boldsymbol{\Phi}\left(\boldsymbol{s}_m^y \odot \boldsymbol{w}_m^y\right) \tag{7.1.4}$$

On dispose donc des vecteurs $\boldsymbol{\alpha}_m^x$ et $\boldsymbol{\alpha}_m^y$ ou alternativement des couples de vecteurs (supports binaires , coefficients réels) $(\boldsymbol{s}_m^x, \boldsymbol{w}_m^x)$ et $(\boldsymbol{s}_m^y, \boldsymbol{w}_m^y)$. Rappelons les objectifs, nous cherchons une mesure capable de mettre en évidence les similarités entre deux scènes sonores, en se basant sur leurs représentations dans un même dictionnaire.

Similarité basée sur une distance euclidienne Une façon simple de comparer deux vecteurs $\boldsymbol{\alpha}_m^x$ et $\boldsymbol{\alpha}_m^y$ est de mesurer la distance quadratique qui les sépare, en utilisant la norme euclidienne :

$$d_{EUC}(\tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m) = \|\boldsymbol{\alpha}_m^x - \boldsymbol{\alpha}_m^y\|_2 \tag{7.1.5}$$

ce qui dans le cas parcimonieux, s'écrit comme la somme de trois termes.

$$d_{EUC}(\tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m) = \sum_{\gamma \in \Gamma_x^m \bigcap \Gamma_y^m} |\alpha_\gamma^x - \alpha_\gamma^y|^2 + \sum_{\gamma \in \Gamma_x^m, \gamma \notin \Gamma_y^m} |\alpha_\gamma^x|^2 + \sum_{\gamma \in \Gamma_y^m, \gamma \notin \Gamma_x^m} |\alpha_\gamma^y|^2 \tag{7.1.6}$$

d_{EUC} s'annule lorsque $\tilde{\boldsymbol{x}}_m = \tilde{\boldsymbol{y}}_m$, ce qui indique (mais ne prouve pas) une redondance exacte $x = y$. En dehors ce cas de figure, cette mesure est peu informative, car elle considère chaque élément indépendamment. Or on sait qu'il existe des dépendances fortes entre les coefficients d'une représentation parcimonieuse.

Similarité en Cosinus Une façon de prendre en compte ces dépendances (étudiée notamment par Sturm et Daudet [SD09] se basant sur les travaux de Jost et Vandergheynst [JV08]) est d'utiliser une mesure de similarité en cosinus :

$$d_{COS}(\tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m) = \frac{\langle \tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m \rangle}{\|\tilde{\boldsymbol{x}}_m\| \|\tilde{\boldsymbol{y}}_m\|} \tag{7.1.7}$$

qui permet de distinguer les cas colinéaires ($d_{COS} = 1$) et opposé (-1) des vecteurs orthogonaux (0). Cette mesure est intéressante, car son calcul est facilité par la formulation :

$$d_{COS}(\tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m) = \frac{\sum_{\gamma_x \in \Gamma_x^m} \sum_{\gamma_y \in \Gamma_y^m} \alpha_{\gamma_x}^x \alpha_{\gamma_y}^y \langle \phi_{\gamma_x}, \phi_{\gamma_y} \rangle}{\|\tilde{\boldsymbol{x}}_m\| \|\tilde{\boldsymbol{y}}_m\|} \tag{7.1.8}$$

$$= \frac{(\boldsymbol{\alpha}_m^x)^T \cdot \boldsymbol{G} \cdot \boldsymbol{\alpha}_m^y}{\|\boldsymbol{\alpha}_m^x\|_2 \|\boldsymbol{\alpha}_m^y\|_2} \tag{7.1.9}$$

qui requiert une complexité limitée (de l'ordre de $\mathcal{O}(m^2)$) si on connaît la matrice de Gram $\boldsymbol{G} = \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ du dictionnaire, qui contient l'ensemble des M^2 produits scalaires entre atomes. Dans le cas de dictionnaires temps-fréquence de type union de bases orthonormales, une grande partie de ces produits sont nuls, la matrice de Gram est donc très creuse. La Figure 7.1.1 montre une telle matrice pour une union de 3 bases MDCT (16, 64 et 256 échantillons).

Le problème de cette mesure est sa sensibilité aux translations temporelles ainsi qu'aux déphasages, en particulier dans le cas de bases MDCT. Les projections sur des bases MDCT sont, en effet, réelles, et l'information de phase est contenue dans les coefficients α. Deux signaux déphasés et/ou retardés auront donc des représentations différentes et la mesure de leur similarité en cosinus ne sera pas unitaire. Il est possible de s'affranchir de cette contrainte en utilisant des dictionnaires invariants par translation, mais l'emploi de ceux-ci pose de nombreux problèmes pratiques (voir section 4.3.2). Une façon de limiter ce phénomène est de considérer une variante complexe de la MDCT, la *Modulated Complex Lapped Transform* (MCLT) proposée par Malvar [Mal99].

Similarité basée sur des distances binaires Une alternative pour s'affranchir des effets liés à la phase, est de comparer uniquement les supports binaires $\boldsymbol{s}_m^x$ et $\boldsymbol{s}_m^y$ des représentations. Une mesure de similarité des supports est alors donnée par :

$$d_{SUP}(\tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m) = \frac{(\boldsymbol{s}_m^x)^T \cdot \boldsymbol{G} \cdot \boldsymbol{s}_m^y}{\|\boldsymbol{s}_m^x\|_2 \|\boldsymbol{s}_m^y\|_2} \tag{7.1.10}$$

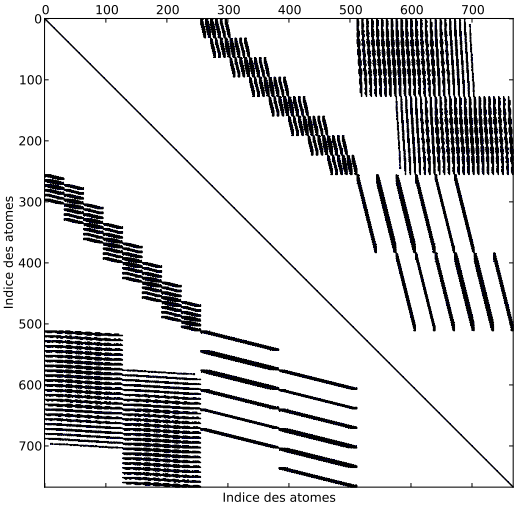

FIGURE 7.1.1: Profil de parcimonie de la matrice de GRAM d'un dictionnaire constitué d'une union de 3 bases MDCT (en noir les éléments non nuls).

En poussant l'idée encore un peu plus loin, on peut se passer entièrement de la matrice de GRAM, et définir des distances sur les vecteurs binaires. On peut par exemple utiliser la distance de Hamming, ou encore la similarité de Jacquard qui procède de la même idée mais offre l'avantage d'être normalisée :

$$d_{JAC}(\tilde{\boldsymbol{x}}_m, \tilde{\boldsymbol{y}}_m) = \frac{\|\boldsymbol{s}_m^x \odot \boldsymbol{s}_m^y\|_0}{\|\boldsymbol{s}_m^x \oplus \boldsymbol{s}_m^y\|_0} \tag{7.1.11}$$

où $\oplus$ est un opérateur de *ou exclusif*. C'est le ratio des cardinalités de l'intersection et de l'union des supports parcimonieux. Il est également possible de définir des distances binaires sur des objets dérivés du support. Nous détaillerons ce type de métriques plus loin, dans la section 2 de ce chapitre.

Similarité basée sur des distances d'information Une autre façon de mesurer la similarité de x et y est de quantifier leur information mutuelle $I(x, y)$. En l'absence de modèles sur leurs distributions respectives, le calcul direct de $I(x, y)$ est malaisé. En revanche il existe comme substitut d'autres distances d'information. BENNETT *et* LI [BL98] proposent de telles distances, dérivées de la notion de complexité de Kolmogorov pour la mesure de similarités entre objets numériques. Formellement, la distance d'information est définie à une constante logarithmique près comme :

$$ID(x, y) = \max\{K(x|y), K(y|x)\} \tag{7.1.12}$$

où $K(\cdot)$ est la complexité de Kolmogorov et $K(x|y)$ dénote la complexité conditionnelle, soit les ressources minimales requises par une machine universelle pour spécifier x étant donné y.

Leurs travaux sont une généralisation du théorème de SLEPIAN-WOLF pour le codage distribué et se comprennent intuitivement de la façon suivante. Si l'information mutuelle permet de diminuer le débit théorique pour le codage joint de x et y (rappelons que $H(x, y) - H(x) - H(y) = I(x, y)$) alors on peut mesurer de façon indirecte cette information mutuelle en mesurant un débit expérimental utilisant les similarités. On retrouve cette idée dans [HW07] et plus récemment dans l'article de BELLO

[Bel11] sous la forme d'une distance de compression normalisée :

$$DCN(x,y) = \frac{R(x,y) - \min\{R(x), R(y)\}}{\max\{R(x), R(y)\}} \tag{7.1.13}$$

où $R(\cdot)$ est un débit expérimental obtenu par compression avec un algorithme standard (*p.ex.* bzip2 dans le cas de [Bel11]). Cette mesure, contrairement à (7.1.12), est calculable, et offre de plus l'avantage d'être normalisée.

Distorsion minimale du codage joint Il y a, en réalité, deux façons symétriques d'utiliser Slepian-Wolf pour détecter des similarités : soit on fixe une distorsion nominale et on mesure le débit minimum nécessaire au codage joint (ou par exemple la distance de compression normalisée proposée ci-dessus), soit on fixe le débit nominal et on mesure la distorsion atteinte par codage joint. Nous proposons d'utiliser cette stratégie, et de fixer un débit minimal pour coder y étant donné x ou plus exactement sa décomposition $\tilde{x}_m$. Le principe est de mesurer la distorsion atteinte lorsqu'on approche y en utilisant uniquement les atomes de $\tilde{x}_m$ (support et amplitude), la mesure de similarité est alors :

$$D_0(x,y) = 10\log_{10}\left(\frac{\|\tilde{x}_m - y\|_2^2}{\|\tilde{x}_m\|_2^2}\right) \tag{7.1.14}$$

où en d'autres termes on mesure la capacité de la représentation de x à représenter y comme l'inverse d'un *Rapport Signal-à-Bruit*. Cette mesure n'offre pas de souplesse, la représentation de x est utilisée directement, sans aucune adaptation à y. Elle est, en conséquence, très sensible aux translations et petites déformations. Une manière d'introduire de la flexibilité est de construire une nouvelle approximation $\hat{y}_m$ de y, à partir de $\tilde{x}_m$, et de mesurer :

$$D_R(x,y) = 10\log_{10}\left(\frac{\|\hat{y}_m - y\|_2^2}{\|\hat{y}_m\|_2^2}\right) \text{ tel que } R(\hat{y}_m|\tilde{x}_m) \leq R \tag{7.1.15}$$

où $R(\hat{y}_m|\tilde{x}_m)$ est une mesure empirique analogue à la complexité conditionnelle. Elle quantifie la quantité d'information que l'on accepte de fournir pour spécifier $\hat{y}_m$ à partir de $\tilde{x}_m$. On peut comprendre $D_R(x,y)$ comme une mesure de la distorsion associée au débit $R(\hat{y}_m|\tilde{x}_m)$, et voir D_0 comme la distorsion pour un débit nul. D_0 ne peut servir que pour la détection de redondances exactes. Il est intéressant de noter qu'augmenter le débit augmente du même coup la sensibilité de D_R aux similarités moins évidentes. D'un autre côté, lorsque $R(\hat{y}_m|\tilde{x}_m)$ devient grand devant $R(\tilde{x}_m)$, la mesure D_R ne capture plus vraiment l'information mutuelle.

Nous forçons les supports de $\hat{y}_m$ et $\tilde{x}_m$ à être identiques, mais on autorise une variation de positionnement temporel des atomes. L'amplitude et la phase de ces atomes est, en revanche, conservée. Une fois cette règle acquise, nous proposons un moyen simple de fixer ce débit à une valeur pertinente. Le débit $R(\hat{y}_m|\tilde{x}_m)$ est simplement la quantité d'information nécessaire à la transmission de la séquence $\Theta = \{\tau^n\}_{n=0..m-1}$ des décalages atomiques :

$$R(\hat{y}_m|\tilde{x}_m) = \sum_{n=0}^{m-1} \log_2 \tau^n \tag{7.1.16}$$

d'où la possibilité de fixer la borne R_{max} à l'aide d'un décalage maximum τ_{max} :

$$R_{max} = m\log_2 \tau_{max} \tag{7.1.17}$$

L'algorithme 4, appelé Factorisation, présente la méthode envisagée :

Algorithm 4 Factorisation

Entrées: y , $\mathbf{\Phi}$, $\tilde{x}_m = \sum_{i \in \Gamma^m} \alpha_i \phi_i$, R_{max}
Sorties: $D_R(y, \tilde{x}_m)$

1: $\hat{y}_0 := 0$, $r_0 := y$,$n = 0$
2: **Pour** $i \in \Gamma^m$ **Faire**
3: Optimisation locale :
$\tau^n = \arg\max_{\tau \in [-\frac{\tau_{max}}{2}, \frac{\tau_{max}}{2}]} |\langle r_n, (\alpha_i.\phi_i * \delta_\tau) \rangle|$
4: Mise à jour :
Approximation : $\hat{y}_{n+1} \leftarrow \hat{y}_n + \alpha_i.(\phi_i * \tau^n)$
Résiduel $r^{n+1} \leftarrow r^n - \alpha_i.(\phi_i * \tau^n)$
5: $n \leftarrow n + 1$
6: **Fin pour**
7: $D_R(y, \tilde{x}_m) = 10 \log_{10} \left(\frac{\|\hat{y}_m - y\|_2^2}{\|\hat{y}_m\|_2^2} \right)$ tel que $R(\hat{y}_m | \tilde{x}_m) \leq R_{max}$

Métrique	Domaine	Score Redondance	Similarité	Dissimilarité
D_{EUC}	$[0, +\infty]$	0	Score faible	Score élevé
D_{COS}	$[-1, 1]$	1	Proche de 1 ou -1	Proche de 0
D_{JAC}	$[0, 1]$	1	Proche de 1	Proche de 0
ND_R	$[-\infty, +\infty]$	1	Proche de 1	Proche de zéro

TABLE 7.1.1: Tableau récapitulatif des métriques de distances utilisant des représentations dans le domaine parcimonieux

Le lecteur fera le rapprochement entre cet algorithme et la phase d'optimisation locale des atomes dans un algorithme MP adaptatif 3.3 page 44. De fait, il est possible avec cette méthode de comparer des représentations obtenues aussi bien par MP que par MP adaptatif ou même SASMP sur des dictionnaires temps-fréquence.

En exécutant cet algorithme (dans la pratique, on range les atomes par ordre décroissant d'amplitude pour minimiser la distorsion) on obtient deux choses : une approximation de y et une mesure de similarité entre x et y. Ce double résultat est intéressant dans la mesure ou il permet dans une même passe de trouver les similarités et de les exploiter dans un schéma de codage distribué.

Dans la pratique, il est intéressant de regarder la mesure normalisée :

$$ND_R(x, y) = \frac{-D_R(x, y)}{SRR(\tilde{x}_m)} = \frac{\log_{10} \left(\frac{\|\hat{y}_m\|_2^2}{\|\hat{y}_m - y\|_2^2} \right)}{\log_{10} \left(\frac{\|\tilde{x}_m\|_2^2}{\|\tilde{x}_m - x\|_2^2} \right)} \tag{7.1.18}$$

Cette valeur est proche de zéro lorsque le dénominateur est grand devant le numérateur, c'est à dire que la qualité de la reconstruction obtenue par factorisation est très inférieure à la qualité de l'approximation utilisée pour cette factorisation. A l'inverse, une factorisation efficace se traduit par des valeurs proches de 1. Il est intéresant de noter que la valeur 1 peut être dépassée. Il faut alors en déduire que la représentation de x est encore plus efficace à représenter y qu'elle ne l'était pour x.

7.1.3 Comparaisons

Pour illustrer les différences entre des mesures de similarités, il est courant de comparer les scores obtenus sur différents segment d'une scène sonore répétitive. Soit X un signal découpé en J trames x_j recouvrantes, la matrice de taille $J \times J$ contenant les scores de similarités obtenus pour chaque paire de segment (x_i, x_j) est appelée matrice de similarité de X.

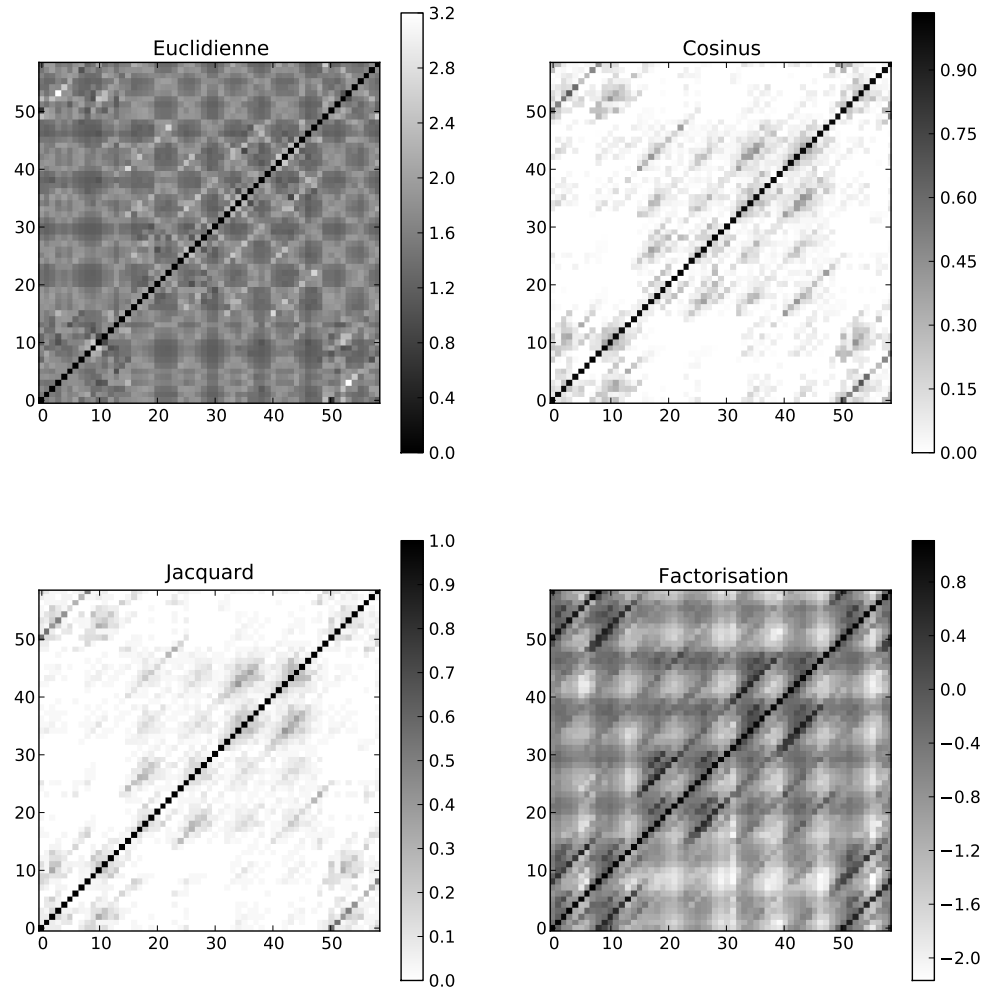

FIGURE 7.1.2: Matrices de similarités obtenues avec différentes métriques pour les 15 premières secondes du Prélude en ut (J.S BACH, Clavier bien tempéré BWV 846) interprétation de Glenn Gould. Scores obtenus avec des approximations MP adaptatif, stoppé après 100 itérations ou lorsqu'un SRR de 5 dB est atteint. La scène est découpée en 57 trames de 1 seconde, recouvrantes à 75%.

Nous choisissons donc une scène sonore comprenant des répétitions musicales, et sur chaque segment nous effectuons une décomposition par MP adaptatif dans une union de bases MDCT. Pour effectuer des mesures de similarités, des décompositions très superficielles suffisent. En pratique, nous considérons des trames d'une seconde, avec un taux de recouvrement de 75% entre trames voisines. Nous stoppons la décomposition lorsqu'un SRR de 5 dB est atteint, ce qui pour un signal simple de piano comme celui choisi, correspond à une centaine d'atomes au grand maximum par trame analysée.

La Figure 7.1.2 montre les matrices de similarités obtenues pour les quatre métriques présentées dans le Tableau 7.1.1. On peut faire les observations suivantes :

- La similarité basée sur la distance euclidienne échoue comme prévu à capturer une autre information que la redondance exacte (comparaison d'un segment avec lui-même).
- Les similarités cosinus et de Jacquard, donnent déjà des résultats plus intéressants, mais leur non-invariance par translation perturbe la lisibilité de la matrice. Les répétitions apparaissent bien sous la forme de traits sous-diagonaux.
- La distance normalisée de factorisation est celle qui fait apparaître le plus nettement les répétitions musicales. On peut noter également qu'en pratique, les scores ne dépassent que très rarement la valeur 1 (il faut en effet pour cela que les atomes choisis pour représenter x se révèlent encore meilleurs pour représenter y). Enfin, on notera que ND_R est la seule distance non-symétrique. Cela s'explique par la nature non-linéaire, à la fois de la décomposition MP, et de l'algorithme 4.

La métrique ND_R définit un critère de rejet simple de l'hypothèse de similarité. Il est en effet possible de filtrer la matrice de similarité en retirant tous les scores négatifs.

7.1.4 Limites

Les bonnes performances de la mesure proposée sur une tâche de détection de similarité ouvrent la voie à une utilisation de l'algorithme de factorisation pour, en pratique, effectuer un codage distribué d'un segment y connaissant un segment x similaire. En étudiant ce type d'approches, nous avons rencontré des limites, notamment en terme de performances de codage à plus haut débit, et surtout de passage à l'échelle.

Passage à l'échelle Il est assez simple de voir que la construction de matrice de similarité a une complexité quadratique. Comparer deux à deux J segments va nécessiter J^2 calculs de similarités. En limitant le nombre d'atomes à une centaine par segment et en développant des heuristiques pour stopper les calculs en cas de mauvaise corrélation dès les premières itérations, il est possible d'envisager un tel traitement sur des scènes sonores allant de quelques secondes à plusieurs minutes. On pourra à la limite, étudier les similarités dans un enregistrement d'une symphonie, mais pas dans une collection de plusieurs milliers de titres, ni dans un flux multimédia excédant quelques minutes.

Cette limite d'échelle est malheureusement difficilement contournable en considérant des segments plus grands. Les similarités ne pourront y être efficacement détectées qu'en augmentant la zone d'optimisation locale contrôlée par τ_{max} et du même coup la complexité et le débit R_{max}. En diminuant m, on augmente la distorsion, au risque de n'être plus capable de distinguer les similarités.

Pertinence du codage distribué de segments similaires à plus haut débit Dans la fin de l'article ICASSP 2011, nous considérons l'application de l'algorithme de factorisation à des débits plus élevés. Nous avons à cette occasion pu établir quelques conclusions :

- Dans les cas de redondances quasi-parfaites (c-$\grave{a}$-d quand $y \approx x$ avec un faible bruit et/ou décalage temporel), on peut diminuer très significativement le débit $R(y|x)$ par rapport à un codage indépendant. Dans ce cas de figure, les principes du codage distribué s'appliquent naturellement et l'algorithme de factorisation s'avère efficace.
- Dans les cas de répétitions (*p.ex.* lorsque y et x sont deux occurrences d'un même motif musical comme par exemple entre les mesures 1 et 4 de l'exemple ci-dessus), la distorsion introduite par

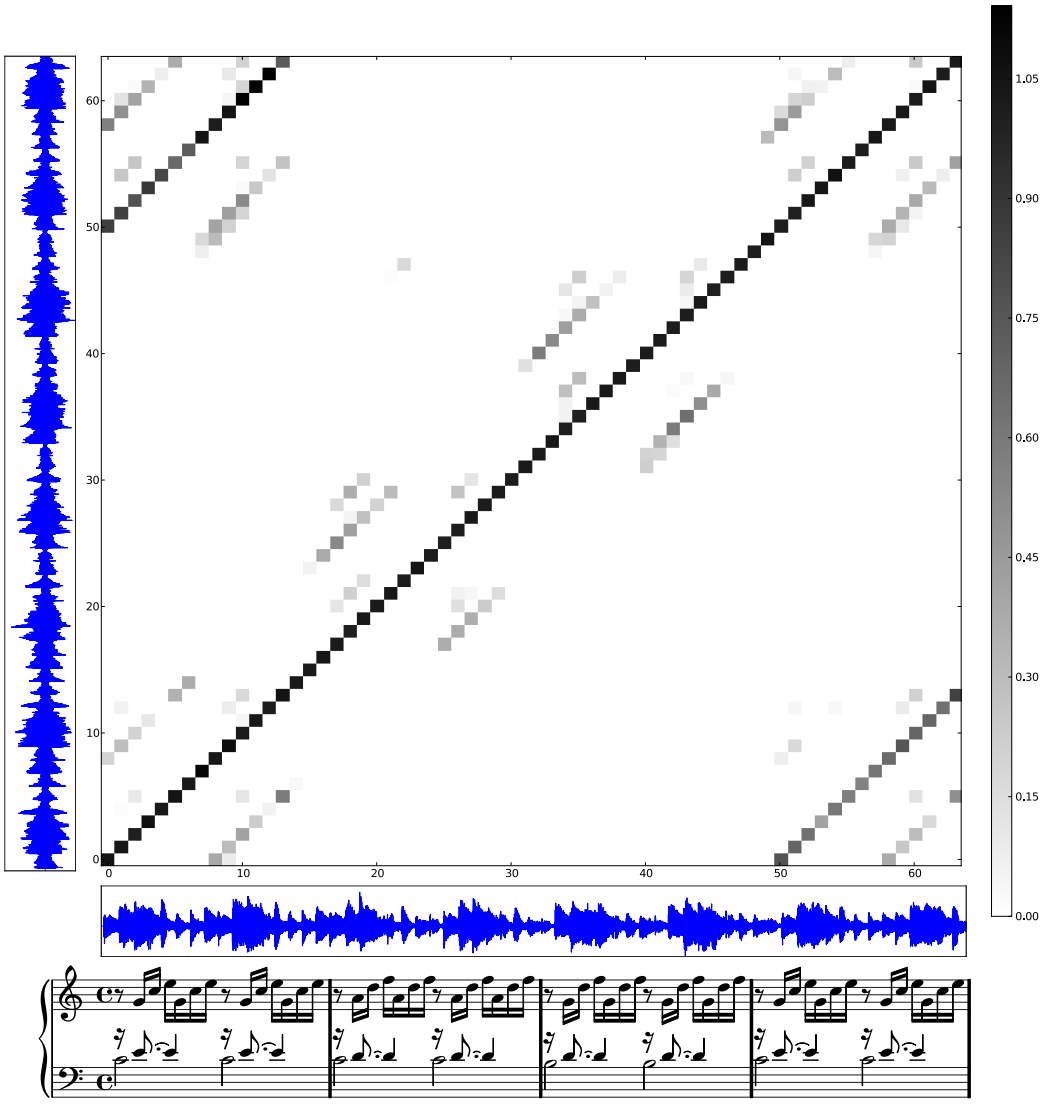

FIGURE 7.1.3: Matrice de similarité par mesure de factorisation ND_R, filtrée (valeur négatives mises à zéro) pour les 16 premières secondes du Prélude en ut (J.S Bach, Clavier bien tempéré BWV 846) interprétation de Glenn Gould. Scores obtenus avec des approximations MP adaptatif, stoppé après 100 itérations ou lorsqu'un SRR de 5 dB est atteint. La scène est découpée en 64 trames de 1 seconde, recouvrantes à 75%. La partition manuellement alignée avec l'audio est donnée à titre indicatif.

la contrainte de l'identité des supports entre $\hat{y}_m$et $\tilde{x}_m$ contre-balance le gain en codage, et ce pour des débits relativement faibles.

Malgré d'autres tentatives depuis, nous n'avons pour le moment pas pu montrer la pertinence du principe de codage distribué de motifs répétés, du moins en suivant cette approche. Nous verrons au chapitre suivant qu'il est possible d'envisager d'autres méthodes pour réaliser des approximations parcimonieuses jointes.

7.2 Détection de motifs récurrents par comparaison d'empreintes acoustiques

Lorsque l'on passe à l'échelle des bases d'archives numériques, les approches par calcul de similarités ne sont plus envisageables. Quelle que soit la puissance de calcul disponible ou la simplicité de la mesure utilisée, il n'est pas raisonnable de vouloir comparer deux à deux tous les objets d'une collection lorsque leur nombre se compte en milliers, voire en millions. Dans ces dimensions, le problème se rapporte à une recherche efficace des plus proches voisins. Dans ce travail nous envisageons une méthode de résolution de ce problème par comparaison d'empreintes acoustiques.

7.2.1 Empreinte acoustique

Une empreinte acoustique est une représentation du signal audio qui peut servir d'identifiant unique. L'identifiabilité cible est souvent celle de l'oreille humaine, c'est-à-dire que deux signaux identifiés comme identiques par un être humain doivent avoir la même signature. A ce titre, une empreinte se veut robuste à de petites déformations, en particulier les décalages temporels et petites modifications de hauteur.

L'empreinte acoustique fait son apparition dans les années 90 [Lou90] comme outil de détection du passage d'une oeuvre, une publicité ou un autre objet sonore dans un flux multimédia. Avec l'éclosion des moyens parallèles de diffusion de musique, le problème d'identification des *contre-façons* devient une préoccupation majeure de l'industrie.

Un système d'identification par empreinte, pour être intéressant, doit être capable de retrouver un signal parmi un très grand nombre. Ceci pose deux contraintes majeures :

- La base de données contenant les empreintes est potentiellement très grande.
- Les empreintes doivent permettre de différencier deux signaux distincts.

Idéalement, un système de comparaison d'empreintes doit présenter à la fois une robustesse aux petites déformations, une sensibilité (ou résolution) fine aux différences entre signaux, et tout cela à très grande échelle.

Les solutions de l'état de l'art se distinguent essentiellement par deux critères :

- Les descripteurs servant à construire l'empreinte.
- Le critère de comparaison.

La dimension d'une tâche typique de détection d'objets multimédia en général, et audio en particulier dépasse facilement le million d'unités. La base de données libres de MusicBrainz [1], propose des empreintes acoustiques (notamment basées sur le projet libre AcoustID [2]) pour plus de 12 millions

1. www.musicbrainz.org
2. www.acoustid.org

de références. Le même ordre de grandeur se retrouve chez The Echo Nest [EWJL10]. Des solutions industrielles sont désormais capables de reconnaître une scène sonore musicale dans des conditions relativement bruitées parmi une collection de cet ordre de grandeur, et en temps réel [Wan06].

Parallèlement, les techniques de marquage ont évoluées vers des systèmes robustes aux transformations usuelles (par compression). Citons par exemple l'approche de Varodayan et Girod [VG08] utilisant les principes du codage distribué pour garantir l'authenticité d'un signal. Le marquage peut également servir à transmettre une information additionnelle telle qu'un ensemble de métadonnées. Pinel *et al* [PGBP10] proposent ainsi un système qui *cache* l'information dans les sous-bandes perceptivement *libres,* au sens où, du fait d'effets de masquages psycho-acoustiques, une grande distorsion dans ces sous-bandes est quasi-inaudible pour l'auditeur.

7.2.2 Recherche du plus proche voisin

Soit une collection $\mathcal{X} = \{x_1, x_2, .., x_L\}$ de L objets (dans notre cas des scènes ou extraits de scènes sonores) et une requête q (un nouvel extrait), on cherche l'élément de $\mathcal{X}$ le plus proche de q. En considérant une mesure de *proximité* $d(x, y)$ entre deux objets x et y le problème s'écrit :

$$PPV(q) = \arg\min_{x \in \mathcal{X}} d(x, q) \tag{7.2.1}$$

Le but étant de retrouver cet élément en utilisant le moins de ressources possibles – donc en évitant d'effectuer L comparaisons entre la requête et les éléments de $\mathcal{X}$ – avec une grande probabilité.

Une difficulté supplémentaire apparait si on considère que les objets de $\mathcal{X}$ (et la requête) n'ont pas tous les mêmes dimensions. Ce cas de figure correspond bien par exemple à une collection de scènes sonores (par exemple musicales). Ensuite parce que la requête peut très bien n'être qu'une partie de l'un des objets (*p.ex.* un refrain d'une chanson, une phrase dans un discours, etc..).

Ces limitations mettent en lumière la nécessité de faire un choix d'architecture. Une approche consiste à définir des descripteurs globaux des scènes sonores, invariants par translation, robustes aux petites déformations, et calculables sur des sous-échantillons. Parmi ces méthodes, citons la contribution de Allamanche *et al* [AHH+01] où les auteurs utilisent des descripteurs bas-niveau de la norme MPEG-7.

A l'opposé, les systèmes de détection de motifs récurrents par comparaison d'empreintes acoustiques s'appuient sur une multitude de descripteurs locaux, pour les objets de référence comme pour la requête. Dans ce travail, nous nous concentrons sur cette deuxième approche. Par fusion des résultats, la recherche fournit alors, non seulement l'objet pour lequel on aura trouvé le plus de similitudes avec la requête, mais aussi l'information du décalage entre celui-ci et celle-la.

Le remède peut paraître paradoxal, puisqu'on va ainsi augmenter considérablement le nombre de recherches pour une seule requête. Le succès de cette méthode réside dans l'utilisation d'une structure de données spécifique : la table de hachage.

Table de hachage Pour mettre en oeuvre un tel système, il faut utiliser des méthodes d'indexation rapide. L'un des modes les plus efficaces pour le stockage et la recherche rapide d'éléments est de passer par une table de hachage. Soit une paire clef-valeur (c, v) dans un espace $\mathcal{C}\times\mathbb{V}$, une fonction de hachage $h : \mathcal{C} \rightarrow \mathbb{H}$ associe à toute clef un indice de localisation dans la table. La localisation de la paire (clef,valeur) dans la table étant une fonction déterministe de la clef ($h(c)$) il n'est pas nécessaire de parcourir toute la table pour connaître sa position. Ce type de stockage est courant dans la plupart

des systèmes de base de données actuels. La complexité d'une opération de recherche en base est (théoriquement) constante car la seule opération est le calcul de la fonction de hachage de la clef) c'est à dire indépendante de la taille de la base. De très nombreuses méthodes [HKO01, Wan03, JLY08, LCY+09] reposent sur cette structure.

Un inconvénient de ce type de structure vient du fait que la fonction de hachage est généralement choisie pseudo-aléatoire pour minimiser les risques de collisions (c'est à dire que deux clefs différentes se voient attribuer le même indice par la fonction de hachage). De ce fait, deux clefs proches vont se voir attribuer des indices très éloignés dans la table. A première vue, cette structure de données semble donc peu propice à la découverte de répétitions non exactes.

Pour pallier ce problème, des fonctions de hachages sensible (*Locality Sensitive Hashing* où LSH) à la localisation ont été proposées [IM98, Buh01]. L'idée est de garder le principe du hachage en introduisant une contrainte de sensibilité à la localisation, garantissant que pour deux clefs proches, la probabilité de collision soit forte, tandis que pour deux clef distantes cette probabilité sera faible. Différentes métriques peuvent être utilisées pour définir cette proximité. Parmi elles, nous retrouvons les similarités exposées plus haut : similarité en cosinus, de Jacquard, ou basée sur une distance binaire (*p.ex.* Hamming).

Cotton et Ellis [CE09] ont notamment utilisé ces structures avec des empreintes obtenues par Matching Pursuit. Récemment, Jegou *et al* [JFF11], se basant sur les travaux de Fuchs [Fuc11] sur les représentations réparties (qui se formalisent comme solution du problème (P_∞) voir 2.2.4 page 28) ont étudié une alternative au LSH.

Dans ce travail, nous nous concentrons sur la détection de redondances exactes de façon rapide. Le choix d'une structure de donnée telle que le LSH n'est donc pas justifié. Nous utilisons donc le système des tables de hachage simples, intégré à la bibliothèque libre Berkeley DataBase[3].

7.2.3 Comparaison d'empreintes

Avant de proposer une empreinte basée sur une décomposition MP dans un dictionnaire multi-échelles, il faut donc présenter l'architecture du système de comparaison d'empreintes que l'on a retenue dans ce travail et en premier lieu la résolution du problème de plus proche voisin par construction d'histogrammes de clefs.

Histogrammes de clefs Haistma [HKO01] est l'un des premiers à utiliser des descripteurs binaires comme clefs dans une table de hachage. Dans la littérature récente, les clefs sont calculées sur des représentations temps-fréquence, en appairant des pics répartis sur l'ensemble du plan temps-fréquence [Wan03, FRG11], comme illustré Figure 7.2.1.

A chaque trame i est associée un ensemble $\mathcal{K}$ de descripteurs-clefs c_k résumant une caractéristique locale du signal. Ainsi, dans le système *Shazam* [Wan03], chaque c_k décrit une paire de pics temps-fréquence de la TFCT. Soit (f_1, t_1) et (f_2, t_2) les localisations en fréquence et en temps de deux maxima locaux, une clef consiste donc en un triplet :

$$c_k = (f_1, f_2, \delta t = |t_1 - t_2|) \tag{7.2.2}$$

Il est important de noter que cette clef ne contient qu'une information temporelle relative. Il faut en effet considérer que la requête n'est pas alignée avec les références, une localisation temporelle absolue

3. http ://www.oracle.com/technetwork/products/berkeleydb/overview/index.html

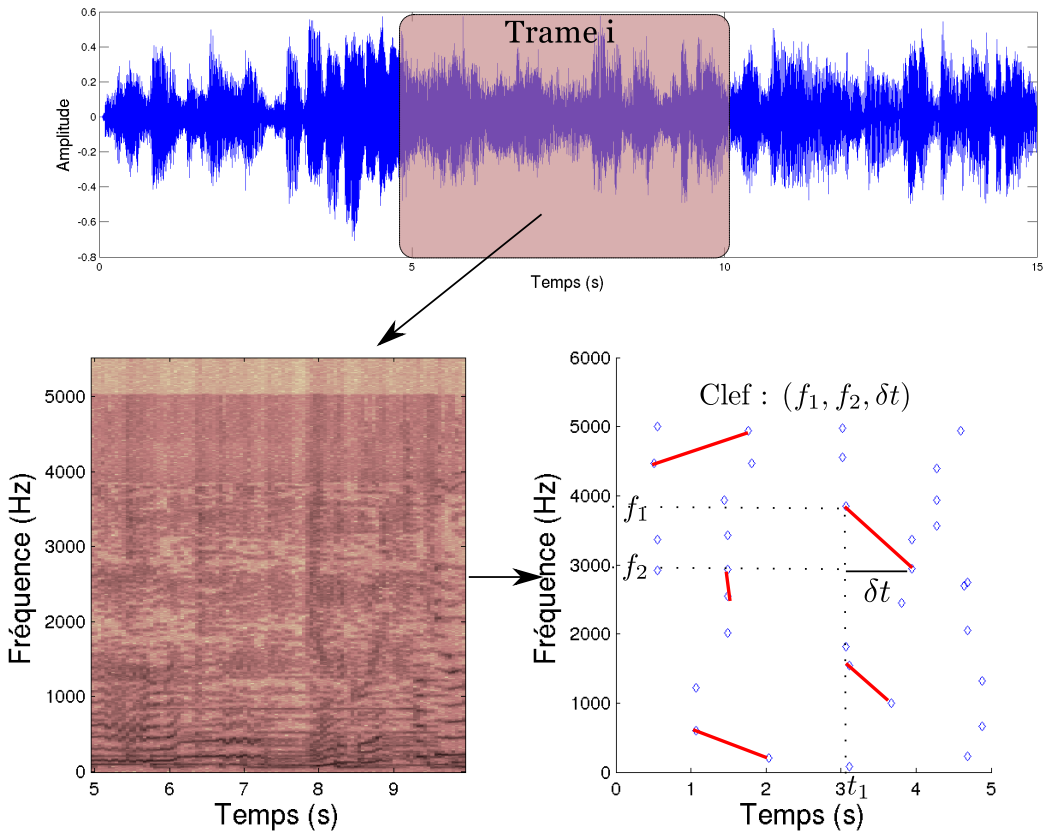

FIGURE 7.2.1: Calcul de descripteurs (clefs) locaux. Illustration de la méthode *Shazam* [Wan03], on calcule une représentation temps-fréquence (TFCT) du signal, sur laquelle on effectue une recherche de maxima locaux répartis dans le plan temps-fréquence. On appaire ensuite chaque pic avec ses voisins pour définir une clef : c_k à la position t_k. Chaque clef est composée des fréquences et de l'écart temporel entre les pics voisins : $c_k = (f_1, f_2, \delta_t = |t_1 - t_2|)$ et la valeur qui lui est associée dans la table de hachage est composée de l'indice de l'objet auquel elle appartient et de sa localisation temporelle.

des clefs n'est donc pas pertinente. FENET *et al* [FRG11] montrent qu'en utilisant une transformée à Q constant en lieu et place de la TFCT, cette clef est, dans une certaine mesure, robuste aux phénomènes de *pitch-shifting* (notamment grâce à un échantillonnage grossier en fréquence). Le positionnement exact de ce descripteur dans la scène sonore est donné par la valeur v_k associée à c_k dans la table de hachage. c_k contient l'information de l'indice l de l'objet de référence et du positionnement (position t_1 du premier pic) du descripteur dans cet objet. Soit en considérant un pas d'avancement fixe Δt entre chaque trame :

$$v_k = (l, t_k = t_1 + i\Delta t) \tag{7.2.3}$$

Lors de la construction de la base, il est naturel qu'une clef apparaisse plusieurs fois, dans différents objets ou pour une même scène sonore. On stocke donc dans la base l'ensemble des valeurs associées à une même clef.

Pour effectuer une recherche, la requête q est découpée en trames de même longueur et des descripteurs locaux sont calculés de la même façon. Chacune des clefs de l'empreinte de q fait l'objet d'une recherche dans la base, qui renvoie selon les cas :

—∘ $\emptyset$ si la clef n'est pas dans la table.

—∘ L'ensemble des valeurs $\{v_k^l\}$ correspondant aux occurrences de la clef dans la table. Connaissant la position t_k^q dans la référence, on en déduit un ensemble de décalages $\{|t_k^q - t_k^l|\}$.

On peut alors établir un décompte du nombre de clefs en commun entre la requête et chaque objet de la collection. Mais ce faisant, on traiterait chaque descripteur local indépendamment des autres. Il est donc plus intéressant de compter le nombre de clefs en commun par décalage temporel. Pour rendre cet estimateur robustes à de petites déformations (telles que le *pitch-shifting*, le *time-stretching*, ou encore de petites troncatures), ces décalages sont souvent quantifiés grossièrement. Des valeurs typiques sont, pour des trames de durée 5 secondes, une centaine de maxima locaux sélectionnés soit, après élagage, environ 400 paires de pics (clefs) par trame. Les décalages sont ensuite quantifiés à la seconde.

Si l'on envisage une collection de L objets sonores, dont on suppose pour simplifier qu'ils ont tous un nombre entier P de trames, le nombre de candidats potentiels au titre de plus proche voisin de la requête est de $L \times 5P$. Lors des campagnes d'évaluations internationales, les objets ont typiquement une durée de 60 secondes ($P = 12$) et L varie de plusieurs milliers ([FRG11]) à plusieurs centaines de milliers voire millions pour des applications industrielles [EWJL10, BMEWL11].

Il faut donc envisager plusieurs centaines (milliers) de recherches en base pour chaque requête. Ce qui souligne l'intérêt de l'utilisation de tables de hachage efficaces pour cette tâche. De plus, la très grande variété des combinaisons de clefs possibles permet une dispersion importante par hâchage. Le nombre de clefs rencontrées effectivement dans une requête est limité, et le rapport de ces deux grandeurs valide l'utilisation de ces structures.

Chaque requête implique la construction d'un histogramme H_q, contenant pour chaque objet x_i et chaque décalage τ le nombre de clefs en commun avec la requête. La recherche du plus proche voisin se réduit ainsi au problème trivial :

$$PPV(q), \tau_{max} = \arg \max_{i \in [0..L-1]} \mathrm{H}_q(i, \tau) \tag{7.2.4}$$

d'où l'on déduit $q \approx x_i * \delta(\tau_{max})$.

Clefs à partir de décomposition MP Il est possible de remplacer l'étape de sélection de maxima locaux dans une représentation temps-fréquence par une décomposition parcimonieuse superficielle du signal. En stoppant un MP à un faible nombre d'itération, on obtient une approximation à m-termes dans un dictionnaire multi-échelles comme illustré Figure 7.2.2.

Nous proposons d'utiliser directement les indices des atomes dans le dictionnaire comme clefs. Sur chaque segment on calcule donc m clefs c_k définies par l'indice de l'échelle et l'indice fréquentiel de l'atome. Là encore, la localisation temporelle absolue des atomes n'est pas utilisée pour la construction des clefs.

Il est tout à fait possible d'appairer les atomes de la même façon que les pics temps-fréquence. Néanmoins, nous avons voulu dans ce travail évaluer directement la capacité du support atomique à servir dans une tâche de comparaison d'empreintes sur des problèmes de dimensions importantes. L'autre intérêt est de garantir une complexité constante à l'étape de construction d'une empreinte, et surtout de limiter le nombre de clefs. Le léger surcoût lié à l'utilisation de MP est ainsi compensé par l'accélération de l'étape de recherche et la limitation des ressources nécessaires en stockage.

En revanche, il convient de modifier légèrement le critère de sélection des atomes. En effet celui-ci dans le cas de l'algorithme MP standard est purement énergétique. Or on aimerait capturer un maximum d'information pour construire des empreintes efficaces. On a vu dans la section 6.2.1 page 104

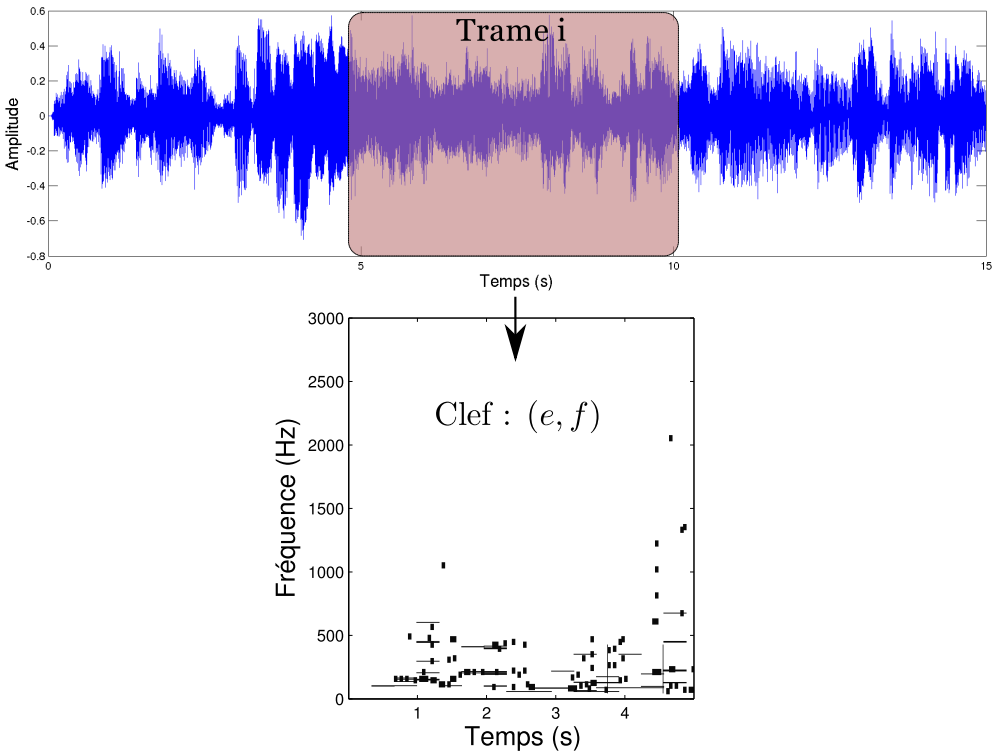

FIGURE 7.2.2: Calcul de descripteurs (clefs) locaux. Illustration de la construction d'une empreinte par décomposition superficielle dans un dictionnaire temps-fréquence (union de bases MDCT). Chaque atome définit une clef par la donnée de son échelle et de sa localisation fréquentielle : $c_k = (e_k, f_k)$ et la valeur qui lui est associée dans la table de hachage est composée de l'indice de la trame et du temps de son occurrence $v_k = (i, t_k)$.

que les probabilités empiriques de co-occurrence entre atomes du dictionnaires sont distribuées non-uniformément dans le plan temps-fréquence (Figures 6.2.2 et 6.2.3). En particulier, les atomes situés dans les basses fréquences sont plus souvent sélectionnés par MP. De même, les atomes d'une décomposition sont concentrés dans des voisinages temps-fréquence.

Il est possible d'apprendre des distributions $p(\boldsymbol{s})$. En particulier, connaissant les n premiers atomes d'une décomposition (*i.e.* le vecteur de supports $\boldsymbol{s}_n$), la probabilité de sélection de l'atome suivant n'est pas uniforme dans le dictionnaire (*i.e.* on connaît $P(\phi_i|\boldsymbol{s}_n)$ pour chaque atome ϕ_i). On peut donc imaginer une pénalisation supplémentaire dans le critère de sélection qui reflète cette information, le but étant ici de maximiser l'entropie $H(\boldsymbol{s}_n)$ du support. Le critère de sélection standard pour un résiduel $R^n x$ est :

$$\mathcal{C}(R^n x, \Phi) = \arg\max_{\phi \in \Phi} |\langle R^n x, \phi \rangle| \tag{7.2.5}$$

que l'on peut remplacer par :

$$\mathcal{C}_H(R^n x, \Phi, \lambda_H) = \arg\max_{\phi_i \in \Phi} (|\langle R^n x, \phi_i \rangle| - \lambda_H P(\phi_i|\boldsymbol{s}_n)) \tag{7.2.6}$$

Si ce critère présente un intérêt en théorie, plusieurs limitations apparaissent :

- Un tel critère ne permet pas de maximiser globalement l'entropie $H(\boldsymbol{s}_n)$, l'optimisation ne se faisant que localement.
- Le paramètre λ_H doit être réglé.

—∘ Apprendre les $P(\phi_i|\boldsymbol{s}_n)$ est une tâche très lourde.

Pour toutes ces raisons, nous proposons en pratique un critère plus simple :

$$\mathcal{C}_{\mathcal{M}}(R^n x, \Phi) = \arg\max_{\phi_i \in \Phi} (|\langle R^n x, \phi_i \rangle| \mathcal{M}(\phi_i|\Gamma^n)) \tag{7.2.7}$$

où $\mathcal{M}(\phi_i|\Gamma^n)$ est un masque temps-fréquence construit à partir du support de l'approximation courante Γ^n de telle sorte que la sélection d'un atome soit pénalisée voire interdite dans le voisinage d'un atome précédemment sélectionné. On peut pour cela utiliser la mesure

$$\mathcal{M}(\phi_i|\Gamma^n) = 1 - \max_{\gamma \in \Gamma^n} |\langle \phi_i, \phi_\gamma \rangle| \tag{7.2.8}$$

Ce terme de pénalisation est une façon très simple de favoriser la sélection d'atomes plus répartis dans le plan temps-fréquence. Il est assez simple de garder à jour ce masque et les produits $\langle \phi_i, \phi_\gamma \rangle$ peuvent être calculé au préalable et stockés dans une matrice de Gram.

Dans la suite, nous utilisons un MP dans une union de bases MDCT avec le critère (7.2.7) pour sélectionner les atomes qui servent de clefs. L'ensemble de ces clefs constitue l'empreinte acoustique du signal.

7.2.4 Détection d'objets dans des flux radiophoniques

L'état de l'art sur la reconnaissance de scènes sonores par comparaison d'empreintes avec une base de référence est déjà très fourni. Une contribution dans ce domaine nécessiterait une logistique de validation lourde (base de données de test très importante). Dans ce travail, nous n'avons pas pour objectifs de proposer un système compétitif de comparaison d'empreintes, mais d'évaluer la pertinence de l'utilisation des représentations parcimonieuses dans ce cadre. Pour cela, le problème de la segmentation de flux radiophoniques est plus adapté.

On s'intéresse en particulier au problème de la détection de motifs récurrents dont on ne connaît ni la position ni la longueur. Ce problème est décrit et formalisé par Herley [Her06]. Nous avons proposé, dans un travail en commun avec S. Fenet, une architecture abordant ce problème. Celle-ci est schématisée Figure 7.2.3.

A l'initialisation, la base d'empreintes est vide. Pour chaque trame de flux, une comparaison d'empreintes par recherche des clefs dans la base permet de trouver parmi les trames précédentes les k plus proches voisins. Une fusion de ces décisions locales est ensuite nécessaire pour déterminer de façon cohérente si une trame et celles qui l'entourent sont des répétitions de trames diffusées plus tôt dans le flux. La Figure 7.2.4 présente le schéma-bloc de l'architecture développée dans l'article [FMG+12]. Les principales caractéristiques de ce système sont :

—∘ Une recherche des k plus proches voisins par comparaison d'empreintes. Nous utilisons pour cela une table de hachage et des histogrammes comme décrit plus haut. Dans cette configuration, il n'y a aucun objet connu a priori, l'histogramme est donc construit uniquement pour les décalages existant, c'est à dire l'écart entre la trame courante et les trames précédentes qui possédaient les même clefs.

—∘ Le mécanisme de fusion des décisions locales. Après l'observation de P trames, si on trouve de façon persistante un décalage parmi les k plus proches voisins, alors on peut considérer qu'il s'agit d'une redondance.

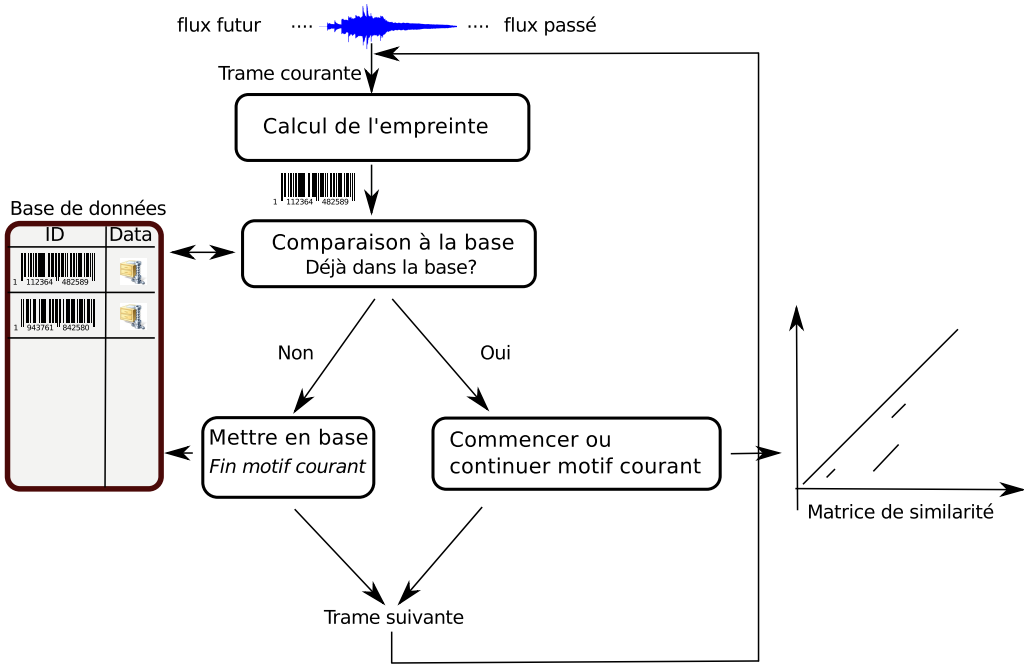

FIGURE 7.2.3: Architecture de détection automatique de motifs récurrents dans un flux audio.

—∘ Le stockage dans la base de données de l'empreinte de la trame courante est conditionnée à la décision de redondance. Une trame originale sera stockée tandis qu'une trame redondante ne le sera pas.

Ce système permet, pour chaque trame analysée, de déterminer s'il s'agit de contenu déjà diffusé ou original. Dans une étape ultérieure, une succession de trames considérées comme redondantes va définir un objet. Cette étape de regroupement permet, étant donné un flux audio, de construire une base d'objets redondants avec leur positionnement dans le flux. Lors de cette étape, on peut fixer un seuil minimal de nombre de trames consécutives marquées comme redondantes pour qu'un objet soit constitué.

Il y a donc deux façons d'évaluer ses performances, soit en utilisant des métriques trame par trame, soit en évaluant la qualité de la base d'objets construite.

7.3 Évaluations

Pour évaluer les performances du système, il faut disposer d'une base de test sur laquelle on a une connaissance a priori des redondances. Il y a deux façons de construire de telles bases :

—∘ Soit en construisant de façon synthétique un flux à partir d'une collection d'objets distincts.

—∘ Soit en annotant à la main un flux réel.

Au sein du projet QUAERO, une collaboration entre l'IRCAM, Yacast, le l'IRIT et Telecom ParisTech a permis de mettre au point une base annotée de dimension importante et un protocole d'évaluation. Les détails sont présentés dans l'article [RFB+12].

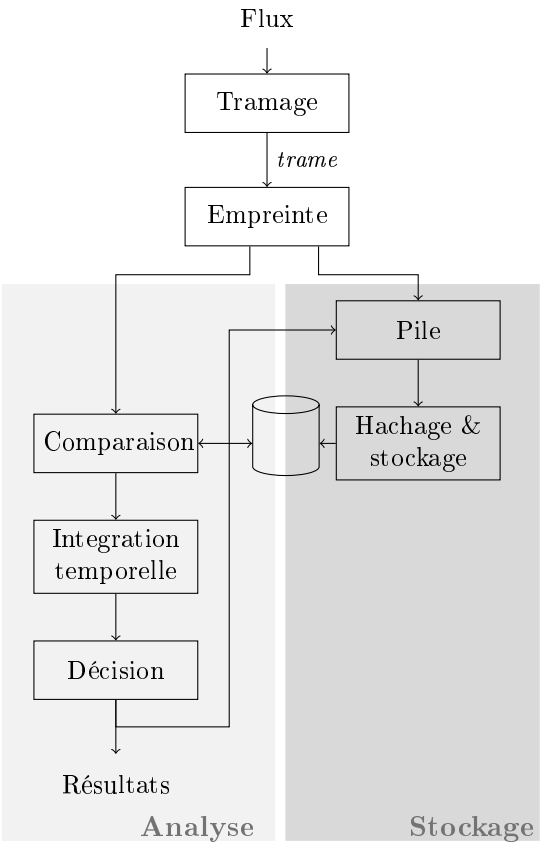

FIGURE 7.2.4: Architecture d'un système de segmentation de flux en objets redondants. D'après [FMG+12]

7.3.1 Métriques

Soit une collection d'éléments $\{x_i\}$ pour lesquels on dispose d'annotations, c'est-à-dire pour chaque x_i, un ensemble $\mathcal{V}(x_i) = \{j_v\}$ d'indices des éléments avec lesquels x_i est redondant. Un système fournit pour x_i une estimation $\mathcal{S}(x_i) = \{j_S\}$ de cet ensemble d'indices et les performances sont mesurées en terme de :

- Précision :

$$P_S(i) = \frac{|\{j_v\} \bigcap \{j_S\}|}{|\{j_S\}|} \tag{7.3.1}$$

- Rappel :

$$R_S(i) = \frac{|\{j_v\} \bigcap \{j_S\}|}{|\{j_v\}|} \tag{7.3.2}$$

- F-mesure :

$$F_1(i) = \frac{2(P_S(i)R_S(i))}{P_S(i) + R_S(i)} \tag{7.3.3}$$

La précision évalue la capacité à ne pas commettre de fausses détections (souvent appelées fausses alarmes), le rappel a retrouver toutes les répétitions. La F-mesure résume ces deux caractéristiques.

Dans cette application, on peut calculer ces métriques sur les objets de la base construite par le système ou directement sur chaque trame du flux. Dans ce dernier cas il faut être sûr de disposer d'annotations suffisamment précises (à l'échelle d'une trame soit par exemple à 5 secondes près) pour

que les scores obtenus aient un sens. En pratique, une telle précision sur les annotations pour des flux réels est très difficile à obtenir. En revanche, sur des flux construits de façon synthétique, la position des redondances est connue avec une précision à l'échantillon près et les performances trame par trame sont pertinentes.

Exemple de segmentation Avant de présenter les résultats obtenus sur des bases réelles et synthétiques, la Figure 7.3.1 donne une idée du comportement du système proposé. Un certain nombre de remarques peuvent déjà être faites :

- Quelques fausses alarmes apparaissent, surtout au démarrage. Ceci s'explique par le fait que la base étant initialement vide, la probabilité qu'un décalage ressorte parmi les k candidats pour P trames successives, même s'il ne s'agit pas d'une redondance, est significative. Pour éviter cela, on interdit toute détection de redondance durant une certaine période, de façon à initialiser correctement la base de données (en anglais il s'agit d'un *cold start problem*).
- Certaines redondances, qui sont pourtant exactes, apparaissent comme des erreurs du fait des imprécisions et lacunes des annotations. Ce problème est malheureusement très fréquent pour des tâches de cette dimension et ne peut être réglé qu'en modifiant manuellement les annotations (on parle d'*adjudication*).
- Certaines erreurs n'en sont pas vraiment. En particulier il arrive que le premier refrain de la deuxième diffusion d'une chanson soit marqué comme une répétition du second refrain de la première diffusion. En réalité, les deux possibilités sont généralement présentes parmi les plus proches voisins retenus et la décision peut se jouer à quelques clefs seulement. On notera que les données utilisées pour cette expérience proviennent d'une radio commerciale en 2011. Les chansons sont de type musique populaire internationale, pour lequel il est courant de trouver des refrains très redondants.

7.3.2 Comparaisons avec une empreinte standard

Nous évaluons le système avec deux types d'empreinte : la première, proposée dans une étude récente [FRG11], est calculée comme dans [FRG11] avec des paires de maxima locaux dans une transformée à Q constant. On la note CQT-Pics. La seconde est l'empreinte obtenue par décomposition MP stoppée à 150 itérations (avec masquage des voisinages temps-fréquence), dans une union de bases MDCT. On la note MP-150.

Comparaison sur une base synthétique Une première expérience consiste à étudier la pertinence des décisions trame par trame. Pour cela on construit un flux synthétique de la façon suivante. 140 extraits de scènes sonores (distinctes deux à deux) sont choisis dans une base de données de musique populaire. Chaque extrait a une durée de 30 secondes. Parmi ces 140 objets, on en duplique 100 pour constituer un ensemble de 240 objets de 30 secondes que l'on concatène pour constituer un flux synthétique d'une durée de 2 heures. La position de chaque objet dans le flux est tirée de façon aléatoire mais connue, ce qui va nous servir de vérité terrain.

Pour évaluer la capacité du système, le flux est découpé en trame de 5 secondes et analysé par le système en utilisant les différentes empreintes. Le Tableau 7.3.1 présente les résultats obtenus pour ce cas de figure simplifié.

Ces résultats amènent les remarques suivantes :

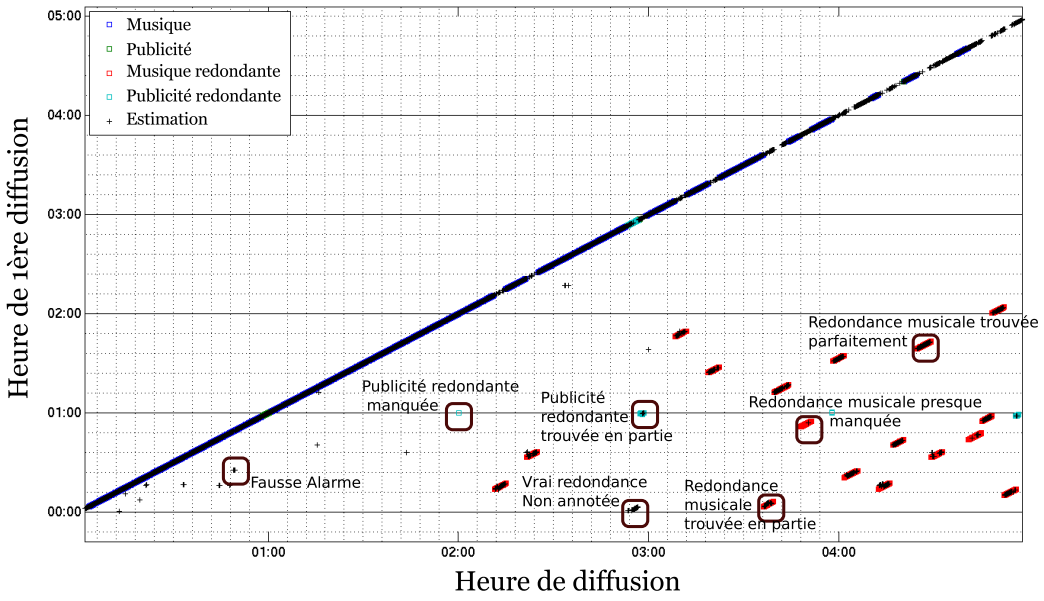

FIGURE 7.3.1: Matrice de similarité obtenue par détection de motifs récurrents dans un flux radiophonique (5 heures). Les résultats sont figurés ici trame par trame (*c-à-d.* avant regroupement des trames consécutives en objets). En carrés de couleur la vérité terrain provenant d'annotations manuelles. En croix noires les décisions locales. Si une trame est jugée originale, la croix est sur la diagonale. Sinon, elle est positionnée en ordonnée sur l'instant de première diffusion estimé.

	CQT-Pics	MP-150
Précision (%)	95.1	94.5
Rappel (%)	97.8	91.5
F-mesure (%)	96.5	93.0
Temps CPU - Calcul Empreinte (s)	0.12	0.33
Temps CPU - Recherche Base (s)	0.08	0.07
Mémoire (Mo)	9.3	2.4

TABLE 7.3.1: Comparaison des ressources et performances trame à trame pour une tâche de détection de motifs récurrents dans un flux synthétique de 2 heures. Temps moyens par traitement d'une trame (5 secondes) obtenus sur une machine Double Coeur à 3.16GHz, 4Go de mémoire RAM.

Empreinte	Nb Rép. Trouvées / Total (Rappel)	Fausses alarmes
CQT-Pics	191 / 191 (=100%)	0
MP-150	178 / 191 (=93.2%)	1

TABLE 7.3.2: Scores sur une tâche de détection d'objets

—∘ Les scores obtenus par le système, quelle que soit l'empreinte utilisée sont encourageants : une F-mesure de plus de 93% sur un flux de 1440 trames.

—∘ L'empreinte calculée par MP est légèrement moins robuste que l'empreinte standard. Elle est cependant significativement plus petite (environ 4 fois). On remarque d'ailleurs que si le temps de calcul de l'empreinte MP est supérieur, le temps de recherche en base est plus petit en utilisant ces empreintes. Cette différence peut paraître négligeable, mais plus les flux traités sont longs, plus le temps de calcul des empreintes (qui est constant) devient négligeable devant celui nécessaire à la recherche de clefs et à la construction des histogrammes.

Comparaison sur des bases réelles La même configuration est évaluée sur un corpus de 24H de radio manuellement annoté dans QUAERO. Pour ces données réelles, l'annotation est trop problématique (en raison des coupes et autres déformations subies par les objets lors de leur diffusion) pour envisager une évaluation trame par trame. Nous nous concentrons donc sur la capacité du système à segmenter le flux, c'est à dire construire une base d'objets pertinents (e.g. chansons, publicités) effectivement redondants, et surtout à situer ces redondances aux bons endroits du flux.

Nous nous intéressons notamment à la détection de chansons, c'est-à-dire en pratique la détection d'objets redondants dont la taille excède 90 secondes. Cette limite correspond expérimentalement à la durée maximale observée d'une publicité sur la base considérée. Le Tableau 7.3.2 présente les performances obtenues.

Le ratio du nombre de répétitions trouvées sur le nombre de répétition totales donnée par les annotations est une mesure de Rappel et le nombre de fausse alarmes donne dans ce cas de figure une idée de la précision. Sur cette tâche, le système utilisant l'empreinte CQT ne commet aucune erreur. En utilisant l'empreinte MP, plus simple, les performances sont légèrement inférieures. Cela s'explique par la taille des empreintes (nombre de clefs plus faible) et par le fait qu'entre plusieurs diffusions d'une même chanson, certaines distorsions de type *pitch-shifting* peuvent apparaître.

Pour rendre notre empreinte robuste à ces petites déformations, une indexation plus poussée (p. ex. en utilisant des paires d'atomes pour définir les clefs) est sans doute nécessaire.

7.3.3 Campagne QUAERO

Le système décrit ci-dessus à été présenté dans une campagne d'évaluation européenne, menée au sein du projet QUAERO. Le corpus d'évaluation comportait 144 heures de flux radiophonique, réparties en deux corpus :

—∘ Un corpus de 72 heures, comprenant 3 jours successifs pour une même radio (commerciale, forte concentration de chansons redondantes et de publicités)

—∘ Un corpus de 72 heures constitué des enregistrements de 3 radios différentes sur une même journée.

Corpus	Empreinte	Nb Rép. Trouvées / Total (Rappel)	Fausses alarmes
3 Jours	CQT-Pics	565 / 565 (=100%)	0
	MP-150	537 / 565 (= 95%)	7
3 Radios	CQT-Pics	626 / 644 (=97%)	8
	MP-150	478 / 644 (= 74%)	5

TABLE 7.3.3: Scores sur une tâche de détection de motifs récurrents pour deux corpus. Évaluation réalisée au sein du projet QUAERO par l'IRIT.

Le premier corpus constitue, sur une échelle plus grande, le même type de données que le cas présenté plus haut. En revanche le second corpus constitue un ensemble de données plus complexe. Le mixage et les distorsions d'un titre varient en effet de façon beaucoup plus importante entre deux diffusions sur deux radios distinctes qu'entre deux diffusions sur une même radio. En pratique, on observe des différences importantes du taux de compression et des modifications de hauteurs allant jusqu'au quart de ton.

Une difficulté supplémentaire vient du fait que deux radios peuvent diffuser deux versions différentes d'une même chanson. Certaines versions sont amputées de parties entières, telles qu'une répétition de refrain, un couplet entier, l'introduction ou la fin de la scène originale. Ces mutilations ont pour but de faire *tenir* les objets sonores dans les cases de diffusion dont la durée est fixe (intervalle entre deux tranches de publicités).

Sur ces évaluations, dont les résultats sont présentés dans le Tableau 7.3.3, nous pouvons faire les remarques suivantes :

- Les performances de l'architecture avec l'empreinte basée sur la représentation parcimonieuses de 150 atomes MDCT pour le corpus 3 Jours sont assez proches de l'optimal. Pour le corpus 3 Radios, ces performances sont nettement moins bonnes, ce qui confirme la difficulté de la tâche et le défaut de robustesse aux petites déformations de la diffusion radiophonique de l'empreinte présentée.
- Clairement, la robustesse introduit par l'appairage des pics améliore les résultats. Mais cela a un coût. Bien que les évaluations conduites ne l'aient pas pris en compte, la complexité (notamment le temps de calcul, mais aussi les ressources mémoires) est plus grande avec ce type d'empreintes.

De même, un certain nombre de paramètres du système n'ont pas été optimisé sur la base de données de test. Sur des cas de figure pratique d'archivage, il semble possible d'envisager l'utilisation de ces empreintes, dans ce type de système, pour effectuer rapidement une tâche de détection de motifs récurrents sur un grand volumes de données.

Conclusion partielle

La détection de motifs récurrents peut s'effectuer à des échelles très diverses. Sur des signaux relativement courts (jusqu'à quelques minutes) il est possible de mettre au jour la structure interne (par exemple d'une chanson). En utilisant les représentations parcimonieuses, nous avons ainsi défini une distance mesurant la capacité d'une représentation de x à représenter y. Cette mesure se substitue à une information mutuelle. L'algorithme de factorisation réalise ainsi une mesure de similarité souple et effectue dans le même temps un commencement de codage distribué.

Sur des échelles plus grandes, ces mesures de similarités sont inadaptées et nous avons recours à des méthodes de comparaisons efficaces d'empreintes acoustiques. Il est possible d'utiliser les mêmes

représentations, dictionnaires et algorithmes pour construire des empreintes de dimensions réduites. Au sein d'une architecture spécifique, ces empreintes permettent de détecter des redondances sur des volumes de plusieurs jours de flux radiophonique.

Il est intéressant de noter que tout au long de ce chapitre, nous utilisons le même dictionnaire et des variantes très proches de MP. Nous avons ainsi une illustration des possibilités de traitements à différentes échelles de données qu'offrent ce type de représentations.

Chapitre 8

Séparation de sources répétitives

Nous avons vu les possibilités de détection de récurrence à différentes échelles, ainsi qu'une application à grande échelle de segmentation de flux radiophonique. Dans ce chapitre, nous présentons une application qui paraît *a priori* éloignée des problématiques de l'archivage : la séparation de sources monophoniques. Nous allons voir que dans certains cas où une information sur la structure des sources est disponible, il est possible d'effectuer une tâche de séparation de sources à l'aide d'algorithmes de compression de type Matching Pursuit.

Le lien s'établit à travers le formalisme d'un problème d'approximations jointes de plusieurs signaux. Nous commencerons donc, section §8.1, par présenter ce formalisme et étudierons à quels cas spécifiques de séparation de sources il correspond. Dans un second temps, section §8.2, nous présenterons un algorithme inspiré des Matching Pursuit Simultanés, et les légères adaptations nécessaires pour effectuer une séparation de sources en même temps que la décomposition du mélange.

Enfin, nous évaluons section §8.3 les performances de l'algorithme, tant en termes de séparation que d'approximations simultanées. Ce chapitre à fait l'objet d'un article à la conférence EUSIPCO 2012 [MRD12].

8.1 Formalisation

La séparation de sources monophoniques est le problème qui consiste, étant donné un (ou plusieurs) signal de mélange monocanal $x \in \mathbb{R}^N$, à retrouver les signaux sources élémentaires qui le composent. Nous nous concentrons sur un cas de figure particulier où l'on dispose de plusieurs de ces mélanges, partageant tous au moins une source commune. Nous entendons ici par *source commune* un signal qui peut lui-même être un mélange de sources sonores. Cette configuration se retrouve dans un certain nombre de problèmes pratiques, lorsque les signaux considérés ont une structure particulière, répétitive ou redondante.

8.1.1 Cas pratiques de séparation de sources répétitives

On peut discerner au moins deux cas pratiques de séparation de sources dans lesquels la redondance des sources joue un rôle central :

—∘ La Séparation de Composante Commune (SCC) : Dans cette configuration, on dispose d'un ensemble de signaux de mélanges et l'on suppose qu'il existe une (et une seule) source commune à chacun de ces mélanges. Par exemple, si l'on dispose des différentes versions (dialogues en

différents langages) de la bande son d'un film, alors on peut supposer que chacun de ces signaux contient comme composante commune la musique et les bruitages en plus des voix des acteurs, spécifiques à chaque version [LL10, BL11]. Plus généralement, si l'on dispose d'enregistrements multi-capteurs bruités d'une scène sonore, alors le débruitage de cette composante commune est typiquement un problème de Séparation de Composante commune [TG05, Gri02].

—∘ La Séparation de motifs récurrents (SMR) : Récemment, Rafii *et al* [RP11a] ont montré qu'on pouvait tirer profit de la nature relativement répétitive de la musique pour séparer du fond musical une composante non (ou plutôt moins) répétitive, typiquement la voix. Une extension de ce travail est proposé dans [LRB+12].

Ces deux exemples sont liés par le modèle commun qu'ils sous-entendent : soit un ensemble de I mélanges $\{X_i\}_{i\in[0..I-1]}$ tel que chaque X_i se comprend comme la somme d'une composante propre P_i et d'une composante partagée X_c (potentiellement déformée de façon propre X_{c^i}). Ce qui distingue les deux problèmes ci-dessus est principalement la modélisation des composantes propres. Pour la SCC, ces composantes sont traitées comme du bruit et seul le recouvrement de X_c à partir de ses multiples exemplaires est recherché. Pour la SMR, X_c capture le fond musical redondant dans les différentes occurrences des motifs. Les sources propres capturent quant à elles la partie variable de chaque occurrence et, contrairement au cas de la SCC, on désire souvent récupérer chacune de ces composantes au même titre que X_c.

La redondance est ici ce qui justifie le modèle des mélanges. La cause de cette redondance (multiplicité des capteurs/versions ou structure répétitive de la musique) est secondaire et les algorithmes traitant l'un ou l'autre de ces problèmes sont donc de même nature. Une formulation plus générale est donnée dans la section suivante.

8.1.2 Approximations Parcimonieuses Simultanées

On peut formuler le problème de séparation de sources comme un problème d'approximations simultanées [TGS06]. Il s'agit en effet, d'estimer simultanément les différentes composantes P_i et X_c. Soit $\boldsymbol{X} = \mathbb{R}^{N\times I}$ une matrice de I mélanges $X_i \in \mathbb{R}^N$. Soit $\boldsymbol{\Phi} \in \mathbb{R}^{N\times M}$ un dictionnaire de M atomes, on cherche une approximation de $\boldsymbol{X}$ dans $\boldsymbol{\Phi}$ sous la forme :

$$\tilde{\boldsymbol{X}} \approx \boldsymbol{\Phi} \cdot \boldsymbol{C_X} \tag{8.1.1}$$

ou $\boldsymbol{C_X} \in \mathbb{R}^{M\times I}$ est une matrice de coefficients que l'on souhaite parcimonieuse. Le problème d'approximations parcimonieuses simultanées s'écrit :

$$(\boldsymbol{SP_0^\epsilon}) : \min_{\boldsymbol{C_X}} \|\boldsymbol{C_X}\|_0 \text{ soumis à } \|\boldsymbol{X} - \boldsymbol{\Phi}\cdot\boldsymbol{C_X}\|_F^2 \leq \epsilon \tag{8.1.2}$$

où $\|\cdot\|_F$ est la norme de Frobenius et $\|\boldsymbol{C}\|_0$ compte le nombre d'entrées non nulles de la matrice $\boldsymbol{C}$. Le problème de recouvrement parcimonieux correspondant est une variante du *Compressed Sensing* connue sous le nom de *Multiple Measurement Vector*. L'algorithme *Simultaneous Orthogonal Matching Pursuit* (SOMP) [LT06, TG05] est une adaptation d'OMP au problème $(\boldsymbol{SP_0^\epsilon})$. On peut de la même façon qu'avec le cas de figure $I = 1$ proposer une relaxation de ce problème $(\boldsymbol{SP_0})$ en $(\boldsymbol{SP_{p,q}})$ à l'aide de norme mixte $\ell_{p,q}$:

$$\|\boldsymbol{C}\|_{p,q} = \left(\sum_{i,j} |c_{i,j}|^p\right)^{\frac{q}{p}} \tag{8.1.3}$$

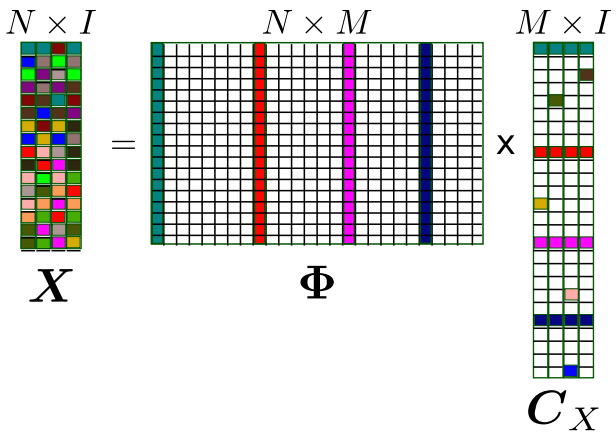

FIGURE 8.1.1: Illustration matricielle du problème d'approximations parcimonieuses simultanées

L'algorithme d'optimisation convexe correspondant est appelé Group-LASSO [YL06, RF08]. La version pénalisée de ce problème s'écrit :

$$(\boldsymbol{SL}^{\epsilon}_{\boldsymbol{p,q}}) : \min_{\boldsymbol{C_X}} \|\boldsymbol{X} - \boldsymbol{\Phi} \cdot \boldsymbol{C_X}\|^2_F + \lambda \|\boldsymbol{C_X}\|_{p,q} \tag{8.1.4}$$

ou p et q peuvent être choisi en fonction du profil de parcimonie recherché et λ est un paramètre de contrôle de la contrainte de parcimonie. KOWALSKI *et al* [KVG10] ont montré que les normes mixtes favorisent la sélection de matrices dont la parcimonie est structurée en lignes et/ou colonnes. Ce type de structure est particulièrement intéressant dans notre cas de figure.

Interprétation des structures Si chaque colonne de $\boldsymbol{X}$ est une occurrence d'un motif, on peut espérer que la composante partagée par toutes les occurrences (*p.ex.* le fond musical) se retrouvera dans la décomposition sous la forme de lignes d'éléments non nuls dans la matrice $\boldsymbol{C_X}$. En effet, le coefficient $c_{m,i}$ de $\boldsymbol{C_X}$ code l'utilisation de l'atome ϕ_m dans la représentation du mélange X_i. Si cet atome appartient à la représentation de la composante partagée, alors toute la ligne m de $\boldsymbol{C_X}$ doit être non nulle. A l'inverse, une ligne pour laquelle seuls quelques éléments sont non nuls dénote un atome appartenant aux composantes propres d'un sous-ensemble de mélange. L'utilisation d'une norme mixte pénalise très fortement ce deuxième cas et favorise donc la sélection d'atomes efficace sur l'ensembles des signaux.

Distinction selon les profils de parcimonie L'observation ci-dessus nous amène à proposer une méthode simple de séparation de sources basée sur la résolution de $(\boldsymbol{SP}^{\epsilon}_{\boldsymbol{0}})$ et la décomposition de la matrice $\boldsymbol{C_X}$ en fonction des profils de parcimonie observés. L'idée est proche de l'esprit de la *Robust PCA* [CLM09] qui a d'ailleurs été reprise pour la séparation de sources par HUANG *et al* [HCSHJ12]. Ici nous ne séparerons pas $\boldsymbol{C_X}$ en une composante de rang faible et une composante parcimonieuse car $\boldsymbol{C_X}$ est déjà parcimonieuse. Nous n'utilisons pas de normes mixtes non plus car les deux types de parcimonie nous intérèssent et la décomposition se fait selon ce type de parcimonie :

$$\boldsymbol{C_X} = \boldsymbol{B_X} + \boldsymbol{P_X} \tag{8.1.5}$$

où $\boldsymbol{B_X}$ est une matrice parcimonieuse *structurée* (*c-à-d.* les éléments d'une même ligne sont tous nuls ou tous non nuls) et d'une matrice $\boldsymbol{P_X}$ parcimonieuse *explicitement non structurée* (*c-à-d.* sur une

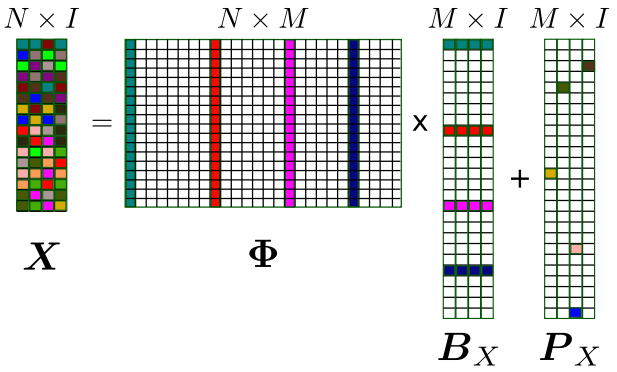

FIGURE 8.1.2: Séparation des sources selon le type de parcimonie.

même ligne, tous les éléments ne peuvent être non nuls). L'intérêt de cette décomposition est évident dans le contexte de cette étude, où la matrice $\boldsymbol{B_X}$ capture la composante commune (*p.ex.* le fond musical) tandis que la matrice $\boldsymbol{P_X}$ regroupe les composantes propres. Le problème ($\boldsymbol{SP_0}$) étant un problème de synthèse, on obtient directement la représentation des sources séparées.

On peut tout à fait tenter de construire $\boldsymbol{B_X}$ et $\boldsymbol{P_X}$ à partir de $\boldsymbol{C_X}$, alternativement, ces deux profils de parcimonie peuvent être directement inclus dans la formulation du problème :

$$(\boldsymbol{SL_{p,q,p',q'}}) : \min_{\boldsymbol{B_X},\boldsymbol{P_X}} \tfrac{1}{2}\|\boldsymbol{X} - \boldsymbol{\Phi}\cdot(\boldsymbol{B_X}+\boldsymbol{P_X})\|_F^2 + \lambda_B\|\boldsymbol{B_X}\|_{p,q} + \lambda_P\|\boldsymbol{P_X}\|_{p',q'} \tag{8.1.6}$$

Limites pratiques Il est possible de proposer des méthodes convexes pour résoudre ce type de problèmes (en supposant que les normes mixtes utilisées gardent le problème convexe). Malheureusement, pour le type de données que l'on veut traiter, ces méthodes ont une complexité importante. De plus, fixer a priori les paramètres λ_B,λ_P, p et q n'est pas évident.

Plus généralement, cette formulation matricielle est en quelque sorte trop rigide pour les cas d'utilisation envisagés, en particulier dans le cas de la SMR. Le fond musical n'est sans doute pas exactement le même d'une occurrence à l'autre, surtout s'il s'agit d'une performance instrumentale (par opposition à une répétition d'un motif enregistré, ou *sample).* Vouloir structurer la parcimonie sur les lignes, revient à s'interdire de prendre en compte des variations (autres que d'amplitude) entre les occurrences de la composante commune. Si ce modèle est adapté à certaines situations (notamment des instances de SCC), il lui manque la souplesse nécessaire à son application aux scènes sonores musicales considérées.

Dans ce travail de thèse, nous utiliserons donc des algorithmes gloutons proches de ceux déjà rencontrés dans cette thèse. Nous proposons notamment un algorithme original de Matching Pursuit Joint, proche dans l'esprit de celui proposé par TROPP [TG05]. Nous allons voir que deux adaptations peuvent être proposées pour les problèmes de SCC et SMR, un changement du critère de sélection des atomes, et la prise en compte des variations entre occurrences à l'aide d'un paramètre temporel.

8.2 Matching Pursuit Joints

8.2.1 Algorithme

Nous avons choisi d'utiliser des variantes du MP basés sur la règle de mise à jour standard (et non orthogonale comme proposée par TROPP [TG05]). Rappelons que nous nous intéressons directement

au problème ($\boldsymbol{SP_0^\epsilon}$) (8.1.2). Les solutions trouvées par ce type d'algorithme sont sous-optimales mais la complexité réduite permet d'envisager simplement le traitement de données de grande dimension. Or nous souhaitons appliquer ces méthodes à des scènes sonores de plusieurs secondes voire minutes.

Une méthode possible serait de décomposer chacun des mélanges X_i séparément des autres, construire la matrice $\boldsymbol{C_X}$ par concaténation et réaliser la séparation dans une phase de post-traitement. Cependant, nous avons trouvé que des performances supérieures peuvent être obtenues lorsque l'on incorpore la connaissance *a priori* de la redondance dans le processus de décomposition. Cette intégration prend la forme de deux changements dans l'algorithme standard :

1. Le critère de sélection des atomes doit prendre en compte la nouvelle dimension.
2. Un mécanisme d'attribution de l'atome sélectionné soit à $\boldsymbol{B_X}$ soit à $\boldsymbol{P_X}$ doit être mis en place.

En revanche, la règle de mise à jour standard peut être conservée. L'algorithme résultant est dénoté *Matching Pursuit Joint* (MPJ) et est décrit ci-dessous :

Algorithm 5 Matching Pursuit Joint (MPJ)

Entrées: $\mathbf{X}$, $\mathbf{\Phi}$
1: $\mathbf{R}^0 := \mathbf{X}$, $n = 0$
2: **Répéter**
3: **Etape 1** : Selection de l'atome $\phi_{\gamma^n} \leftarrow \mathcal{C}(\mathbf{\Phi}, \mathbf{R}^n)$
4: **Etape 2** : Decide si $\phi_{\gamma^n} \in$ source propre ou commune
5: **Si** $\phi_{\gamma^n} \in$ source commune **alors**
6: $\forall i, \mathbf{B_X}[\gamma^n, i] = \langle \phi_{\gamma^n}, R_i^n \rangle$
7: **Sinon**
8: Trouve quelles sources $J \subset I$ contiennent ϕ_{γ^n}.
9: $\forall j \in J, \mathbf{P_X}[\gamma^n, j] = \langle \phi_{\gamma^n}, R_j^n \rangle$
10: **Fin Si**
11: **Etape 3** : Mise à jour :
$\mathbf{R}^n = \mathbf{X} - (\mathbf{B_X} \cdot \mathbf{\Phi} + \mathbf{P_X} \cdot \mathbf{\Phi})$
$n \leftarrow n + 1$
12: **Jusqu'à** condition d'arrêt
Sorties: $\mathbf{R}^n$, $\mathbf{B_X}$ and $\mathbf{P_X}$

L'algorithme est initialisé avec une matrice résiduelle $\boldsymbol{R^0} = \boldsymbol{X}$, les atomes sont sélectionnés séquentiellement selon un critère $\mathcal{C}(\mathbf{\Phi}, \mathbf{R}^n)$ et attribués soit à $\boldsymbol{B_X}$ soit à $\boldsymbol{P_X}$ selon une règle décrite plus loin. La notation $\boldsymbol{B}[n_l, n_c]$ est utilisée pour indiquer l'élément situé sur la n_l-ième ligne et la n_c-ième colonne de la matrice $\boldsymbol{B}$. Cet algorithme se distingue du SOMP [TGS06] par l'étape 2 d'attribution des atomes à l'une ou l'autre des matrices, et par l'utilisation d'une règle de mise à jour standard.

Adaptation des atomes Nous avons vu qu'une des difficultés est de prendre en compte les légères variations qui peuvent survenir entre deux occurrences d'un même motif. Par exemple, les composantes rythmiques peuvent être légèrement décalées, et plus généralement, il faut considérer que les musiciens ne jouent jamais deux fois de la même façon un motif donné. L'algorithme MPJ n'autorise qu'une variation d'amplitude entre les occurrences. Si on utilise un dictionnaire d'atomes localisés dans le plan temps-fréquence, chaque atome attribué à la source commune aura une localisation identique dans tous les mélanges (le coefficient de projection sera en revanche, spécifique pour chaque mélange).

Pour introduire de la flexibilité et prendre en compte ces petites variations d'exécutions, il faut adapter cette localisation en cherchant pour chaque mélange, la localisation optimale dans un voisinage. Cette technique s'apparente à celle des *Matching Pursuit* Adaptatif (voir 3.3.3 page 46) elle

est donc appelée *Matching Pursuit Adaptatif Joint* (MPAJ) et est décrite par l'algorithme (6) où l'adaptation se fait uniquement sur la localisation temporelle des atomes à l'aide d'un paramètre de décalage.

Algorithm 6 Matching Pursuit Adaptatif Joint (MPAJ)

Entrées: $\mathbf{X}$, $\mathbf{\Phi}$

$\mathbf{R}^0 := \mathbf{X}$, $n = 0$

Répéter

  Etape 1 : Selection de l'atome $\phi_{\gamma^n} \leftarrow \mathcal{C}(\mathbf{\Phi}, \mathbf{R}^n)$

  Etape 1-bis : Adaptation locale de l'atome au mélange : $\forall i, \tau_n^i = \arg\max_\tau |\langle (\phi_{\gamma^n} * \delta_\tau), R_i^n \rangle|$

  Etape 2 : Decide si $\phi_{\gamma^n} \in$ source(s) propre(s) ou commune

  Si $\phi_{\gamma^n} \in$ source commune **alors**

    $\forall i, \mathbf{B_X}[\gamma^n, i] = \left\langle (\phi_{\gamma^n} * \delta_{\tau_n^i}), R_i^n \right\rangle$

    $\forall i, R_i^{n+1} = R_i^n - \mathbf{B_X}[\gamma^n, i](\phi_{\gamma^n} * \delta_{\tau_n^i})$

  Sinon

    Trouve quelles sources $J \subset I$ contiennent ϕ_{γ^n}.

    $\forall j \in J, \mathbf{P_X}[\gamma^n, j] = \left\langle (\phi_{\gamma^n} * \delta_{\tau_n^j}), R_j^n \right\rangle$

    $\forall j \in J, R_j^{n+1} = R_j^n - \mathbf{P_X}[\gamma^n, j](\phi_{\gamma^n} * \delta_{\tau_n^j})$

  Fin Si$n \leftarrow n + 1$

Jusqu'à condition d'arrêt

Sorties: $\mathbf{R}^n$, $\mathbf{B_X}$, $\mathbf{P_X}$ et τ_n^i

8.2.2 Critères de sélection

Le critère de sélection reprenant de façon naturelle celui d'un Matching Pursuit simple est :

$$\mathcal{C}_S(\mathbf{\Phi}, \boldsymbol{R}^n) = \arg\max_{\phi \in \Phi} \sum_{i=0}^{I-1} r_i^n(\phi) \tag{8.2.1}$$

où $r_i^n(\phi)$ dénote la projection de l'atome ϕ sur le résiduel R_i^n du mélange X_i à l'itération n :

$$r_i^n(\phi) = |\langle R_i^n, \phi \rangle|^2 \tag{8.2.2}$$

Le critère (8.2.1) est un simple critère énergétique qui évalue la corrélation moyenne d'un atome avec les mélanges. On le retrouve par exemple dans [GRS07]. Ce critère ne présume pas de l'appartenance de l'atome sélectionné à la composante commune ou un sous-ensemble de composantes propres.

Alternativement, on peut définir un critère de sélection minimisant le risque de sélectionner un atome n'appartenant pas à la composante commune. Pour cela on adopte un critère de type minimax :

$$\mathcal{C}_{Min}(\mathbf{\Phi}, \boldsymbol{R}^n) = \arg\max_{\phi \in \Phi} \min_i r_i^n(\phi) \tag{8.2.3}$$

ce critère pénalise très fortement la sélection d'atomes mal corrélés avec un des mélanges. Il est donc particulièrement intéressant pour un problème de type séparation de composante commune, où l'on cherche à retrouver uniquement cette composante partagée par les mélanges. Un critère un peu moins fort est basé sur le produit des projections, que l'on peut aussi formuler comme la somme de leurs logarithmes :

$$\mathcal{C}_{log}(\mathbf{\Phi}, \boldsymbol{R}^n) = \arg\max_{\phi \in \Phi} \sum_{i=0}^{I-1} \log\left(r_i^n(\phi)\right) \tag{8.2.4}$$

	Formule ($\arg\max_{\phi\in\Phi}$)	Stratégie mise en oeuvre	linéaire
$\mathcal{C}_S$	$\sum_{i=0}^{I-1} r_i^n(\phi)$	aucune	oui
$\mathcal{C}_{Min}$	$\min_i\ r_i^n(\phi)$	uniquement source commune	non
$\mathcal{C}_{log}$	$\sum_{i=0}^{I-1} \log\left(r_i^n(\phi)\right)$	favorise source commune	oui
$\mathcal{C}_{Max}$	$\max_i\ r_i^n(\phi)$	favorise sources propres	non
$\mathcal{C}_{Med}$	$\text{median}_i\ r_i^n(\phi)$	favorise sources propres	non
$\mathcal{C}_W$	$w(\phi, \boldsymbol{R}^n) \cdot \sum_{i=0}^{I-1} r_i^n(\phi)$	favorise source commune	non
$\mathcal{C}_P$	$\sum_{i=0}^{I-1} r_i^n(\phi) + \lambda \sum_{i\neq j} \lvert r_i^n(\phi) - r_j^n(\phi)\rvert$	favorise sources propres	non

TABLE 8.2.1: Récapitulatif critères de sélection

A l'inverse, si l'on cherche explicitement les atomes des sources propres, on peut définir le critère :

$$\mathcal{C}_{Max}(\boldsymbol{\Phi}, \boldsymbol{R}^n) = \arg\max_{\phi\in\Phi} \max_i r_i^n(\phi) \tag{8.2.5}$$

ou de façon moins radicale, remplacer le critère de moyenne de $\mathcal{C}_S$ par la médiane :

$$\mathcal{C}_{Med}(\boldsymbol{\Phi}, \boldsymbol{R}^n) = \arg\max_{\phi\in\Phi} \text{median}\ r_i^n(\phi) \tag{8.2.6}$$

On peut voir que c'est le profil des projections $r_i^n(\phi)$ qui caractérise l'appartenance d'un atome à l'une ou l'autre des sources. On peut donc dériver deux autres critères prenant explicitement en compte ce profil :

$$\mathcal{C}_W(\boldsymbol{\Phi}, \boldsymbol{R}^n) = \arg\max_{\phi\in\Phi} w(\phi, \boldsymbol{R}^n) \cdot \sum_{i=0}^{I-1} r_i^n(\phi) \tag{8.2.7}$$

$$\mathcal{C}_P(\boldsymbol{\Phi}, \boldsymbol{R}^n, \lambda) = \arg\max_{\phi\in\Phi} \sum_{i=0}^{I-1} r_i^n(\phi) + \lambda \sum_{i\neq j} \lvert r_i^n(\phi) - r_j^n(\phi)\rvert \tag{8.2.8}$$

$\mathcal{C}_W(\boldsymbol{\Phi}, \boldsymbol{R}^n)$ est une version pénalisée de $\mathcal{C}_S(\boldsymbol{\Phi}, \boldsymbol{R}^n)$ par le rapport des moyennes géométriques et arithmétiques des projections (en valeur absolue) :

$$w(\phi, \boldsymbol{R}^n) = \frac{\sqrt[I]{\prod \lvert\langle R_i^n, \phi\rangle\rvert}}{\frac{1}{I}\sum \lvert\langle R_i^n, \phi\rangle\rvert} \leq 1 \tag{8.2.9}$$

cette mesure (également appelée entropie de Wiener) est en effet un descripteur de *flatness,* c'est à dire de la faible dispersion des échantillons considérés. Ce score mesure la disparité des coefficients $r_i^n(\phi)$ et pénalise la sélection d'atomes pour lesquels cette répartition présente un profil très variable. La borne supérieure 1 est au contraire atteinte si tous les échantillons sont identiques. En particulier, ce score est nul si l'une des projections $r_i^n(\phi)$ est nulle.

Enfin le critère $\mathcal{C}_P(\boldsymbol{\Phi}, \boldsymbol{R}^n, \lambda)$ est une pénalisation de $\mathcal{C}_S(\boldsymbol{\Phi}, \boldsymbol{R}^n)$ par la somme des écarts entre les projections. A l'inverse de $\mathcal{C}_W(\boldsymbol{\Phi}, \boldsymbol{R}^n)$, ce critère donne des scores supérieurs aux atomes dont les projections sur les mélanges ont de fortes disparités. Ces quatre critères expriment chacun une stratégie délibérée soit de favoriser la sélection d'atomes pour la composante commune, soit pour les composantes propres.

8.2.3 Répartition des atomes dans les sources

Une fois sélectionné, on doit décider de l'attribution d'un atome au support de la source commune (dans la matrice B_X) ou dans ceux d'un sous-ensemble de sources propres (dans la matrice P_X).

C'est l'étape 2 des algorithmes MPJ et MPAJ. Soit ϕ_{γ^n} l'atome choisi à l'itération n, pour décider de son attribution, il faut étudier la répartition des coefficients $r_i^n(\phi_{\gamma^n})$. Pour cela, considérons ces valeurs comme des échantillons d'une variable aléatoire réelle r. Ainsi, à chaque itération, les $r_i^n(\phi_{\gamma^n})$ présentent un tirage de I valeurs de r. Un estimateur statistique de la dispersion de r est défini pour un ensemble $\{r_i\}$ de I échantillons par l'écart type relatif σ_r :

$$\sigma_r(\{r_i\}) = \frac{1}{\sqrt{I}} \cdot |\frac{\sigma}{\mu}| \tag{8.2.10}$$

ou σ et μ sont respectivement l'écart type et la moyenne empirique des échantillons. Il est possible de normaliser cette grandeur par un facteur $\sqrt{I}$ pour assurer que $0 \leq \sigma_r \leq 1$. Cette mesure est d'autant plus faible que la dispersion des valeurs est petite. Pour illustrer son intérêt, prenons l'exemple de I mélanges X_i qui vérifient $\forall i, X_i = X_c + P_i$. La projection de ces mélanges dans un dictionnaire Φ est donnée par :

$$\boldsymbol{A} = \boldsymbol{\Phi}^T \cdot \boldsymbol{X} \tag{8.2.11}$$

où chaque ligne m de la matrice $\boldsymbol{A} \in \mathbb{R}^{M \times I}$ contient les projections $r_i^0(\phi_m)$ de l'atome ϕ_m sur les mélanges. Avant d'observer un cas réel, étudions un cas dégénéré où les projections sont binaires. Pour une ligne m, et une colonne i de $\boldsymbol{A}$, un 0 indique que l'atome ϕ_m a une corrélation nulle avec le i-ième mélange. A l'inverse, un 1 indique une corrélation forte. Dans ce cas très simple, si un atome ϕ_m appartient à la source commune, la ligne correspondante de $\boldsymbol{A}$ ne contient que des 1. Si en revanche, il appartient à un sous-ensemble de sources propres, la ligne m de A contient au moins un 0.

Observons les valeurs caractéristiques de la mesure σ_r dans ce cas de figure simplifié, par exemple pour $I = 3$ mélange. L'ordre des colonnes n'a ici pas d'influence car la mesure σ_r est invariante aux permutations des éléments. On note alors $P(\{b_0, b_1, b_2\})$ une permutation quelconque des éléments binaires b_0, b_1, b_2. Il y a donc quatre cas de figure :

- $\sigma_r(P(\{0,0,1\})) = 1$ Configuration où l'une des projections $r_i(\phi)$ est très grande devant les autres. Ce cas de figure indique que l'atome est corrélé avec l'une des sources propres, mais ni avec les autres, ni avec la source commune.

- $\sigma_r(P(\{1,1,1\})) = 0$ Configuration où les projections $r_i(\phi)$ sont égales. Ce cas de figure indique que l'atome est corrélé avec la source commune à tous les mélanges

- $\sigma_r(P(\{0,1,1]\})) = 0.5$ Configuration où l'atome est corrélé avec un sous-ensemble des mélanges (deux parmi trois) , il peut donc appartenir à plusieurs sources propres, mais pas à la source commune.

- $\sigma_r(P(\{0,0,0]\})) = 0$ Configuration où les projections $r_i(\phi)$ sont égales et nulles. Ce cas de figure indique que l'atome n'est corrélé avec aucun des mélanges. En pratique, un tel atome ne sera pas sélectionné par l'algorithme.

Dans le cas plus général de I sources, on peut de même montrer que I modes principaux (*c-à-d.* non tous nuls) vont émerger. Soit $J \leq I$ le nombre de composantes non nulles d'un échantillon binaire $\{r_i\} \in \{0,1\}^I$ de taille I. On note $P(J|I)$ une permutation quelconque des éléments de l'échantillon :

$$\begin{aligned}
\mu(P(J|I)) &= \frac{J}{I} \\
\sigma(P(J|I)) &= \sqrt{\frac{1}{I-1}\sum_{i=1}^{I}\left(x_i - \mu(P(J|I))\right)^2} \\
&= \sqrt{\frac{1}{I-1}\left[J\cdot(1-\mu(P(J|I)))^2 + (I-J)\cdot(\mu(P(J|I)))^2\right]} \\
&= \sqrt{\frac{1}{I-1}J\cdot(1-\mu(P(J|I)))}
\end{aligned}$$

Le cas $J = 0$ présente peu d'intérêt et on peut poser $\sigma_r(P(0|I)) = 0$, pour $J > 0$:

$$\begin{aligned}
\sigma_r(P(J|I)) &= \tfrac{1}{\sqrt{I}}\cdot \quad \frac{\sigma(P(J|I))}{\mu(P(J|I))} \\
&= \tfrac{1}{\sqrt{I}}\cdot \quad \frac{I}{J}\cdot\sqrt{\frac{1}{I-1}J\cdot\left(1-\frac{J}{I}\right)} \\
\sigma_r(P(J|I)) &= \sqrt{\frac{I-J}{J\cdot(I-1)}} \qquad (8.2.12)
\end{aligned}$$

on obtient ainsi les valeurs des modes pour les cas dégénérés où un atome à une corrélation unitaire avec J mélanges parmi I.

Plus l'atome sélectionné est corrélé à un sous-ensemble réduit de mélanges, plus σ_r sera proche de 1 et l'on peut attribuer cet atome à un sous-ensemble de sources propres. A l'inverse, une mesure σ_r proche de zéro indique une corrélation répartie sur tous les mélanges et incite à attribuer l'atome à la source commune.

La figure 8.2.1 montre les histogrammes obtenus sur des mélanges synthétiques. La source commune est un extrait d'un signal de voix d'homme, les sources propres sont du glockenspiel, un signal d'orchestre, un signal de trompette et une voix de femme. A chaque fois, une source commune est présente dans tous les mélanges. Plus un atome a une corrélation répartie sur un grand nombre de mélanges, plus son écart-type relatif est proche de 0. Le cas $I = 2$ correspond donc à deux mélanges :

1. Voix d'homme + Glockenspiel
2. Voix d'homme + Orchestre

L'histogramme montre une répartition des valeurs de dispersion des projections très claire, permettant de distinguer aisément si un atome est dans le support de la source commune ($\sigma_r \simeq 0$) ou d'une source propre ($\sigma_r \simeq 1$). On voit que plus le nombre de mélanges est grand, moins il sera aisé de discriminer de tels atomes car les modes sont plus proches. Ce résultat peut paraître paradoxal (plus il y a mélange et moins la séparation est aisée), mais il reflète le fait que les différentes sources n'ont pas des supports disjoints. Pour le cas $I = 2$, la composante source (voix d'homme) a un support très différent des composantes propres (signaux musicaux) et la séparation peut se faire avec une incertitude très faible. En revanche, lorsque les composantes propres ont des supports qui recouvrent celui de la composante commune (ou qui se recouvrent entre elles), alors l'incertitude augmente car un même atome peut servir pour un sous-ensemble de composantes propres comme pour la composante commune.

Connaissant le nombre de sources, on peut fixer une valeur de seuil τ_r en dessous de laquelle la dispersion d'un atome est considérée suffisamment faible pour l'affecter à la source commune. Au dessus de ce seuil, on l'assignera à un sous-ensemble de sources propres. En première approximation,

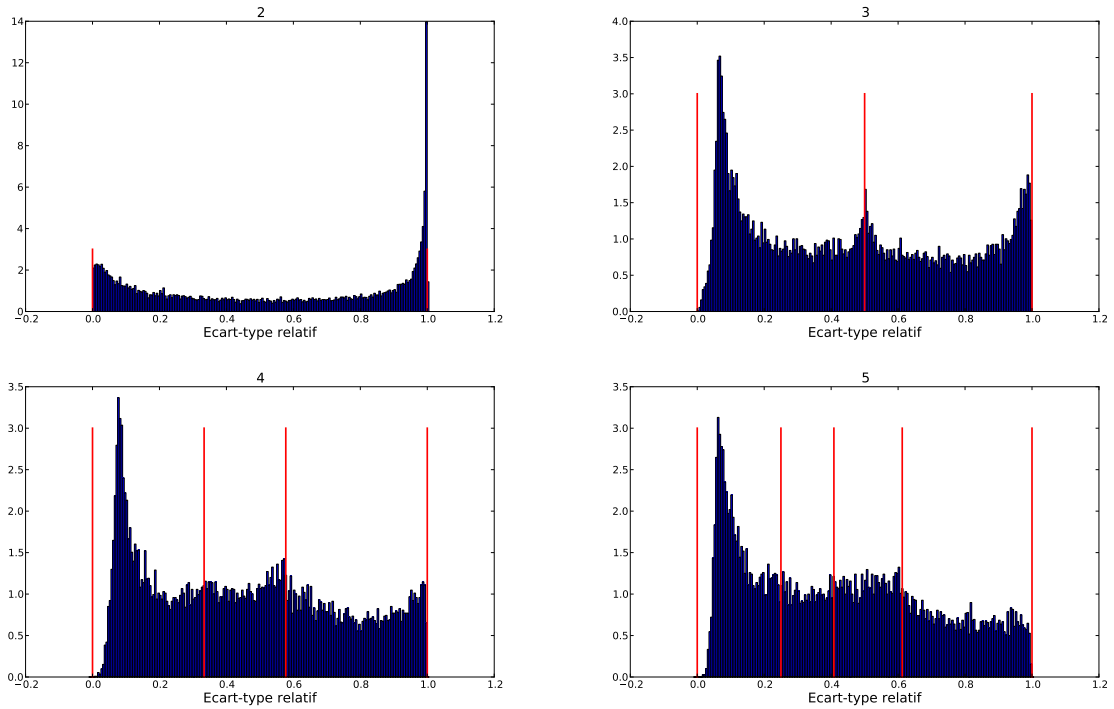

FIGURE 8.2.1: Histogrammes des écarts-types relatifs σ_r empiriques obtenus pour I mélanges synthétiques de signaux audio (2 secondes) pour $I = 2, 3, 4$ et 5. En rouge les modes calculés à l'aide de (8.2.12). Le dictionnaire est une union de 3 bases MDCT. Pour 2 mélanges (en haut à gauche) on distingue deux modes selon qu'un atome est bien corrélé avec les deux mélanges (valeurs proches de zéro) et doit être attribué à la source commune, ou à un seul (valeurs proches de 1) et doit être attribué à une seule source propre. Pour le cas $I = 3$ (en haut à droite), un nouveau mode apparait (autour de 0.5) qui indique des atomes bien corrélés avec 2 mélanges parmi 3, soit un sous-ensemble de sources propres. Plus le nombre de mélanges augmente, moins les différents modes sont faciles à distinguer.

on pourra prendre comme seuil la moitié de la valeur $\sigma_r(P(I-1|I))$ qui vaut pour le cas où l'atome est corrélé à tous sauf un seul des mélanges

$$\begin{aligned} \tau_r &= \frac{1}{2}\sigma_r(P(I-1|I)) \\ &= \frac{1}{2\cdot(I-1)} \end{aligned} \tag{8.2.13}$$

Ainsi pour $I = 2$ sources on fixera ce seuil à 0.5, puis à 0.25 pour 3 sources et ainsi de suite. En pratique néanmoins, comme on peut l'observer sur la Figure 8.2.1, au delà de 3 sources, ce seuil ne prend plus suffisamment en compte les interférences (qui se traduisent notamment par un décalage du mode en zéro), et il conviendra de l'augmenter un peu.

La méthode que nous adoptons présente des similitudes avec une approche classique de la séparation de sources aveugle : l'algorithme DUET présenté par YILMAZ et RICKARD [YR04] qui estime des masques temps-fréquence à partir de la recherche de pics dans des histogrammes. Il y a néanmoins des différences fondamentales. D'abord, DUET s'intéresse à la répartition de sources à travers les différents canaux d'un enregistrement. Ensuite, ici le choix de l'appartenance à l'une ou l'autre des sources se fait pour chaque atome sélectionné par l'algorithme.

Nom	Plage de Valeurs	Remarques
$\mathcal{C}$	voir 8.2.1	contrôle la stratégie de sélection des atomes
τ_r	autour de $\frac{1}{2(I-1)}$	contrôle la répartition des atomes dans B_X ou P_X
$Nmax$	$\mathbb{N}$	nombre d'atomes de la décomposition
ϵ	$\mathbb{R}$	erreur de reconstruction cible $\|\boldsymbol{X} - \boldsymbol{\Phi} \cdot \boldsymbol{C_X}\|_F^2$

TABLE 8.3.1: Tableau des paramètres de contrôle de MPAJ

8.3 Évaluations

Nous avons évalué l'algorithme MPAJ avec différents critères de sélection sur des exemples synthétiques ainsi que sur des répétitions de motifs musicaux (phrase instrumentale, couplets etc..). Il est utile de rappeler ici que MPAJ est avant tout un algorithme de construction d'approximations parcimonieuses, modifié pour la séparation de sources. Nous pouvons donc évaluer ses performances sur chacune de ces tâches. Nous présentons auparavant les différents paramètres et les métriques utilisées.

8.3.1 Paramètres et métriques

Paramètres MPAJ est un algorithme de type Matching Pursuit, son réglage s'appuie sur un nombre très limité de paramètres. Ceux-ci sont donnés dans le tableau 8.3.1.

Métriques de séparation de sources. L'évaluation de la qualité d'une séparation de sources doit prendre en compte différentes erreurs (reconstruction, interférences, artefacts, ...). Dans l'article de VINCENT *et al* [VGF06], on trouve une définition de ces différentes erreurs ainsi qu'une boite à outil pour leur évaluation (*BSS_Eval*[1]). Dans ce travail, nous utilisons trois des mesures définies dans l'article, à savoir :

- Le SDR : *Sources-to-Distortion Ratio*, la métrique de référence. Analogue au SRR, ce score décrit la qualité de reconstitution d'une source. Plus ce score est haut, plus cette qualité est grande
- Le SIR : *Sources-to-Interference Ratio*. Ce score décrit la présence des autres sources dans une source séparée, c'est à dire l'erreur de répartition commise entre les sources.
- Le SAR : *Sources-to-Artefact Ratio* : Ce score décrit le niveau d'artefacts (éléments non désirables).

Ces trois mesures, en échelle logarithmique (dB) sont couramment utilisées dans la littérature et permettent une évaluation objective de la qualité d'une séparation étant données les sources originales.

L'algorithme proposé est itératif, il est donc possible de visualiser l'évolution de ces métriques au cours d'une décomposition. A l'itération n, MPAJ fournit pour chaque mélange X_i deux approximations $\tilde{B}_i^n$ et $\tilde{P}_i^n$ respectivement de la source commune et de la source propre.

La Figure 8.3.1 montre l'évolution des scores pour un cas synthétique de décomposition jointe de 3 mélanges linéaire instantanés, $\forall i = 1..3$, $X_i = X_c + P_i$. Le signal commun utilisé est une voix d'homme, les sources propres sont glockenspiel, orchestra et voix de femme. Les mesures montrent qu'il est plus simple de séparer la voix du glockenspiel (SIR, SDR et SAR de P_0 élevés) que d'une autre voix ou d'un signal d'orchestre (scores de P_1 et P_2 plus faibles). Ceci est dû au fait que l'énergie de ces signaux se superposent dans le plan temps-fréquence. Dans le cas du glockenspiel par exemple, il y

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

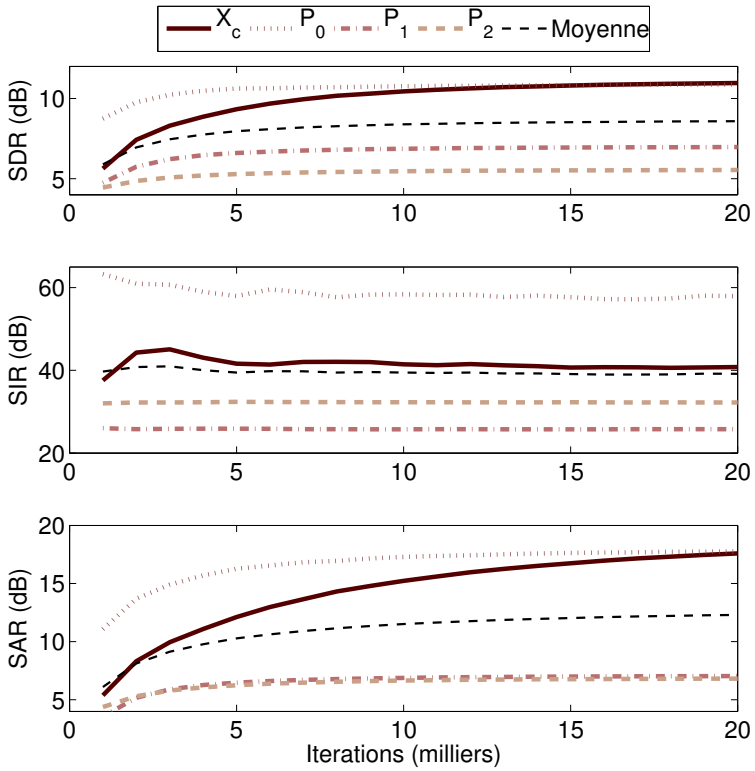

FIGURE 8.3.1: Scores de séparation en fonction du nombre d'itération de MPAJ pour un critère de sélection $\mathcal{C}_P$ sur des mélanges synthétiques (linéaires instantanés). Ici, X_c est un signal de voix d'homme, P_0, P_1 et P_2 sont respectivement des signaux de glockenspiel, d'orchestre et de voix de femme.

a moins de recouvrement et la séparation est plus aisée. A l'écoute, les écarts de valeur se traduisent effectivement par une séparation de meilleure qualité pour le glockenspiel que pour les deux autres composantes propres.

8.3.2 Performances en séparation de sources

Comparaison des critères de sélection sur une tâche simple

Avant de comparer les performances de MPAJ avec celles d'autres méthodes de séparation, nous étudions quel critère choisir parmi les 6 proposés plus haut.

L'expérience suivante à été mise en place. Avec 4 courts extrait audio de 5 secondes (notés Y_i), on crée 4 jeux de 3 mélanges synthétiques (en permutant la source commune et les sources propres). Pour simuler une plus grande variété de simulation, on réalise cette opération pour 3 niveaux différents de mixage (-5, 0 et 5dB) entre la composante commune et la composante propre (le rapport $\frac{\|X_c\|_2}{\|P_i\|_2}$) d'un mélange. Au total 12 jeux de 3 mélanges sont donc disponibles.

L'algorithme MPAJ est lancé sur chacun de ces jeux. Il retourne dans chaque cas un ensemble de 6 signaux séparés $\tilde{Y}_i$, 3 signaux de composante commune (qui ont même support atomique) et 3 composantes propres. Étant donné la nature triviale du problème, les 3 signaux de source commune sont identiques. On évalue les performances de séparation à l'aide des signaux originaux Y_i. Sur cette tâche, on fixe le nombre d'atomes à 5000. Le paramètre τ_r est également fixé à 0.4.

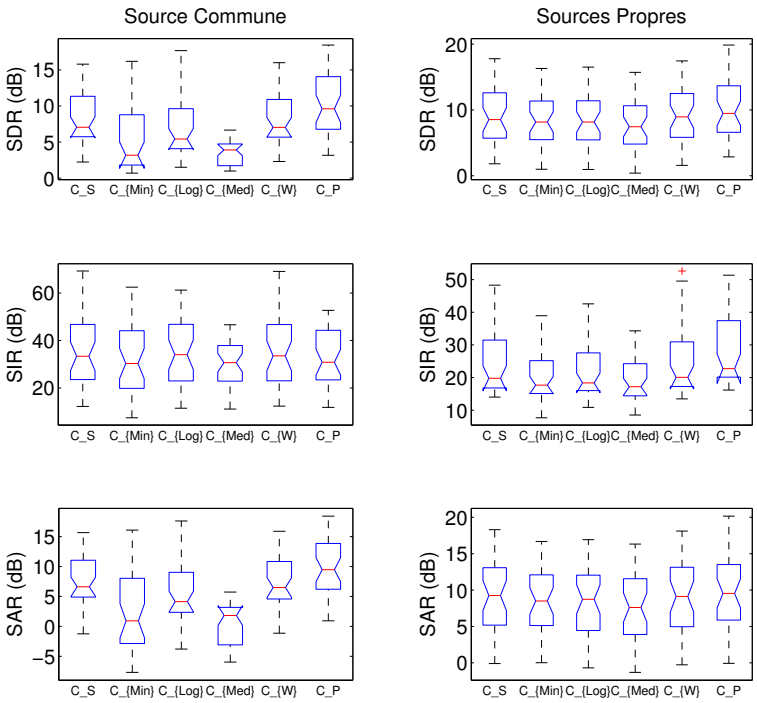

FIGURE 8.3.2: Performances en séparation de sources pour des mélanges synthétiques et différents critères de MPAJ.

L'expérience, dont les résultats sont présentés Figure 8.3.2, disqualifie les critères basés sur le minimum et la médiane. Ceux qui donnent les meilleurs résultats sont $\mathcal{C}_S$, $\mathcal{C}_W$ et surtout $\mathcal{C}_P$ qui arrive en tête pour presque toutes les métriques.

Comparaison avec des méthodes de masquage temps-fréquence

Les méthodes de masquage temps-fréquence effectuent la séparation en appliquant un masque directement sur la représentation temps fréquence (généralement la TFCT). Soit X un signal compris comme le mélange de deux sources B et P, soient $\hat{X}$, $\hat{B}$ et $\hat{P}$ les TFCT respectives de X, B et P, un estimateur de B basé sur un masque temps-fréquence W_B est donné par :

$$\tilde{B} = TFCT^{-1}\left[W_B \odot \hat{B}\right] \tag{8.3.1}$$

où W_B est un masque appliqué sur chaque point temps-fréquence du spectrogramme. W_B peut être construit de façon binaire (seuillage dur comme dans [RP11a]), soit de façon continue ($W_B(f,t) \in [0,1]$) par exemple comme dans un filtre de Wiener [BBG06] (seuillage doux comme dans [LRB+12]). L'autre source est ensuite estimée par :

$$\tilde{P} = TFCT^{-1}\left[(1 - W_B) \odot \hat{P}\right] \tag{8.3.2}$$

Il est possible de généraliser ce principe à un nombre arbitraire de sources, le lecteur intéressé trouvera un état de l'art plus poussé ainsi qu'une justification du filtrage doux dans l'article de BENAROYA *et al* [BBG06] et plus récemment dans le cas de processus gaussien localement stationnaires dans l'article de LIUTKUS *et al* [LBR11]. Dans l'application qui nous concerne, on se limite au cas de séparation de deux sources dans un ensemble de I mélanges X_i. Dans l'article [RP11a], RAFII et PARDO

proposent de construire le masque à partir du calcul d'un spectrogramme $\hat{V}$ obtenu en moyennant les spectrogrammes des mélanges $|\hat{X}_i|^2$:

$$\hat{V}(f,t) = \left(\prod_{i=1}^{I} |\hat{X}_i(f,t)|^2 \right)^{\frac{1}{I}} \tag{8.3.3}$$

Ce spectrogramme moyen sert ensuite à construire pour chaque mélange le masque binaire W_B^i :

$$W_B^i(f,t) = \begin{cases} 1 & \text{si } \left| \log \left(\frac{|\hat{X}_i(f,t)|^2}{\hat{V}(f,t)} \right) \right| \leq t \\ 0 & \text{sinon} \end{cases} \tag{8.3.4}$$

où t est un paramètre dit de tolérance permettant de contrôler la séparation. Dans un article récent [LRB+12], Liutkus *et al* améliorent ce modèle en proposant la construction de masques W_B^i continus et l'estimation du spectrogramme de la source commune par une médiane plutôt que la moyenne, soit :

$$\overline{V}(f,t) \quad = \quad \text{median} \left\{ |\hat{X}_i(f,t)|^2 \right\} \tag{8.3.5}$$

et pour chaque mélange X_i le masque temps-fréquence :

$$W_B^i(f,t) = \exp \left(- \frac{\left(\log |\hat{X}_i(f,t)|^2 - \log \left(\min\{ |\hat{X}_i(f,t)|^2, \overline{V}(f,t) \} \right) \right)^2}{2\lambda^2} \right) \tag{8.3.6}$$

où λ joue ici le même rôle que t. Il est également possible, en lieu et place de la médiane dans (8.3.5), d'utiliser le minimum :

$$\overline{V}(f,t) = \min \left\{ |\hat{X}_i(f,t)|^2 \right\} \tag{8.3.7}$$

Ce principe de filtrage médian doux se retrouve sous de nombreuses formes dans la littérature de séparation de sources [FG10].

Nous comparons les scores de séparation de sources obtenus par filtrage doux (Équation (8.3.6)) avec ceux de MPAJ dans la Figure 8.3.3. Sur ce cas de figure trivial, les performances de séparation, au bout d'un certain nombre d'itérations sont meilleures avec MPAJ pour toutes les métriques. Il est important de rappeler ici que les scores de MPAJ sont calculés sur des approximations des mélanges tandis que les méthodes par filtrage de Wiener utilisent les signaux complets. Ces scores ne témoignent pas toujours d'une qualité perceptive équivalente, mais les deux méthodes présentent des artefacts distincts. On pourrait imaginer construire des masques temps-fréquence à partir de la représentation parcimonieuse.

La pertinence de ces approches repose sur le bon alignement des mélanges. Dans un problème de séparation de source commune tel que l'extraction de fond musical à partir d'une collection de bandes originales de film [LL10], cet alignement est souvent problématique. De même pour la séparation de voix dans une chanson par détection de motifs récurrents. Dans les méthodes REPET [RP11a, LRB+12], si l'estimation des périodes de répétition est imprécise, la séparation va s'en trouver affectée de façon conséquente. Dans la même situation, MPAJ montre une robustesse plus importante, qui lui vient de son caractère adaptatif. La Figure (8.3.4) montre que les performances en séparation sont insensibles à des décalages allant jusqu'à 30 ms entre les mélanges, ce qui correspond à l'ordre de grandeur habituel des trames d'une TFCT. Il faut remarquer ici que les méthodes utilisant des masques doux effectuent généralement un lissage des spectrogrammes pour pallier ce problème.

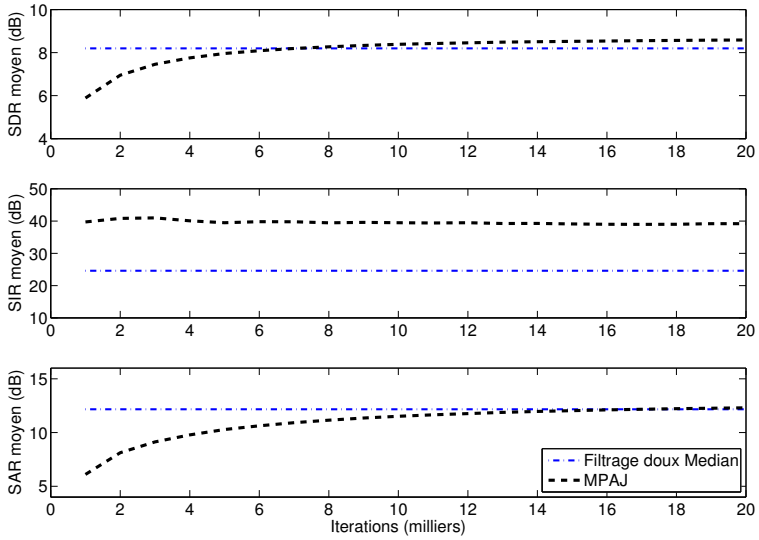

FIGURE 8.3.3: Score moyen de séparation obtenu sur des mélanges linéaires instantanés (3 mélanges) en fonction du nombre d'itérations de MPAJ. Comparaison avec les scores obtenus par Filtrage de Wiener médian.

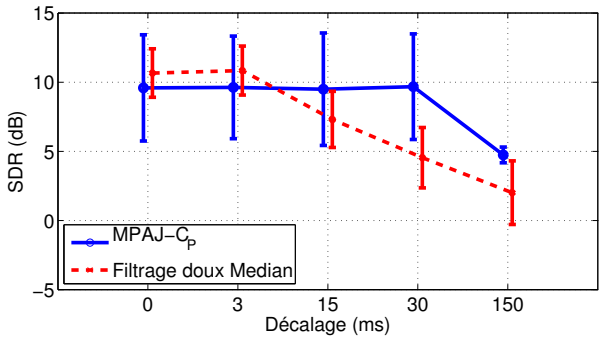

FIGURE 8.3.4: SDR en fonction du décalage moyen absolu entre mélanges. Scores moyens pour 3 mélanges de 5 secondes sur 12 configurations. MPAJ stoppé à 15000 atomes.

Performance sur la séparation de signaux réels Juger de la qualité d'une séparation est toujours un exercice difficile, en particulier lorsqu'il s'agit de comparer des méthodes très différentes. Nous avons mené des expériences de séparation sur des données réelles, notamment sur les *Beach Boys* (sur les titres *'Wouldn't it be nice'*, *'Caroline, No'*, *'I Just Wasn't Made For These Times'*, *'When I Grow up to be a Man'*, et *'All Summer Long'* tous extraits de l'album Pet Sounds), et pour lesquels les sources séparées sont disponibles comme l'ont remarqué FITZGERALD ET GAINZA [FG10]. Pour ces 5 morceaux, nous avons découpé à la main des motifs récurrents (d'une durée de quelques secondes) contenant un fond musical répétitif et des voix variables. Trouver automatiquement ces motifs n'est pas en soi une tâche très complexe (voir chapitre précédent). Il est important de noter que l'algorithme REPET, notamment dans sa version [LRB+12] ne travaille pas sur des segments constitués de cette manière, mais évalue pour chaque point temps-fréquence du spectrogramme une période de répétition locale.

L'évaluation que l'on mène dans ce travail porte donc uniquement sur la comparaison de notre méthode avec la méthode de séparation, à savoir la construction d'un masque doux. Nous considérons

Méthode	3 Versions SDR (dB)	3 Versions SIR (dB)	3 Versions SAR (dB)	4 Versions SDR (dB)	4 Versions SIR (dB)	4 Versions SAR (dB)
	Musique					
Masque-Min	**3.16** ± 1.7	3.41 ± 5.8	**10.03** ± 1.9	**3.47** ± 1.2	3.29 ± 4.7	**11.23** ± 2.0
Masque-Médian	2.49 ± 0.6	8.08 ± 6.4	3.28 ± 1.5	2.62 ± 0.7	7.61 ± 6.3	4.23 ± 1.6
MPAJ - $\mathcal{C}_P$	1.96 ± 0.6	**19.14** ± 7.2	-0.87 ± 2.2	2.06 ± 0.6	**17.42** ± 6.0	-0.60 ± 2.3
	Voix					
Masque-Min	1.67 ± 0.9	**9.96** ± 3.2	0.25 ± 3.0	1.39 ± 0.7	**11.17** ± 3.1	-0.55 ± 2.4
Masque-Médian	2.91 ± 0.6	5.47 ± 2.7	4.71 ± 1.8	2.92 ± 0.4	5.40 ± 2.2	**4.96** ± 1.5
MPAJ - $\mathcal{C}_P$	**3.62** ± 0.8	5.94 ± 2.5	**5.21** ± 1.8	**3.48** ± 1.0	6.03 ± 2.5	4.79 ± 2.3

TABLE 8.3.2: Scores de séparation de source (moyenne et écart type) sur des segments musicaux réels (*The Beach Boys*, *Pet Sounds*). MPAJ est stoppé après 10000 itérations.

néanmoins deux méthodes de constructions, celle basée sur la médiane des spectrogrammes (8.3.5) et celle basée sur le minimum (8.3.7). Les résultats obtenus sont présenté dans le Tableau 8.3.2. La méthode Masque-min privilégie l'estimation du fond musical tandis qu'à l'opposé, MPAJ propose une meilleure reconstruction des sources propres (ici donc, de la voix chantée). La méthode Masque médian présente un cas de figure intermédiaire.

Des exemples sonores de séparation sont disponibles en ligne [2]. On constate à l'écoute une différence de nature entre les artefacts généré par chacune de ces approches. La reconstruction par synthèse que nous adoptons crée notamment des effets de pré-écho dommageables. On peut également noter une atténuations des basses fréquences dans les composantes propres isolées. Les atomes basses-fréquences sont en effet plus facilement attribués à la composante commune. Pour contrer ce défaut, il faudrait sans doute adapter le seuil de répartition en fonction de la fréquence.

En dehors de ces artefacts de synthèse, MPAJ fournit une séparation intéressante, dans laquelle on trouve notamment moins d'interférences (au sens de sources mal isolées) qu'avec les méthodes par masquage.

8.3.3 Performances en approximation

En premier lieu, MPAJ est un algorithme de construction d'approximations jointes. On a vu que compte tenu du modèle adopté, la nature de la parcimonie à l'oeuvre dans les approximations construites permet dans certains cas une séparation de la composante commune. Néanmoins, MPAJ est avant tout destiné à traiter le problème ($\boldsymbol{SP_0}$) (et pas une version pénalisée de type ($\boldsymbol{SL_{p,q,p',q'}}$)).

Exemples synthétiques

En reprenant les données synthétiques décrites en 8.3.2 on peut comparer les critères de sélection à l'aune de la convergence de l'erreur quadratique de reconstruction qu'ils induisent. La Figure 8.3.5 présente les moyennes obtenues pour les quatre critères les plus performants sur la tâche de séparation de sources.

Le critère $\mathcal{C}_P$ apparaît là aussi comme le plus efficace. Cette observation nous interpelle, elle semble indiquer un lien entre performances pour la tâche d'approximation jointe, et performances pour la séparation de sources. Ce constat est confirmé lorsque l'on fait varier d'autres paramètres que

2. lien temporaire : http ://perso.telecom-paristech.fr/~moussall/Research/DSC/dscPage.html

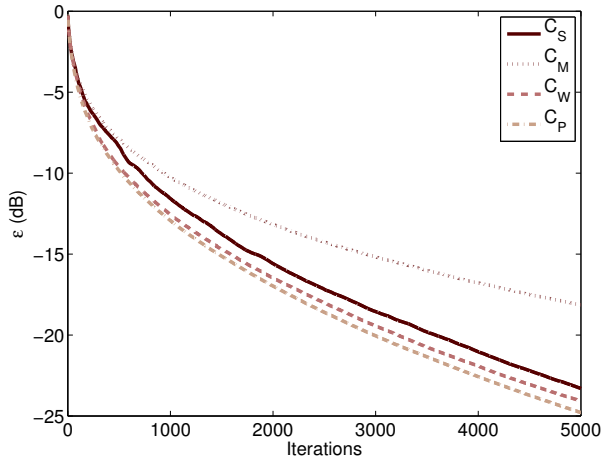

FIGURE 8.3.5: Décroissance de l'erreur quadratique pour MPAJ et différents critères de sélection.

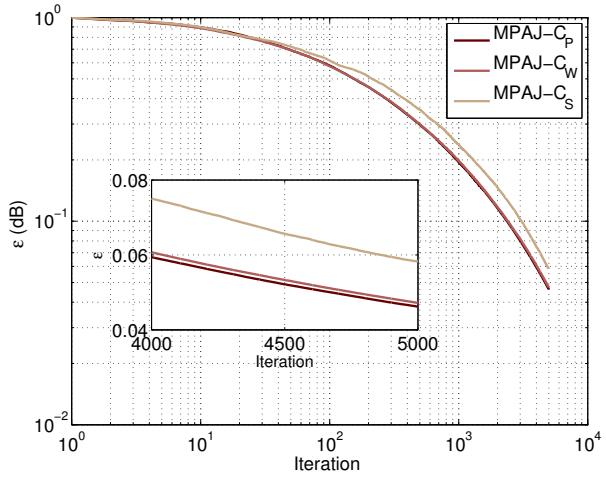

FIGURE 8.3.6: Décroissance de l'erreur quadratique de reconstruction pour MPAJ et différents critères de sélection. En incrustation : zoom sur les 1000 dernières itérations.

le critère de sélection. Par exemple, le seuil τ_r peut lui aussi être changé, et on observe de la même façon que les valeurs optimales pour la séparation et l'approximation sont très proches.

Exemples réels

En reprenant l'exemple des *Beach Boys*, on peut faire les mêmes observations. La Figure (8.3.6) montre la décroissance de l'erreur quadratique (moyenne sur 3 mélanges de 4 à 6 secondes par chanson et 5 chansons). Pour I mélanges X_i cette erreur est définie par :

$$\epsilon(n) = 10\log_{10}\left(\frac{\sum_{i=1}^{I}\|X_i - \tilde{X}_i^n\|_2^2}{\sum_{i=1}^{I}\|X_i\|_2^2}\right) \tag{8.3.8}$$

où $\tilde{X}_i^n = \tilde{B}_i^n + \tilde{P}_i^n$ est l'approximation du mélange X_i par MPAJ en n itérations. Les critères $\mathcal{C}_W$ et $\mathcal{C}_P$ donnent des résultats assez équivalents, un peu meilleurs que $\mathcal{C}_S$. Les autres critères donnent des performances largement moins intéressantes.

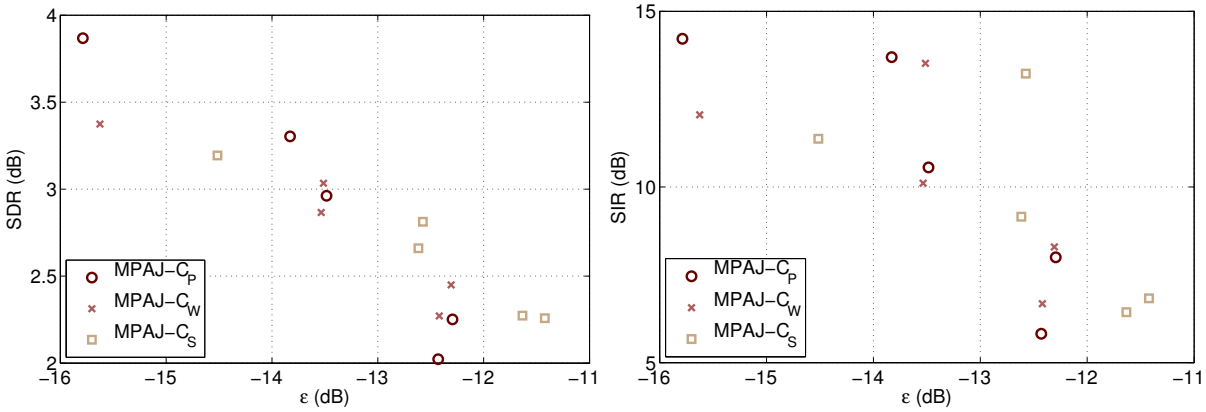

FIGURE 8.3.7: Performances de Séparation (SDR et SIR) en fonction de l'erreur quadratique d'approximation. Plus le marqueur est haut, meilleure est la séparation, plus le marqueur est à gauche, meilleure est l'approximation.

On peut pour finir, visualiser les performances jointes de séparation et d'approximation. La Figure (8.3.7) présente pour chaque morceau de la base des *Beach Boys*, le SDR et SIR en fonction de l'erreur quadratique au bout de 5000 itérations. Cette figure montre assez nettement, quel que soit le critère de sélection choisi, que les objectifs de séparation et d'approximation sont optimisés de façon jointe par MPAJ.

Il y a une conséquence pratique assez intéressante à ce constat. Il est difficile d'évaluer la qualité d'une séparation lorsqu'on ne dispose pas des sources originales. Dans ce cas, les mesures de qualité de reconstruction (de type SRR) peuvent servir de substitut aux métriques traditionnelles de qualité de la séparation de sources. Cette remarque nous permet d'insister sur un point important, MPAJ réalise non seulement une séparation des sources sur un ensemble d'occurrences d'un motif musical, mais il réalise en même temps la première étape d'un codage distribué (le lien peut d'ailleurs être fait avec l'algorithme de Factorisation 4 page 116. Le débit nécessaire pour transmettre la source commune est, étant donnée la structure de la matrice $\boldsymbol{B_X}$, réduit d'un facteur I par rapport à I transmissions indépendantes.

Nous pouvons faire le lien ici avec la métrique de similarité basée sur une distance d'information proposée au chapitre précédent (7.1.2 page 115). La mesure d'une distorsion, étant donné un débit fixé nous donnait alors une indication de la similarité. Ici nous pouvons envisager une mesure de même nature, nous informant cette fois-ci sur la qualité d'une séparation de sources.

Conclusion partielle

Nous avons présenté une méthode de séparation de sources tirant parti d'une information préalable sur la structure d'une scène sonore. L'algorithme réalisant cette séparation est une variante de MP, destinée à construire des solutions sous-optimales à un problème d'approximation simultanée de plusieurs signaux. Le paramétrage de l'algorithme peut être adapté aux cas de figure, et de façon un peu surprenante, les valeurs de paramètres qui donnent de bons résultats en séparation de sources sont aussi celles pour lesquelles l'algorithme présente les meilleures vitesses de décroissance de l'erreur de reconstruction.

Les performances sont globalement comparables à celles des méthodes par masquage temps-

fréquence doux. Si l'approche proposée a une complexité plus importante, elle effectue en plus de la séparation, une compression efficace des sources, analogues à un codage distribué.

Chapitre 9

Conclusion et perspectives

9.1 Conclusions générales

L'archivage de scènes sonores est une tâche composite, à cheval sur plusieurs problématiques de traitement du signal audio. Elle suppose deux capacités : celle de représenter les scènes sonores de façon compacte, fidèle et efficace, et celle de mettre à jour les différents niveaux de structures de ces scènes.

Les représentations parcimonieuses dans des dictionnaires temps-fréquence redondants ont prouvé ces dernières années leur pertinence sur ces différents objectifs. L'addition de modèles de structures permet de faire le lien entre les propriétés de ces représentations et l'organisation interne des scènes sonores. En revanche, si les représentations parcimonieuses structurées sont désormais appliquées avec succès à la compression, le débruitage, la restauration et autres tâches élémentaires de l'archivage, il reste difficile d'envisager un traitement simultané de ces tâches.

Parmi les méthodes de construction de ces représentations, les algorithmes gloutons possèdent deux avantages. Premièrement, leur simplicité autorise leur utilisation sur de grands volumes de données, remplissant ainsi une contrainte forte de l'archivage. Surtout, leur nature itérative, intrinsèquement hiérarchique, permet d'envisager une variation dynamique de leurs paramètres et ainsi d'aborder différentes tâches à différentes profondeurs.

Le premier objectif de cette thèse était donc d'examiner le comportement de ces algorithmes au cours de la décomposition de scènes sonores. Nous avons pour cela présenté un algorithme utilisant une séquence pseudo-aléatoire de sous-dictionnaires, puis une extension dynamique dans laquelle les sous-dictionnaires sont choisis de plus en plus petits au fur et à mesure que le résidu de la décomposition d'un signal se rapproche d'un bruit gaussien.

Cette vision dynamique permet également d'envisager une variation dynamique d'autres paramètres que le dictionnaire, en particulier le critère de sélection des atomes. Pour justifier cette approche, nous avons vu deux exemples de tâches pour lesquelles une modification du critère de sélection présente un intérêt évident :

- La construction d'empreintes acoustiques pour la détection de motifs récurrents. La pénalisation de la sélection d'atomes dans un voisinage temps-fréquence d'atomes précédemment sélectionnés permet d'obtenir des empreintes acoustiques plus informatives.
- L'approximation jointe de signaux/mélanges partageant une source commune. Là aussi, la pénalisation du critère énergétique – par les corrélations des projections – permet de sélectionner

des atomes plus pertinents, à la fois du point de vue de la reconstruction des signaux, mais également selon des critères de séparation de sources.

La flexibilité des algorithmes gloutons est leur meilleur atout pour une tâche d'archivage. L'indexation – qui elle-même comprend différentes tâches à différentes niveaux – se fait en effet à des échelles d'analyse différentes de la compression. En hiérarchisant ces niveaux d'analyse, on voit se dessiner une stratégie complète d'archivage, utilisant une décomposition à paramétrage dynamique.

Lors des premières itérations, le critère de sélection ne doit pas être purement énergétique, mais assumer une part informative, dans l'esprit de la construction d'empreintes acoustiques présentée dans ce travail. A une échelle plus fine, les similarités locales peuvent être détectées, par exemple avec l'algorithme proposé de factorisation des représentations. La connaissance de ces similarités peut alors permettre une décomposition jointe des parties identifiées comme partageant un contenu similaire. Des approches de MP joint comme celle nous avons présentée dans ce travail pour la séparation de sources répétitives, peuvent s'acquitter de cette tâche.

Dans un dernier temps, lors de la phase d'analyse la plus profonde, un codage efficace des scènes sonores par leurs représentations parcimonieuses dans des dictionnaires redondants peut s'appuyer sur l'étude du comportement asymptotique réalisée dans ce travail, et en particulier sur des algorithmes utilisant des séquences pseudo-aléatoires de dictionnaires sous-échantillonnés de plus en plus grossièrement.

Si dans ce travail, nous nous sommes intéressés aux différentes parties de cette architecture hiérarchisée, celle-ci reste encore à être validée expérimentalement dans son ensemble. Ceci représente un défi important, car s'il est relativement facile de comparer des performances sur une tâche en particulier (compression, classification, détection de récurrences, séparation de sources etc..), évaluer une architecture multi-tâches est, en soi, un problème complexe – donc intéressant – à résoudre.

9.2 Perspectives

La première des extensions de ce travail est une validation expérimentale de l'architecture d'archivage.

9.2.1 Archivage semi-automatique de flux radiophoniques

On peut envisager un cas d'utilisation pratique précis, par exemple l'archivage de flux radiophoniques comme effectué par l'INA pour le dépôt légal. Actuellement, l'indexation des flux (c'est à dire quel contenu est diffusé à quel moment) est effectué manuellement, à la fois par la station émettrice et des annotateurs. Le stockage de ces flux est effectué de façon indépendante. En particulier, les annotations ne sont pas utilisées pour réduire le stockage en regroupant les éléments redondants du flux, et cela principalement parce que les annotations manuelles ne sont pas assez précises.

Le système de détection de motifs récurrents présenté dans cette thèse peut fournir un découpage rapide et précis du flux en objets redondants qui faciliterait grandement le travail des annotateurs. Ceux-ci n'auraient plus qu'à annoter chaque objet découvert par l'algorithme. Une telle approche semi-automatique peut améliorer grandement la précision des annotations, en particulier la position exacte des redondances peut être déterminée.

Connaissant les localisations des différentes occurrences d'un objet, il n'est plus nécessaire de stocker les parties du flux concernées. Sous réserve que l'objet soit répété sans trop d'altérations, beaucoup d'espace de stockage peut être économisé en ne codant qu'une seule fois le signal et en

conservant (par exemple dans une table) l'information de localisation des occurrences du motifs dans le flux.

Le codage du motif peut, en outre, être réalisé efficacement en poursuivant la décomposition démarrée pour le calcul de l'empreinte, mais en changeant le critère de sélection des atomes (plus de pénalisation pour voisinage temps-fréquence d'atomes déjà sélectionnés) et en utilisant une séquence de sous-dictionnaires.

Amélioration des empreintes Nous avons présenté dans ce travail une méthode de calcul d'empreintes acoustiques basique. Sans même considérer de changer de dictionnaire, on peut déjà mentionner deux axes d'amélioration des empreintes :

- La construction de clefs plus robustes, dans l'esprit des paires de pics utilisées dans les méthodes de l'état de l'art [Wan06, FRG11].
- L'adoption d'autres critères de sélection des atomes. Dans ce travail, nous avons utilisé un critère énergétique (maximum de la valeur absolue des projections du résiduel sur les atomes du dictionnaire) pénalisé par un masque temps-fréquence construits à partir des atomes déjà sélectionnés. Cette solution *ad hoc* donne des résultats déjà intéressants, mais il serait plus satisfaisant de pouvoir formaliser cette pénalisation comme une contrainte, par exemple, de maximisation de l'information apportée par le choix d'un atome, étant donnés ceux sélectionnés précédemment.

Un modèle de parcimonie structurée sur le support des décompositions superficielles doit alors être proposé. Si les modèles de type Machine de Boltzmann proposés récemment [PEE12, DHD12] nous semblent prometteurs, de très fortes contraintes pratiques restreignent pour l'instant leur utilisation directe sur des problèmes de grande dimension, telles que ceux qui nous intéressent dans cette étude.

9.2.2 Amélioration des codeurs basés sur des représentations parcimonieuses

Un codeur audio complet se caractérise généralement par une structure à deux étages :

1. Un étage d'analyse du signal, qui prend en entrée le signal numérique (p.ex. une forme d'onde) et rend en sortie un ensemble de paramètres et coefficients calculés sur le signal (*p. ex.* les coefficients d'une représentation parcimonieuse dans un dictionnaire redondant)
2. Un étage de codage de source, qui transforme les paramètres et coefficients d'analyse en un flux binaire prêt à être transmis ou stocké. Cette étape contient également la phase de quantification.

Dans cette thèse, nous nous sommes concentrés sur la première étape et avons utilisé un codage de source naïf (quantification uniforme, codage uniforme des coefficients) pour illustrer expérimentalement la validité de SASMP. Une évaluation équitable avec, par exemple le codeur proposé par Ravelli nécessite d'utiliser un codage de source plus efficace. En particulier, il sera important d'introduire un modèle psycho-acoustique.

En effet, nous avons dans ce travail uniquement considéré que le critère d'erreur (le terme d'attaches aux données entre notre modèle et le signal d'entrée) était une simple norme Euclidienne, mais cette mesure n'est pas toujours pertinente pour évaluer la qualité d'un codage de scène sonore. Des mesures perceptives objectives sont souhaitables pour pouvoir juger de la qualité d'une approche. Dans une seconde phase, des tests d'écoute subjectifs pourront être menés.

Parmi les lacunes de la méthode présentée, on peut lister :

- L'absence de mécanisme de contrôle de pré-écho ([SSDR08, RRD08]). Ce type d'artefacts apparait fréquemment lorsqu'on utilise des dictionnaires dont les atomes sont peu ou prou des cosinus modulés par une fenêtre symétrique.
- Une complexité accrue par rapport aux codeurs existants. De gros efforts d'optimisation doivent être conduits dans l'implémentation de la méthode pour permettre son utilisation pratique.
- L'extension au cas stéréophonique et multicanal. Nous nous sommes dans cette étude, cantonnés au cas monophonique, mais la plupart des scènes sonores rencontrées actuellement sont au moins stéréophoniques. Des redondances fortes entre les canaux sont à attendre, et leur exploitation au travers de méthodes telles que proposées dans ce travail peut être intéressante. Mais il faudra, dès lors se comparer aux méthodes de codage audio multicanal.

Chacune de ces limitations doit être prise en compte avant qu'un codeur réellement compétitif puisse être développé.

Codage distribué de scènes sonores répétitives En audio, la grande majorité des techniques de codage joint portent sur le codage de signaux multicanaux. Une vue d'ensemble des techniques actuelles de codage spatial audio se trouve dans l'article de Elfriti *et al* [EGK11]. Bien que le cadre théorique soit tout à fait pertinent, le théorème de Slepian-Wolf est rarement utilisé dans ces travaux. De même, les quelques travaux portant sur le codage audio distribué (p.ex. les travaux de Roy et Vetterli [RV07] ou encore de Matta et Creusere [MC09]) sont centrés sur le cas de redondances spatiales, généralement l'acquisition d'une scène par un réseau de microphones.

Nous avons vu dans ce travail un cadre différent où le codage audio distribué peut s'appliquer, celui des redondances temporelles. Ce type de compression de scènes sonores répétitives est relativement nouveau. Dans certains cas (*p.ex.* les musiques électroniques et techno), la redondance temporelle est très importante. Une réduction substantielle du débit est alors possible. Le codage joint des parties répétitives est un axe d'amélioration des codeurs bas-débit qui nous parait très prometteur. Les gains en débit potentiels sont, selon nous, d'un ordre de grandeur plus importants que ceux qu'il est encore possible d'obtenir par codage traditionnel, où les scènes sonores sont découpées en trames codées indépendamment.

Malheureusement, cette compression est conditionnée à la détection efficace de ces redondances temporelles. Un système de codage audio distribué de scènes sonores répétitives ne peut s'affranchir de ce pré-traitement.

9.2.3 Séparation de sources répétitives

Dans le même temps, nous avons vu qu'une décomposition parcimonieuse jointe des différentes occurrences d'un motif redondant pouvait permettre une séparation de la composante commune et des sources variables. Là aussi, on trouve principalement dans l'état de l'art ce type de méthode pour des problèmes multicapteurs (par exemple de débruitage [LT06, TGS06] ou de séparation de sources spatiales dans un cadre multicanal [Gri02, GRS07]).

En utilisant la mesure de similarité proposée au chapitre 7 et l'algorithme de séparation de sources du chapitre 8, il doit être possible de proposer un système capable de séparer la voix dans une piste complète de musique populaire, dans l'esprit de l'algorithme REPET proposé par Rafii [RP11a] et amélioré dans [LRB+12].

Parallèlement, il reste à déterminer des moyens de réduction des artefacts induits par la séparation par synthèse parcimonieuse dans des dictionnaires redondants.

Dé-construction de musiques électroniques Une part importante (et en progression) de la musique populaire produite actuellement est construite à l'aide de sons très courts (appelés *samples* par les musiciens électro) apparaissant régulièrement et de façon rigoureusement identique dans un morceau. Dans certains cas, la piste entière peut se comprendre comme, une superposition dans le temps et l'espace d'un nombre limité de *samples.* Dans ce genre de cas, il doit être possible à partir de la piste de mélange, de détecter différents niveaux de redondance et d'en déduire les *samples.*

Ce cas particulier peut permettre de constituer une base d'évaluation de méthodes de séparation bien maîtrisée. Il présente de plus une application susceptible d'intéresser un large public.

A plus longue échéance..

Si l'on considère le déluge de données auquel les archivistes contemporains sont (et seront de plus en plus) confrontés, la nécessité de trier ce qui, parmi ces données, constitue de l'information, ne peut que croître dans les prochaines années. Certes, la puissance de calcul et les ressources disponibles augmenteront, mais le principal atout de l'archiviste devra être sa compréhension de l'organisation intrinsèque des données. Toutes les scènes sonores (musique, voix, cris d'animaux, sons environnementaux etc..) sont construites et structurées sur différents niveaux. Les modèles de parcimonie, notamment structurée, et des méthodes appropriées nous permettent de *déconstruire* les signaux, et d'expliciter ces niveaux structurels.

Mais plus profondément encore, les enregistrements de scènes sonores sont comme les paroles gelées imaginées par Rabelais, ils capturent une *expérience,* ils figent une part de la réalité du monde et peuvent la restituer. Ce niveau ultime de structure, le plus complexe, c'est celui du langage. Les scènes sonores décrivent le monde dans une langue qu'il nous reste encore à apprendre.

Quatrième partie

Annexes

Annexe A

Modélisation des poursuites à l'aide de statistiques d'ordre

A.1 Statistiques d'ordre

Commençons par introduire quelques outils fournis par la théorie des statistiques d'ordre. Le lecteur intéressé trouvera des détails par exemple dans [Nag90, GNP72]. Soit $z_1, z_2, ..z_n$, n échantillons i.i.d d'une variable aléatoire continue Z de densité de probabilité f_Z et de fonction de répartition F_Z. On dénote par $Z_{1:n}, Z_{2:n}, .., Z_{n:n}$ les statistiques d'ordre de Z. $Z_{i:n}$ est une variable aléatoire continue représentant le $i^{ième}$ plus petit élément parmi les n échantillons. La densité de probabilité de $Z_{i:n}$ est dénotée $f^Z_{i:n}$ et son expression est donnée par :

$$f^Z_{i:n}(z) = \frac{n!}{(n-i)!(i-1)!} F_Z(z)^{i-1} f_Z(z)(1-F_Z(z))^{n-i} \tag{A.1.1}$$

La densité du maximum se déduit de cette formule :

$$f^Z_{n:n}(z) = nF_Z(z)^{n-1} f_Z(z) \tag{A.1.2}$$

Enfin le moment d'ordre m de $Z_{i:n}$ est noté $\mu^{(m)}_{i:n}$:

$$\mu^{(m)}_{i:n} = \mathbb{E}(Z^m_{i:n}) = \int_{-\infty}^{\infty} z^m f^Z_{i:n}(z) dz \tag{A.1.3}$$

Dans un souci de clarté on notera directement l'espérance par $\mu_{i:n} = \mu^{(1)}_{i:n}$.

A.2 Modélisation d'une poursuite à l'aide de statistiques d'ordres

Étant donné la nature gloutonne de MP, il est intéressant de modéliser la convergence de la série des résiduels $R^n x$. C'est ce que nous allons faire en utilisant les statistiques d'ordre. Soit $x \in \mathbb{R}^N$. Lors de la première itération de MP, les projections du résiduel $R^0 x = x$ sur un dictionnaire Φ complet composé de M atomes $\{\phi_i\}_{i\in[1,M]}$ $(M > N)$ de norme ℓ_2 unitaire sont données par :

$$\forall i \in [1..M], \alpha_i = \langle R^0 f, \phi_i \rangle \tag{A.2.1}$$

Notons alors $z_i = |\alpha_i|$ et considérons ces M éléments comme des échantillons i.i.d d'une variable aléatoire Z vivant dans $[0, \|x\|]$. Notons que l'hypothèse d'indépendance est en quelque sorte une

hypothèse de quasi-incohérence du dictionnaire. MP sélectionne l'atome $\phi_{\gamma_0} = \arg\max z_i$ avec un coefficient $\alpha_{\gamma_0} = \langle R^0 x, \phi_{\gamma_0} \rangle$ dont la valeur absolue correspond au maximum des $\{z_i\}$, c'est à dire la statistique d'ordre $Z_{M:M}$.

$$|\alpha_{\gamma_0}| = \max_{\phi_\gamma \in \Phi} |\langle R^0 x, \phi_\gamma \rangle| = Z_{M:M} \tag{A.2.2}$$

Un MP standard construit un nouveau résiduel $R^1 x$ en soustrayant la contribution de l'atome sélectionné et en itérant ce processus. Si l'on fait une hypothèse de quasi-incohérence entre les atomes, on peut dire qu'au bout de n itérations, la contribution des n atomes les plus importants a été soustraite. MP va donc sélectionner un nouvel atome, dont la valeur absolue de la contribution est donnée par l'élément le plus grand de la séquence $\{z_i\}$ sachant que l'on a retiré les n plus grand, ce qui correspond à la statistique d'ordre $Z_{M-n:M}$:

$$|\alpha_{\gamma_n}| = \max_{\phi_\gamma \in \Phi} |\langle R^n x, \phi_\gamma \rangle| = Z_{M-n:M} \tag{A.2.3}$$

La conservation de l'énergie permet alors d'écrire (A.2.4).

$$\|x\|^2 = \|R^n x\|^2 + \sum_{i=0}^{n-1} |\alpha_{\gamma_i}|^2 \tag{A.2.4}$$

En combinant (A.2.4) et (A.2.3) et en prenant l'espérance on obtient (A.2.5)

$$\mathbb{E}(\|R^n x\|^2) = \|x\|^2 - \sum_{i=0}^{n-1} \mu^{(2)}_{M-i:M} \tag{A.2.5}$$

où l'on reconnaît le moment d'ordre 2 $\mu^{(2)}_{M-i:M} = \sigma^2_{M-i:M} + \mu^2_{M-i:M}$. De la même façon, on peut dériver la variance :

$$Var(\|R^n x\|^2) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} cov(Z^2_{M-i:M}, Z^2_{M-j:M}) \tag{A.2.6}$$

Ce qui signifie que, sachant la distribution de Z, les équations A.2.5 et A.2.6 donnent des estimateurs de la moyenne et la variance de la norme des éléments de la séquence de résiduels construits par un MP standard sur un Φ complet et quasi-incohérent.

A.2.1 Nouveaux tirages : Poursuite dans une séquence de dictionnaires

L'idée au coeur de l'algorithme SSMP présenté en 4.2 page 58 équivaut à réaliser un nouveau tirage des projections z_i à chaque itération en forçant un changement de dictionnaire. Soit $\{\Phi_i\}_{i\in[0,n]}$ une séquence de dictionnaires quasi-incohérent de même taille M, faisons de plus l'hypothèse que les projections de x sur chaque dictionnaire suivent la même distribution.

Pour la première itération, le processus est identique à celui du MP standard et l'atome ϕ_{γ_0} dans Φ_0 est sélectionné par $\arg\max_{\phi_\gamma \in \Phi_0} |\langle R^0 f, \phi_\gamma \rangle|$ avec un coefficient identique (A.2.2). Après soustraction de cet atome, on obtient le résiduel $R^1 x$. SSMP cherche ensuite l'atome le plus corrélé dans un nouveau dictionnaire Φ_1. les valeurs absolues des projections dans Φ_1 sont M échantillons i.i.d z^1_i :

$$\forall i \in [1..M], z^1_i = |\alpha_i| = |\langle R^1 f, \phi_i \rangle| \tag{A.2.7}$$

Supposons maintenant que ces échantillons z^1_i suivent la même loi de distribution que le premier jeu d'échantillons z_i, à une différence d'échelle près car $R^1 x$ a une énergie plus petite que $R^0 x$. En

d'autres termes, on définit une nouvelle variable aléatoire $Z1$ distribuée comme $Z \times \frac{\|R^1 x\|}{\|x\|}$. On note alors $Z1_{M:M}$ sa statistique d'ordre M. Son moment d'ordre 2 est alors :

$$\mathbb{E}(Z1^2_{M:M}) = \mathbb{E}(Z^2_{M:M}).\mathbb{E}(\frac{\|R^1 x\|^2}{\|x\|^2}) + cov(Z^2_{M:M}, \frac{\|R^1 x\|^2}{\|x\|}) \tag{A.2.8}$$

or on sait que$\|R^1 x\|^2 = \|x\|^2 - Z^2_{M:M}$ d'où :

$$cov(Z^2_{M:M}, \|R^1 x\|^2) = -cov(Z^2_{M:M}, Z^2_{M:M}) = (\mu^{(2)}_{M:M})^2 - \mu^{(4)}_{M:M} \tag{A.2.9}$$

ce qui donne :

$$\begin{aligned} \mathbb{E}(Z1^2_{M:M}) &= \mu^{(2)}_{M:M}.(1 - \frac{\mu^{(2)}_{M:M}}{\|x\|^2}) + \frac{1}{\|x\|^2}\left((\mu^{(2)}_{M:M})^2 - \mu^{(4)}_{M:M}\right) \\ &= \mu^{(2)}_{M:M} - \frac{\mu^{(4)}_{M:M}}{\|x\|^2} \end{aligned}$$

A l'itération suivante, le nouveau résiduel $R^2 x$ est tel que $\|R^2 x\|^2 = \|R^1 x\|^2 - Z1^2_{M:M}$, soit par passage à l'espérance :

$$\mathbb{E}(\|R^2 x\|^2) = \|x\|^2 - 2\mu^{(2)}_{M:M} + \frac{\mu^{(4)}_{M:M}}{\|x\|^2} \tag{A.2.10}$$

et l'on voit apparaître des moments d'ordre supérieur de la statistique d'ordre M. Le fait de changer de dictionnaire garanti que l'on sélectionne le meilleur parmi un nouvel ensemble de M éléments ce qui justifie que seuls les moments de la statistique d'ordre M soient nécessaires à l'expression de $\mathbb{E}(\|R^2 x\|^2)$. Par récurrence, il est aisé de montrer qu'à l'itération n on obtient l'expression A.2.11) :

$$\mathbb{E}(\|R^n x\|^2) = \|x\|^2 + \sum_{i=1}^{n} (-1)^i \binom{n}{i} \frac{\mu^{(2i)}_{M:M}}{\|x\|^{2(i-1)}} \tag{A.2.11}$$

De la même façon on obtient l'expression de la variance :

$$Var(\|R^n x\|^2) = \sum_{i=1}^{n}\sum_{j=1}^{n} (-1)^{j+i} \binom{n}{i}\binom{n}{j} \frac{cov(Z^{(2i)}_{M:M}, Z^{(2j)}_{M:M})}{\|x\|^{2(i+j-1)}} \tag{A.2.12}$$

A.3 Simulations

On peut comparer les deux stratégies (projections fixes ou retirées à chaque itération) sur des cas de figure synthétiques ou la distribution originale f^Z est connue.

On calcule ainsi l'erreur relative $\epsilon(n) = 10 \log \frac{\|R^n x\|}{\|x\|^2}$ après n itérations pour chacune des stratégies. les équations A.2.5 à A.2.12 fournissent une expression analytique de la moyenne et la variance de $\epsilon(n)$ pour chacune des stratégies. Par exemple, si Z est distribué uniformément $Z \sim \mathcal{U}(0,1)$, ses statistiques d'ordres suivent une loi de distribution Gamma : $Z_{k:M} \sim Beta(k, M+1-k)$ et tous les moments $\mu^{(m)}_{M-k:M}$ sont aisément déductibles. Pour des distributions plus complexes, nous proposons plus bas le calcul. Pour l'instant, on utilise une simulation grossière de poursuite gloutonne respectant les hypothèses faites plus haut à savoir :

- Dans le cas fixe : la sélection d'un atome n'influe pas sur la valeur des autres projections
- Dans le cas variable : à chaque itération la séquence est retirée selon la même loi avec une variance réduite.

La Figure A.3.1 illustre différents cas de figures. Pour 3 distributions de f^Z – uniforme, demi-normale et exponentielle on affiche les distributions $f^Z_{M:M}$ et $f^Z_{M/2:M}$. ces deux éléments combinés donnent une idée de la rapidité avec laquelle les coefficients sélectionnés dans le cas fixe décroissent. Au vu des résultats on peut faire les remarques suivantes :

- Dans le cas uniforme, les coefficients décroissent relativement lentement, au bout de $M/2$ itérations dans le cas fixe, on peut encore espérer sélectionner un élément assez grand (*c-à-d.* un *bon* atome). Dans ce cas de figure, retirer les projections (*c-à-d.* changer de dictionnaire) semble même être une mauvaise idée, comme le montre le profil dégradé de convergence de l'erreur.
- En revanche, si les projections sont distribuées selon une demi-gaussienne ou exponentiellement, la stratégie de changement de dictionnaire est sensiblement meilleure que la stratégie fixe. La différence est d'autant plus importante que la queue de la distribution est longue, ce qui est une conséquence assez naturelle de l'allure des équations A.2.5 et A.2.12.

Ces observations sont prometteuses, car en pratique le modèle uniforme ne correspond pas vraiment au profil des projections d'un signal dans un dictionnaire redondant. En revanche le modèle exponentiel est régulièrement associé à la notion de parcimonie d'une représentation. Plus généralement, les distributions à longue queue apparaissent souvent comme dans les problèmes pratiques ou l'hypothèse de parcimonie est avancée.

Dans le pire des cas, le modèle demi-gaussien va lui être préféré pour modéliser un signal mal corrélé avec un dictionnaire, c'est à dire pour lequel l'hypothèse de parcimonie est caduque. Dans ce cas là aussi, changer de dictionnaire semble être une bonne stratégie pour accélérer la convergence de l'erreur quadratique.

Pour être complet, rappelons qu'il s'agit d'une modélisation simpliste du comportement des algorithmes de poursuites, et non d'une preuve formelle que la stratégie proposée accélère la convergence. Néanmoins il s'agit à notre connaissance d'une modélisation originale. Et les chapitres 4 et 5 montrent que ces considérations sont valables en pratique sur des signaux et dictionnaires réels.

A.4 Calcul de moments

Dans cette section nous présentons les calculs des deux premiers moments des statistiques d'ordre, en particulier ceux du maximum, pour deux distributions particulières : la loi exponentielle et la loi demi-normale.

A.4.1 Distribution exponentielle

Une distribution exponentielle sur f^Z est un modèle souvent associé (pas toujours avec raison, voir [GCD12]) aux vecteurs parcimonieux et/ou compressibles. En effet pour ce type de signaux, les projections sont distribuées selon une loi de Laplace. Z suit donc une loi exponentielle. On pose donc :

$$f^Z(z) = \frac{1}{\beta} e^{-\beta z}$$

pour $z \geq 0$ et donc :

$$F^Z(z) = (1 - e^{-\beta z})$$

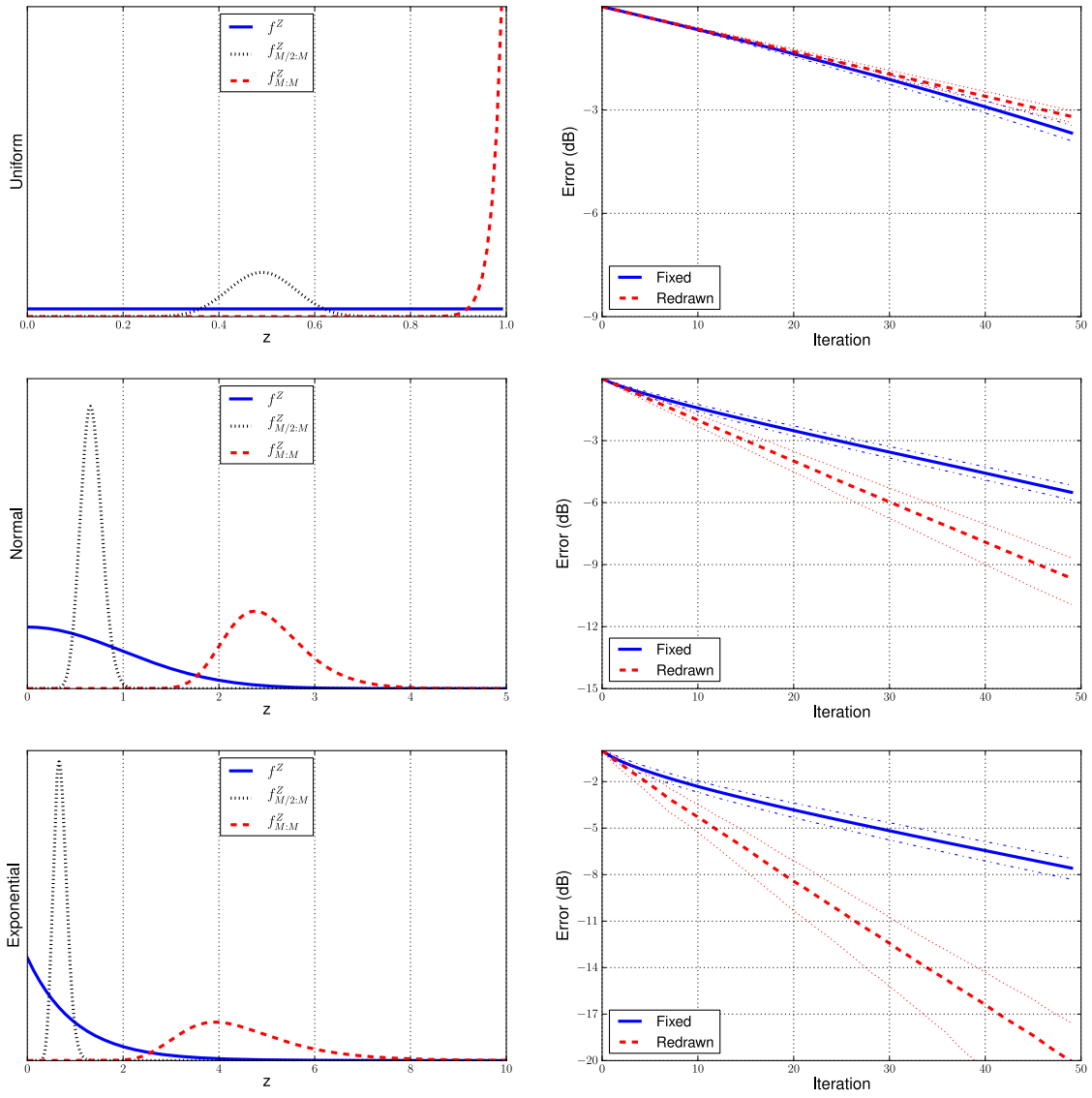

FIGURE A.3.1: Gauche : densité de probabilité de Z, de son maximum (statistique d'ordre M) et de sa statistique d'ordre $M/2$. Droite : erreur relative (moyenne et variance) fonction du nombre d'itérations. De haut en bas : modèle uniforme, normal ($\sigma = 1$) et exponentiel ($\mu = 1$). $M = 100$ résultats moyens sur 1000 simulations.

le moment d'ordre 1 de la statistique d'ordre i est donc :

$$\begin{aligned}
\mu_{i:n} &= \int_0^{+\infty} i\binom{n}{n-i}\left(1-e^{-\beta z}\right)^{i-1}\frac{z}{\beta}e^{-\beta z}\left(e^{-\beta z}\right)^{n-i}dz \\
&= \frac{i}{\beta}\binom{n}{n-i}\int_0^{+\infty} ze^{-\beta z(n-i+1)}\left(1-e^{-\beta z}\right)^{i-1}dz \\
&= \frac{i}{\beta}\binom{n}{n-i}\int_0^{+\infty} ze^{-\beta z(n-i+1)}\sum_{k=0}^{i-1}(-e^{-\beta z})^k\binom{i-1}{k}dz \\
&= \frac{i}{\beta}\binom{n}{n-i}\sum_{k=0}^{i-1}\binom{i-1}{k}\int_0^{+\infty} ze^{-\beta z(n-i+1)}(-e^{-\beta z})^k dz \\
&= \frac{i}{\beta}\binom{n}{n-i}\sum_{k=0}^{i-1}\binom{i-1}{k}(-1)^k\int_0^{+\infty} ze^{-\beta z(n-i+1+k)}dz
\end{aligned}$$

or :

$$\int_0^{+\infty} ze^{-\beta z(n-i+1+k)}dz = \frac{1}{\left(\beta(n-i+1+k)\right)^2}$$

d'où :

$$\mu_{i:n} = \frac{i}{\beta}\binom{n}{n-i}\sum_{k=0}^{i-1}\binom{i-1}{k}(-1)^k\frac{1}{\left(\beta(n-i+1+k)\right)^2}$$

d'où on en déduit la valeur particulière :

$$\mu_{n:n} = \frac{n}{\beta}\sum_{k=0}^{n-1}\binom{n-1}{k}\frac{(-1)^k}{\left(\beta(k+1)\right)^2}$$

on peut montrer en utilisant :

$$\int_0^{+\infty} z^2e^{-\beta z}dz = \frac{2}{(\beta)^3}$$

que pour le moment d'ordre 2 :

$$\mu^{(2)}_{n:n} = \frac{2n}{\beta}\sum_{k=0}^{n-1}\binom{n-1}{k}\frac{(-1)^k}{\left(\beta(k+1)\right)^3}$$

A.4.2 Distribution demi-normale

Une distribution demi-normale sur f^Z est un bon modèle pour des signaux de bruit. En particulier, c'est à rapprocher de la borne $\Lambda_W(\Phi)$

$$f^Z(z,\sigma) = \frac{\sqrt{2}}{\sigma\sqrt{\pi}}e^{-\frac{z^2}{2\sigma^2}}$$

pour $z \geq 0$. La fonction de répartition :

$$F^Z(z,\sigma) = \mathrm{erf}\left(\frac{z}{\sigma\sqrt{2}}\right)$$

Passons directement au calcul du maximum :

$$\mu_{n:n} = \int_0^{+\infty} z\frac{n}{\sigma}\sqrt{\frac{2}{\pi}}\left[\mathrm{erf}\left(\frac{z}{\sigma\sqrt{2}}\right)\right]^{n-1}\exp\left(\frac{-z^2}{2\sigma^2}\right)dz$$

on doit alors remarquer que :

$$\frac{\partial\left(\mathrm{erf}\left(\frac{z}{\sigma\sqrt{2}}\right)^n\right)}{\partial z} = \frac{n}{\sigma}\sqrt{\frac{2}{\pi}}\left[\mathrm{erf}\left(\frac{z}{\sigma\sqrt{2}}\right)\right]^{n-1}\exp\left(\frac{-z^2}{2\sigma^2}\right)$$

du coup on peut écrire :

$$\mu_{n:n} = \int_0^{+\infty} z\frac{\partial\left(\mathrm{erf}\left(\frac{z}{\sigma\sqrt{2}}\right)^n\right)}{\partial z}dz$$

Cette intégrale est calculable (on peut d'ailleurs aisément vérifier les valeur pour $\mu_{1:1}$et $\mu_{2:2}$) mais le calcul devient très vite fastidieux, et aucune relation simple de récurrence n'a pu être trouvée. En revanche, on observe expérimentalement (voir par exemple la Figure 5.1.5 page 83) que la distribution de $f_{n:n}$ est uni-modale et que la médiane est assez proche de la moyenne.

Calcul de la médiane On trouve assez simplement la distribution cumulative :

$$\begin{aligned} F^Z_{n:n}(z) &= \int_0^z f^z_{n:n}(x)dx \\ &= \int_0^z \frac{d\left(\mathrm{erf}\left(\frac{x}{\sigma\sqrt{2}}\right)^n\right)}{dx}dx \\ &= \mathrm{erf}\left(\frac{z}{\sigma\sqrt{2}}\right)^n \end{aligned}$$

pour trouver la médiane $\nu_{n:n}$, on cherche à résoudre :

$$F^Z_{n:n}(\nu_{n:n}) = 0.5$$

d'où :

$$\nu_{n:n} = \sigma\sqrt{2}\mathrm{erf}^{-1}\left(0.5^{\frac{1}{n}}\right)$$

Le lecteur intéressé pourra voir que quelle que soit la distribution de Z, $f_{n:n}$suit une loi dite de valeur extrême généralisée qui regroupe trois type de distribution (Weibull, Gumbel et Fréchet). En théorie il est donc possible de proposer un modèle général pour le calcul des moments de $Z_{n:n}$. Une discussion se trouve dans la littérature [CHT09, HCT09] et se retrouve notamment pour la modélisation de risque en finance [HB00], mais sort très largement du cadre de ce travail.

Ces valeurs constituent la base de la modélisation du comportement dynamique des algorithmes gloutons proposée au Chapitre 5.

Bibliographie

[ACR99] C.Allauzen, M.Crochemore et M.Raffinot : Factor Oracle : a new approach for pattern matching. Rapport technique, Institut Gaspard-Monge, Universite de Marne-la-Vallee, 1999.

[AEJ+12] A.Adler, V.Emiya, M. G.Jafari, M.Elad, R.Gribonval et M. D.Plumbley : Audio Inpainting. *IEEE Transactions on Audio, Speech and Language Processing*, 20(3):922 – 932, 2012.

[AeqGC06] A.Abu-el quran, R.Goubran et A.C. Chan : Adaptive Feature Selection for Speech / Music Classification. *In IEEE Workshop on Multimedia Signal Processing*, pages 212–216, octobre 2006.

[AF95] F.Auger et P.Flandrin : Improving the readability of time-frequency and time-scale representations by the reassignment method. *IEEE Transactions on Signal Processing*, 43(5):1068–1089, mai 1995.

[AFLM06] G.Aloupis, T.Fevens, S.Langerman et T.Matsui : Algorithms for computing geometric measures of melodic similarity. *Computer Music*, pages 1–17, 2006.

[AHH+01] E.Allamanche, J.Herre, O.Hellmuth, B.Froba, T.Kastner et M.Cremer : Content-based Identification of Audio Material Using MPEG-7 Low Level Description. *In International Society for Music Information Retrieval Conference*, Bloomington, Indiana, USA, octobre 2001.

[AM11] J.Andén et S.Mallat : Multiscale scattering for audio classification. *In International Society for Music Information Retrieval Conference*, pages 657–662, 2011.

[ASCA12] I.Ari, U.Simsekli, A.Cemgil et L.Akarun : Large Scale Polyphonic Music Transcription Using Randomized Matrix Decompositions. *In European Signal Processing Conference*, pages 2020–2024, 2012.

[BBG06] L.Benaroya, F.Bimbot et R.Gribonval : Audio source separation with a single sensor. *IEEE Transactions on Audio, Speech and Language Processing*, 14(1):191–199, janvier 2006.

[BCTL07] L.Barrington, A.Chan, D.Turnbull et G.Lanckriet : Audio Information Retrieval using Semantic Similarity. *In 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, pages 725—-728. IEEE, 2007.

[BD06] T.Blumensath et M.Davies : Sparse and shift-Invariant representations of music. *IEEE Transactions on Audio, Speech and Language Processing*, 14(1):50–57, janvier 2006.

[BD08] T.Blumensath et M.Davies : Gradient Pursuits. *IEEE Transactions on Signal Processing*, 56(6):2370–2382, juin 2008.

[BD09] T.Blumensath et M.Davies : Stagewise Weak Gradient Pursuits. *IEEE Transactions on Signal Processing*, 57(11):4333–4346, novembre 2009.

[BDA+05] J.Bello, L.Daudet, S.Abdallah, C.Duxbury, M.Davies et M.Sandler : A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, 13(5):1035–1047, septembre 2005.

[BDDW08] R.Baraniuk, M.Davenport, R.DeVore et M.Wakin : A Simple Proof of the Restricted Isometry Property for Random Matrices. *Constructive Approximation*, 28(3):253–263, janvier 2008.

[BDW+05] D.Baron, M.Duarte, M.Wakin, S.Sarvotham et R.Baraniuk : Distributed Compressive Sensing. Rapport technique, janvier 2005.

[Bel11] J. P.Bello : Measuring Structural Similarity in Music. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2013–2025, septembre 2011.

[Ber09] N.Bertin : *Les factorisations en matrices non-négatives. Approches contraintes et probabilistes, application à la transcription automatique de musique polyphonique.* Thèse de doctorat, Institut Telecom - Telecom ParisTech, 2009.

[BF01] R.Baraniuk et P.Flandrin : Measuring Time – Frequency Information Content Using the Rényi Entropies. *IEEE Transactions on Information Theory*, 47(4):1391–1409, 2001.

[BJM11] F.Bach, R.Jenatton et J.Mairal : Convex optimization with sparsity-inducing norms. Rapport technique, INRIA, 2011.

[BL98] C.Bennett et M.Li : Information distance. *IEEE Transactions on Information Theory*, 44(4):1407–1423, 1998.

[BL11] J. J.Burred et P.Leveau : Geometric multichannel common signal separation with application to music and effects extraction from film soundtracks. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 201–204, 2011.

[BM95] F.Bergeaud et S.Mallat : Matching pursuit of images. *In IEEE International Conference on Image Processing*, volume 1, pages 53–56. IEEE Comput. Soc. Press, 1995.

[BMEWL11] T.Bertin-Mahieux, D.Ellis, B.Whitman et P.Lamere : The million song dataset. *In International Society for Music Information Retrieval Conference*, 2011.

[Bro91] J. C.Brown : Calculation of a constant Q spectral transform. *Journal of the Acoustical Society of America*, 89(1):425–434, 1991.

[BT09] A.Beck et M.Teboulle : A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, janvier 2009.

[Buh01] J.Buhler : Efficient large-scale sequence comparison by locality-sensitive hashing. *Bioinformatics*, 17(5):419–428, mai 2001.

[BYD07] T.Blumensath, M.Yaghoobi et M. E.Davies : Iterative Hard Thresholding and L0 Regularisation. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 877—-880, 2007.

[CDS98] S.Chen, D.Donoho et M.Saunders : Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1998.

[CE09] C.Cotton et D. P. W.Ellis : Finding similar acoustic events using matching pursuit and locality-sensitive hashing. *In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 125–128, 2009.

[CE10] C. V.Cotton et D. P. W.Ellis : Audio fingerprinting to identify multiple videos of an event. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2386–2389, 2010.

[Cev08] V.Cevher : Learning with Compressible Priors. *In Neural Information Processing Systems Conference*, 2008.

[CHT09] D. A.Clifton, S.Hugueny et L.Tarassenko : Novelty detection with multivariate Extreme Value Theory, part I : A numerical approach to multimodal estimation. *In IEEE International Workshop on Machine Learning for Signal Processing*, numéro x, pages 1–6, septembre 2009.

[CJ07] M. G.Christensen et S. H.Jensen : The Cyclic Matching Pursuit and its Application to Audio Modeling and Coding. *Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers*, pages 550–554, novembre 2007.

[CK80] I.Csiszár et J.Körner : Towards a general theory of source networks. *IEEE Transactions on Information Theory*, 26(2):155–165, 1980.

[CL97] A.Chambolle et P.Lions : Image recovery via total variation minimization and related problems. *Numerische Mathematik*, pages 167–188, 1997.

[CLD11] G.Chardon, A.Leblanc et L.Daudet : Plate impulse response spatial interpolation with sub-Nyquist sampling. *Journal of Sound and Vibration*, 330(23):5678–5689, novembre 2011.

[CLM09] E.Candes, X.Li et Y.Ma : Robust principal component analysis ? *Journal of the ACM*, 58(3):1–37, 2009.

[Con08] A.Cont : *Modeling Musical Anticipation*. Thèse de doctorat, University of Paris 6 (UPMC), and University of California San Diego (UCSD), 2008.

[CRE06] M.Chabin, J.Rietsch et C.Eric : *Dématérialisation et archivage électronique*. Dunod, 2006.

[CRT06] E.Candes, J.Romberg et T.Tao : Robust uncertainty principles : exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, février 2006.

[CW92] R.Coifman et M.Wickerhauser : Entropy-based algorithms for best basis selection. *IEEE Transactions on Information Theory*, 38(2):713–718, mars 1992.

[CW05] P. L.Combettes et V. R.Wajs : Signal Recovery by Proximal Forward-Backward Splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, janvier 2005.

[Dau06] L.Daudet : Sparse and structured decompositions of signals with the molecular matching pursuit. *IEEE Transactions on Audio, Speech and Language Processing*, 14(5):1808–1816, septembre 2006.

[DB95] P. J.Durka et K. J.Blinowska : Analysis of EEG transients by means of matching pursuit. *Annals of biomedical engineering*, 23(5):608–11, 1995.

[DDFG10] I.Daubechies, R.DeVore, M.Fornasier et C. S.Güntürk : Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1):1–38, janvier 2010.

[DE10] A.Divekar et O.Ersoy : Probabilistic Matching Pursuit for Compressive Sensing. *Technical Report, Purdue University, West Lafayette*, 2010.

[DET06] D.Donoho, M.Elad et V.Temlyakov : Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1):6–18, janvier 2006.

[DHD12] A.Dremeau, C.Herzet et L.Daudet : Boltzmann Machine and Mean-Field Approximation for Structured Sparse Decompositions. *IEEE Transactions on Signal Processing*, 60(7):3425–3438, juillet 2012.

[DHS96] R.Duda, P.Hart et D.Storck : *Pattern Classification*. wiley & sons, 2nd édition, 1996.

[DIB01] P.Durka, D.Ircha et K.Blinowska : Stochastic time-frequency dictionaries for matching pursuit. *IEEE Transactions on Signal Processing*, 49(3):507–510, mars 2001.

[DIFM10] E.Didiot, I.Illina, D.Fohr et O.Mella : A wavelet-based parameterization for speech/music discrimination. *Comput. Speech Lang.*, 24(2):341–357, 2010.

[Dix94] R.Dixon : *Spread Spectrum Systems*. John Wiley & Sons, 1994.

[DM09] W.Dai et O.Milenkovic : Subspace Pursuit for Compressive Sensing Signal Reconstruction. *IEEE Transactions on Information Theory*, 55(5):2230–2249, mai 2009.

[DMM09] D. L.Donoho, A.Maleki et A.Montanari : Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences of the United States of America*, 106(45):18914–9, novembre 2009.

[DMR11] P.Dymarski, N.Moreau et G.Richard : Greedy sparse decompositions : A comparative study. *Journal on Advances in Signal Processing*, 2011.

[DMV90] P.Dymarski, N.Moreau et A.Vigier : Optimal and sub-optimal algorithms for selecting the excitation in linear predictive coders. *In IEEE International Conference on Acoustics Speech and Signal Processing*, pages 485–488, 1990.

[DMZ94] G.Davis, S.Mallat et Z.Zhang : Adaptive time-frequency decompositions. *Optical Engineering*, 33(7):2183, 1994.

[Don06] D.Donoho : Compressed sensing. *IEEE Transactions on Information Theory*, 52(4): 1289–1306, avril 2006.

[DT10] D. L.Donoho et J.Tanner : Precise Undersampling Theorems. *Proceedings of the IEEE*, 98(6):913–924, juin 2010.

[DTDS06] D. L.Donoho, Y.Tsaig, I.Drori et J.-l.Starck : Sparse Solution of Underdetermined Linear Equations by Stagewise Orthogonal Matching Pursuit. Rapport technique March, 2006.

[EC03] C.Etemoglu et V.Cuperman : Matching pursuits sinusoidal speech coding. *IEEE Transactions on Speech and Audio Processing*, 11(5):413–424, septembre 2003.

[EGK11] I.Elfitri, B.Günel et A. M.Kondoz : Multichannel Audio Coding Based on Analysis by Synthesis. *Proceedings of the IEEE*, 99(4):657–670, avril 2011.

[EMKPK00] K.El-Maleh, M.Klein, G.Petrucci et P.Kabal : Speech/music discrimination for multimedia applications. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages 2445–2448, 2000.

[Ess05] S.Essid : *Classifictaion automatique des signaux audio-fréquences : reconnaissance des instruments de musiques.* Thèse de doctorat, ENST, 2005.

[EWJL10] D.Ellis, B.Whitman, T.Jehan et P.Lamere : The echo nest musical fingerprint. *In International Society for Music Information Retrieval Conference*, volume 32, 2010.

[EY09] M.Elad et I.Yavneh : A Plurality of Sparse Representations Is Better Than the Sparsest One Alone. *IEEE Transactions on Information Theory*, 55(10):4701–4714, octobre 2009.

[FBD+96] L. D.Fielder, M.Bosi, G.Davidson, M.Davis, C.Todd et S.Vernon : AC-2 and AC-3 : Low-Complexity Transform-Based Audio Coding. *In Collected Papers on Digital Audio Bit-Rate Reduction*, 1996.

[FBD09] C.Févotte, N.Bertin et J.-L.Durrieu : Nonnegative matrix factorization with the Itakura-Saito divergence : with application to music analysis. *Neural computation*, 21(3):793–830, mars 2009.

[FBR12] B.Fuentes, R.Badeau et G.Richard : Blind Harmonic Adaptative Decomposition applied to Suprevised Source Separation. *In European Signal Processing Conference*, numéro 1, pages 2654 – 2658, 2012.

[FDBB00] S. E.Ferrando, E. J.Doolittle, A.Bernal et L.Bernal : Probabilistic matching pursuit with Gabor dictionaries. *Signal Processing*, 80(10):2099–2120, octobre 2000.

[FELR11] R.Foucard, S.Essid, M.Lagrange et G.Richard : Multi-scale temporal fusion by boosting for music classification. *In International Society for Music Information Retrieval Conference*, 2011.

[FG05] C.Fevotte et S.Godsill : A Bayesian approach to time-frequency based blind source separation. *In Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, numéro 2, 2005.

[FG06] C.Févotte et S.Godsill : Blind separation of sparse sources using Jeffrey's inverse prior and the EM algorithm. *In International Conference on Independent Component Analysis*, pages 593–600, 2006.

[FG10] D.FitzGerald et M.Gainza : Single channel vocal separation using median filtering and factorisation techniques. *ISAST Transactions on Electronic and Signal Processing*, 4(1):62–73, 2010.

[FLBR12] B.Fuentes, A.Liutkus, R.Badeau et G.Richard : Probabilistic Model for Main Melody Extraction Using Constant-Q Transform. *In IEEE International Conference on Acoustics Speech and Signal Processing*, pages 5357 – 5360, 2012.

[FMG+12] S.Fenet, M.Moussallam, Y.Grenier, L.Daudet et G.Richard : A Framework for Fingerprint-Based detection of Repeating Objects in Multimedia Streams. *In European Signal Processing Conference*, pages 1464–1468, 2012.

[FP12] T.Fillon et J.Prado : A Flexible Multi-Resolution Time-Frequency Analysis Framework for Audio Signal. *In 11th International Conference on Information Science, signal processing and their applications (ISSPA)*, numéro 2, 2012.

[FRG11] S.Fenet, G.Richard et Y.Grenier : A Scalable Audio Fingerprint Method with Robustness to Pitch-Shifting. *In International Society for Music Information Retrieval Conference*, pages 121–126, 2011.

[FTDG08] C.Févotte, B.Torrésani, L.Daudet et S. J.Godsill : Sparse Linear Regression With Structured Priors and Application to Denoising of Musical Audio. 16(1):174–185, 2008.

[Fuc11] J.-J.Fuchs : Spread representations. *In 2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 814–817. IEEE, novembre 2011.

[FV01] P.Frossard et P.Vandergheynst : Redundancy in non-orthogonal transforms. *In IEEE International Symposium on Information Theory*, 2001.

[FVF06] R. M.Figueras i Ventura, P.Vandergheynst et P.Frossard : Low-rate and flexible image coding with redundant representations. *IEEE Transactions on Image Processing*, 15(3):726–739, mars 2006.

[FVFK04] P.Frossard, P.Vandergheynst, R. M.Figueras et M.Kunt : A posteriori quantization of progressive matching pursuit streams. *IEEE Transactions on Signal Processing*, 52(2):525–535, 2004.

[GaARRM05] B.Girod, a.M. Aaron, S.Rane et D.Rebollo-Monedero : Distributed Video Coding. *Proceedings of the IEEE*, 93(1):71–83, janvier 2005.

[GB03] R.Gribonval et E.Bacry : Harmonic Decomposition of Audio Signals with Matching Pursuit. 51(1):101–111, janvier 2003.

[GBM+96] R.Gribonval, E.Bacry, S.Mallat, P.Depalle et X.Rodet : Analysis of sound signals with high resolution matching pursuit. *In Proceedings of Third International Symposium on Time-Frequency and Time-Scale Analysis (TFTS-96)*, pages 125–128. IEEE, juin 1996.

[GCD12] R.Gribonval, V.Cevher et M. E.Davies : Compressible Distributions for High-Dimensional Statistics. *IEEE Transactions on Information Theory*, 58(8):5016–5034, août 2012.

[GE12] R.Giryes et M.Elad : Cosamp and SP for the Co-sparse Analysis Model. *In European Signal Processing Conference*, pages 964–968, 2012.

[GN01] R.Gribonval et M.Nielsen : Approximate weak greedy algorithms. *Advances in Computational Mathematics*, 14(4):361–378, 2001.

[GNP72] S. S.Gupta, K.Nagel et S.Panchapakesan : On the order statistics from equally correlated normal random variables. *Biometrika*, 60:403–413, 1972.

[Got03] M.Goto : A chorus-section detecting method for musical audio signals. *In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, volume 2003, page 50, 2003.

[GR97] I.Gorodnitsky et B.Rao : Sparse signal reconstruction from limited data using FOCUSS : a re-weighted minimum norm algorithm. *IEEE Transactions on Signal Processing*, 45(3):600–616, mars 1997.

[Gri01] R.Gribonval : Fast matching pursuit with a multiscale dictionary of Gaussian chirps. *IEEE Transactions on Signal Processing*, 49(5):994–1001, mai 2001.

[Gri02] R.Gribonval : Sparse decomposition of stereo signals with Matching Pursuit and application to blind separation of more than two sources from a stereo mixture. *In IEEE International Conference on Acoustics Speech and Signal Processing*, pages 3057—-3060, 2002.

[GRS07] R.Gribonval, H.Rauhut et K.Schnass : Atoms of all channels, unite! Rapport technique, INRIA, 2007.

[Gun10] A.Gunawan : Classification of Fast Magnetic Resonance Image Reconstruction Using Matching Pursuit Family Algorithm. 2010.

[GV99] M. M.Goodwin et M.Vetterli : Matching pursuit and atomic signal models based on recursive filter banks. *IEEE Transactions on Signal Processing*, 47:1890–1902, 1999.

[GV06] R.Gribonval et P.Vandergheynst : On the exponential convergence of matching pursuits in quasi-incoherent dictionaries. *IEEE Transactions on Information Theory*, 52(1):255–261, janvier 2006.

[HA06] K.Huang et S.Aviyente : Sparse Representation for Signal Classification. *In Advances in neural computations*, 2006.

[Haz12] B.Hazra : Independent Component Analysis Using Maximization of L-Kurtosis. *International Journal of Statistics and Applications*, 2(2):1–10, 2012.

[HB00] F.Herzberg et C.Bennemann : *Order Statistics for Value at Risk Estimation and Option Pricing.* 2000.

[HBD11a] R.Hennequin, R.Badeau et B.David : NMF with time-frequency activations to model non stationary audio events. *IEEE Transactions on Audio, Speech and Language Processing*, 19(4):744–753, 2011.

[HBD11b] R.Hennequin, R.Badeau et B.David : Scale-invariant probabilistic latent component analysis. *In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 129–132, 2011.

[HCF+06] J.-P.Haton, C.Cerisara, D.Fohr, Y.Laprie et K.Smaïli : *Reconnaissance automatique de la parole*. Dunod, 2006.

[HCSHJ12] P.-S.Huang, S. D.Chen, P.Smaragdis et M.Hasegawa-Johnson : Singing-voice separation from monaural recordings using robust principal component analysis. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 57–60, 2012.

[HCT09] S.Hugueny, D. A.Cliftony et L.Tarassenko : Novelty detection with multivariate Extreme Value Theory, part II : An analytical approach to unimodal estimation. *In IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6. IEEE, septembre 2009.

[Her06] C.Herley : ARGOS : automatically extracting repeating objects from multimedia streams. *IEEE Transactions on Multimedia*, 8(1):115–129, 2006.

[HG97] G.Hinton et Z.Ghahramani : Generative Models for Discovering Sparse Distributed Representations. *Philosphical Transactions of the Royal Society*, B:1–25, 1997.

[HGT06] K.Herrity, A.Gilbert et J.Tropp : Sparse Approximation Via Iterative Thresholding. *In IEEE International Conference on Acoustics Speech and Signal Processing*, volume 3, pages 624—-627, 2006.

[HIK12] H.Hassanieh, P.Indyk et D.Katabi : Simple and Practical Algorithm for Sparse Fourier Transform. *In Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1183–1194, 2012.

[HKO01] J.Haitsma, T.Kalker et J.Oostveen : Robust Audio Hashing for Content Identification. *In International Workshop on Content-Based Multimedia Indexing*, 2001.

[HMT09] N.Halko, P.-G.Martinsson et J. A.Tropp : Finding structure with randomness : Probabilistic algorithms for constructing approximate matrix decompositions. pages 1–74, septembre 2009.

[Hos90] J.Hosking : L-moments : analysis and estimation of distributions using linear combinations of order statistics. *Journal of the Royal Statistical Society. Series B (*, 1990.

[HR08] N.Hurley et S.Rickard : Comparing measures of sparsity. *IEEE Workshop on Machine Learning for Signal Processing*, pages 55–60, 2008.

[Hub85] P.Huber : Projection pursuit. *The annals of Statistics*, 13(2):435–475, 1985.

[HW07] A. B.Hillel et D.Weinshall : Learning distance function by coding similarity. *In International Conference on Machine Learning*, pages 65–72, New York, New York, USA, 2007. ACM Press.

[IM98] P.Indyk et R.Motwani : Approximate nearest neighbors. *In Proceedings of the thirtieth annual ACM symposium on Theory of computing - STOC '98*, pages 604–613, New York, New York, USA, 1998. ACM Press.

[Iwe10] M. A.Iwen : Combinatorial Sublinear-Time Fourier Algorithms. *Foundations of Computational Mathematics*, 10(3):303–338, janvier 2010.

[JAB09] R.Jenatton, J.Audibert et F.Bach : Active set algorithm for structured sparsity-inducing norms. *In Neural Information Processing Systems Conference*, 2009.

[JE09] C.Joder et S.Essid : Temporal integration for audio classification with application to musical instrument classification. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(1):174–186, 2009.

[Jeh05] T.Jehan : *Creating music by listening.* Thèse de doctorat, Massachussets Institute of Technology, 2005.

[JER08] C.Joder, S.Essid et G.Richard : Alignment kernels for audio classification with application to music instrument recognition. *European Signal Processing Conference*, 2008.

[JFF11] H.Jégou, T.Furon et J.-J.Fuchs : Anti-sparse coding for approximate nearest neighbor search. (October), octobre 2011.

[JKMW98] S.Jaggi, W. C.Karl, S.Mallat et A. S.Willsky : High resolution pursuit for feature extraction. *Applied and Computational Harmonic Analysis*, 5:428–449, 1998.

[JLY08] Y.Jiao, M.Li et B.Yang : Compressed domain robust hashing for AAC audio. *In IEEE International Conference on Multimedia and Expo*, pages 1545–1548, 2008.

[JM10] R.Jenatton et J.Mairal : Proximal methods for sparse hierarchical dictionary learning. *In International Conference on Machine Learning*, 2010.

[JV08] P.Jost et P.Vandergheynst : On finding approximate nearest neighbours in a set of compressible signals. *In European Signal Processing Conference*, 2008.

[JVF06] P.Jost, P.Vandergheynst et P.Frossard : Tree-Based Pursuit : Algorithm and Properties. *IEEE Transactions on Signal Processing*, 54(12):4685–4697, décembre 2006.

[KG06] S.Krstulovic et R.Gribonval : MPTK : Matching Pursuit Made Tractable. *In IEEE International Conference on Acoustics Speech and Signal Processing*, volume 3, pages 496–499, 2006.

[KMS+12] F.Krzakala, M.Mézard, F.Sausset, Y.Sun et L.Zdeborová : Statistical-Physics-Based Reconstruction in Compressed Sensing. *Physical Review X*, 2(2):1–18, mai 2012.

[KT08] M.Kowalski et B.Torrésani : Random models for sparse signals expansion on unions of bases with application to audio signals. *IEEE Transactions on Signal Processing*, 8:3468–3481, 2008.

[KVG10] M.Kowalski, E.Vincent et R.Gribonval : Beyond the Narrowband Approximation : Wideband Convex Methods for Under-Determined Reverberant Audio Source Separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(7):1818–1829, septembre 2010.

[LBF11] A.Lefevre, F.Bach et C.Févotte : Itakura-Saito nonnegative matrix factorization with group sparsity. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 21 – 24, 2011.

[LBR11] A.Liutkus, R.Badeau et G.Richard : Gaussian Processes for Underdetermined Source Separation. *IEEE Transactions on Signal Processing*, 59(7):3155–3167, juillet 2011.

[LCY+09] Y.Liu, K.Cho, H. S.Yun, J. W.Shin et N. S.Kim : DCT based multiple hashing technique for robust audio fingerprinting. *In IEEE International Conference on Acoustics Speech and Signal Processing*, pages 61–64, 2009.

[LDCR07] P.Leveau, L.Daudet, G.Cornuz et E.Ravelli : Object Coding of Harmonic Sounds using Sparse and structured representations. *In Proceedings of the 10th International Conference on Digital Audio Effects*, pages 41–46, 2007.

[LdSD10] L.Lovisolo, E.da Silva et P. S.Diniz : On the statistics of matching pursuit angles. *Signal Processing*, 90(12):3164–3184, décembre 2010.

[Lev07] P.Leveau : *Décompositions parcimonieuses structurées : application à la représentation objet de la musique*. Thèse de doctorat, EDITE, 2007.

[LKCC07] H.Lee, Y.-D.Kim, A.Cichocki et S.Choi : Nonnegative tensor factorization for continuous EEG classification. *International journal of neural systems*, 17(4):305–17, août 2007.

[LL10] A.Liutkus et P.Leveau : Separation of music+ effects sound track from several international versions of the same movie. *In 128th AES Convention*, 2010.

[LLA01] A.Lefebvre, T.Lecroq et J.Alexandre : Compror : On-line lossless data compression with a factor oracle. *Information Processing Letters*, 2001.

[Lou90] J. G.Lourens : Detection and Logging Advertisements using its Sound. *In COMSIG 90, IEEE Symposium on Communications and Signal Processing*, pages 209–212, Johannesburg, South Africa, juin 1990.

[LOX06] H.Lin, Z.Ou et X.Xiao : Generalized Time-Series Active Search With Kullback-Leibler Distance for Audio Fingerprinting. *IEEE*, 13(8), août 2006.

[LRB+12] A.Liutkus, Z.Rafii, R.Badeau, B.Pardo et G.Richard : Adaptive filtering for music/voice separation exploiting the repeating musical structure. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 53–56, 2012.

[LS99] M.Lewicki et T.Sejnowski : Coding Time-Varying Signals Using Sparse, Shift-Invariant Representations. *Neural Information Processing Systems Conference*, 1999.

[LS00] M. S.Lewicki et T. J.Sejnowski : Learning overcomplete representations. *Neural computation*, 12(2):337–65, février 2000.

[LS01] D.Lee et S.Seung : Algorithms for non-negative matrix factorization. *Neural Information Processing Systems Conference*, pages 556–562, 2001.

[LT06] D.Leviatan et V. N.Temlyakov : Simultaneous approximation by greedy algorithms. *Advances in Computational Mathematics*, 25(1-3):73–90, juillet 2006.

[LVRD08] P.Leveau, E.Vincent, G.Richard et L.Daudet : Instrument-Specific Harmonic Atoms for Mid-Level Music Representation. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):116–128, janvier 2008.

[Mac03] D. J. C.MacKay : Model Comparison and Ockham ' s Razor. *In Information Theory, Inference and Learning Algorithms*. 2003.

[Mal99] H.Malvar : A Modulated Complex Lapped Transform and its Applications to Audio Processing. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1421–1424. Microsoft Research, 1999.

[Mal09] S.Mallat : *A Wavelet Tour of Signal Processing*. Elsevier, third édition, 2009.

[Mal12] S.Mallat : Group Invariant Scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, octobre 2012.

[MBAG07] E.Magli, M.Barni, A.Abrardo et M.Grangetto : Distributed Source Coding Techniques for Lossless Compression of Hyperspectral Images. *EURASIP Journal on Advances in Signal Processing*, 2007(1):045493, 2007.

[MBF94] O.Michel, R.Baraniuk et P.Flandrin : Time-frequency based distance and divergence measures. *In Proceedings of IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, pages 64–67. IEEE, 1994.

[MBP+08] J.Mairal, F.Bach, J.Ponce, G.Sapiro et A.Zisserman : Supervised dictionary learning. *In Neural Information Processing Systems Conference*, 2008.

[MC09] S.Matta et C. D.Creusere : Distributed Audio Coding with Efficient Source Correlation Extraction. *In IEEE Digital Signal Processing Workshop*, pages 16–20. IEEE, janvier 2009.

[MDR11] M.Moussallam, L.Daudet et G.Richard : Audio Signal Representations for Factorization in the sparse Domain. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 513–516, 2011.

[MDR12a] M.Moussallam, L.Daudet et G.Richard : Matching Pursuits with Random Sequential Subdictionaries. *Signal Processing*, 92:2532–2544, 2012.

[MDR12b] M.Moussallam, L.Daudet et G.Richard : Random time-frequency subdictionary design for sparse representations with greedy algorithms. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3577–3580, 2012.

[MFRD10] M.Moussallam, T.Fillon, G.Richard et L.Daudet : How Sparsely Can A SIgnal Be Approximated While Keeping Its Class Identity. *Workshop on Music and Machine Learning , ACM Multimedia*, 2010.

[MGBV09] B.Mailhe, R.Gribonval, F.Bimbot et P.Vandergheynst : A low complexity Orthogonal Matching Pursuit for sparse signal approximation with shift-invariant dictionaries. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3445–3448, 2009.

[MKC05] M.Müller, F.Kurth et M.Clausen : Audio matching via chroma-based statistical features. *International Society for Music Information Retrieval Conference*, 2005.

[MLA10] M.Moussallam, P.Leveau et S. M.Aziz Sbai : Sound enhancement using sparse approximation with speclets. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 221–224, 2010.

[MP392] Coding of moving pictures and associated audio for digital, storage at speed up to about 1.5Mbits/s Part 3 : Audio IS11192-3, 1992.

[MPE99] MPEG-4 Audio Version 2 (Final Committee Draft 14496-3 AMD1), 1999.

[MPE03] Report on Informal MPEG-4 Extension 1 (Bandwidth Extension) Verification Tests. Rapport technique, ISO/IEC JTC1/SC29/WG11/N5571, 2003.

[MRD12] M.Moussallam, G.Richard et L.Daudet : Audio Source Separation informed by Redundancy with greedy multiscale Decomposition. *In European Signal Processing Conference*, pages 2644–2648, 2012.

[MZ93] S.Mallat et Z.Zhang : Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, décembre 1993.

[Nag90] H. N.Nagaraja : Order Statistics from Discrete Distributions. *Statistics : A Journal of Theoretical and Applied Statistics*, 23, 1990.

[NDEG12] S.Nam, M.Davies, M.Elad et R.Gribonval : The cosparse analysis model and algorithms. *Applied and Computational Harmonic Analysis*, mars 2012.

[Nes07] Y.Nesterov : Gradient methods for minimizing composite objective function. Rapport technique, 2007.

[NT10] D.Needell et J. A.Tropp : CoSaMP. *Communications of the ACM*, 53(12):93, décembre 2010.

[NT12] D.Needell et J. A.Tropp : Paved with Good Intentions : Analysis of a Randomized Block Kaczmarz Method. 1(August):1–21, août 2012.

[NV10] D.Needell et R.Vershynin : Signal Recovery From Incomplete and Inaccurate Measurements Via Regularized Orthogonal Matching Pursuit. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):310–316, avril 2010.

[OF96] B. A.Olshausen et D. J.Field : Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, juin 1996.

[OF97] B. A.Olshausen et D. J.Field : Sparse coding with an overcomplete basis set : A strategy employed by V1 ? *Vision research*, 37(23):3311–3325, 1997.

[Pau07] S.Paukner : *Foundations of Gabor Analysis for Image Processing*. Thèse de doctorat, University of Vienna, Austria, 2007.

[PB86] J.Princen et A.Bradley : Analysis/Synthesis filter bank design based on time domain aliasing cancellation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(5):1153–1161, octobre 1986.

[Pee03] G.Peeters : A Large Set of Audio Features for Sound Description. Rapport technique, 2003.

[PEE12] T.Peleg, Y. C.Eldar et M.Elad : Exploiting Statistical Dependencies in Sparse Representations for Signal Recovery. *IEEE Transactions on Signal Processing*, 60(5): 2286–2303, mai 2012.

[PERA12] T.Peel, V.Emiya, L.Ralaivola et S.Anthoine : Matching Pursuit with Stochastic Selection. *In European Signal Processing Conference*, 2012.

[PGBP10] J.Pinel, L.Girin, C.Baras et M.Parvaix : A high-capacity watermarking technique for audio signals based on MDCT-domain quantization. *In 20th International Congress on Acoustics*, numéro August, 2010.

[PK09] Y.Panagakis et C.Kotropoulos : Music genre classification via sparse representations of auditory temporal modulations. *In European Signal Processing Conference*, 2009.

[PLR02] G.Peeters, A.La Burthe et X.Rodet : Toward Automatic Music Audio Summary Generation from Signal Analysis. *In International Society for Music Information Retrieval Conference*, pages 94–100, 2002.

[PM00] H.Purnhagen et N.Meine : HILN-the MPEG-4 parametric audio coding tools. *In IEEE International Symposium on Circuits and Systems. Emerging Technologies for the 21st Century*, volume 3, pages 201–204. Presses Polytech. Univ. Romandes, 2000.

[PNZTL11] R.Pichevar, H.Najaf-Zadeh, L.Thibault et H.Lahdili : Auditory-Inspired Sparse Representation of Multimedia Signals with Applications to Audio Coding. *Speech Communication (Elsevier)*, 53:643–657, 2011.

[PR03] S.Pradhan et K.Ramchandran : Distributed source coding using syndromes (DISCUS) : design and construction. *IEEE Transactions on Information Theory*, 49(3):626–643, mars 2003.

[PRAO02] J.Pinquier, J.-L. J.Rouas et R.André-Obrecht : Robust speech/music classification in audio documents. *International Conference on Spoken Language Processing*, 2002.

[PRAO03] J.Pinquier, J.Rouas et R.André-Obrecht : A fusion study in speech/music classification. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 409–412, 2003.

[PRK93] Y. C.Pati, R.Rezaiifar et P. S.Krishnaprasad : Orthogonal Matching Pursuit : Recursive Function Approximation with Applications to Wavelet Decomposition. *In Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers*, pages 40–44. IEEE Comput. Soc. Press, 1993.

[PVGW12] G.Puy, P.Vandergheynst, R.Gribonval et Y.Wiaux : Universal and efficient compressed sensing by spread spectrum and application to realistic Fourier imaging techniques. *EURASIP Journal on Advances in Signal Processing*, 2012(1):6, 2012.

[RA07] Y.Raimond et S.Abdallah : The music ontology. *In International Society for Music Information Retrieval Conference*, pages 1–6, 2007.

[Rak11] A.Rakotomamonjy : Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms. *Signal Processing*, 91(7):1505–1526, juillet 2011.

[Rak12] A.Rakotomamonjy : Direct Optimization of the Dictionary Learning Problem. *preprint arXiv*, 2012.

[Rav08] E.Ravelli : *Audio signal representations with overcomplete transforms for coding and indexing*. Thèse de doctorat, Universite Pierre et Marie Curie, 2008.

[RBPU08] A.Rizzi, N. M.Buccino, M.Panella et A.Uncini : Genre classification of compressed audio data. *IEEE 10th Workshop on Multimedia Signal Processing*, 4:654–659, 2008.

[RF08] V.Roth et B.Fischer : The group-lasso for generalized linear models : uniqueness of solutions and efficient algorithms. *In International Conference on Machine Learning*, 2008.

[RFB+12] M.Ramona, S.Fenet, R.Blouet, H.Bredin, T.Fillon et G.Peeters : A Public Audio Identification Evaluation Framework for Broadcast Monitoring. *Applied Artificial Intelligence : An International Journal*, 1-2(26):119–136, 2012.

[RP11a] Z.Rafii et B.Pardo : A simple music/voice separation method based on the extraction of the repeating musical structure. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 221–224, 2011.

[RP11b] M.Ramona et G.Peeters : Audio identification based on spectral modeling of bark-bands energy and synchronization through onset detection. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 477–480, 2011.

[RR09] M.Ramona et G.Richard : Comparison of different strategies for a SVM-based audio segmentation. *In European Signal Processing Conference*, 2009.

[RRD08] E.Ravelli, G.Richard et L.Daudet : Union of MDCT Bases for Audio Coding. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(8):1361–1372, novembre 2008.

[RRD10] E.Ravelli, G.Richard et L.Daudet : Audio Signal Representations for Indexing in the Transform Domain. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):434–446, mars 2010.

[RRE07] G.Richard, M.Ramona et S.Essid : Combined Supervised and Unsupervised Approaches for Automatic Segmentation of Radiophonic Audio Streams. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 461–464, 2007.

[RRVCMn+08] N.Ruiz-Reyes, P.Vera-Candeas, J. E.Muñoz, S.García-Galán et F. J.Cañadas : New speech/music discrimination approach based on fundamental frequency estimation. *Multimedia Tools and Applications*, 41(2):253–286, octobre 2008.

[RV07] O.Roy et M.Vetterli : Distributed Spatial Audio Coding in Wireless Hearing Aids. *In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 227–230, 2007.

[RV10] N.Ruiz Reyes et P.Vera Candeas : Adaptive Signal Modeling Based on Sparse Approximations for Scalable Parametric Audio Coding. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):447–460, mars 2010.

[RZE10] R.Rubinstein, M.Zibulevsky et M.Elad : Double Sparsity : Learning Sparse Dictionaries for Sparse Signal Approximation. *IEEE Transactions on Signal Processing*, 58(3):1553–1564, mars 2010.

[Sau96] J.Saunders : Real-time discrimination of broadcast speech/music. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2:993–996, 1996.

[SC10] B. L.Sturm et M. G.Christensen : Cyclic matching pursuits with multiscale time-frequency dictionaries. *In Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*, numéro 3, pages 581–585. IEEE, novembre 2010.

[SD09] L. B.Sturm et L.Daudet : On Similarity Search in Audio Signals Using Adaptive Sparse Approximations. *Workshop on Adaptive Multimedia Retrieval, Madrid Spain*, 2009.

[SM12] L.Sifre et S.Mallat : Combined scattering for rotation invariant texture analysis. *In European Symposium on Artificial Neural Networks*, 2012.

[SN12] B.Sturm et P.Noorzad : On Automatic Music Genre Recognition by Sparse Representation Classification using Auditory Temporal Modulations. *In International Symposium on Computer Music Modelling and Retrieval*, pages 1–16, 2012.

[SPZ08] P.Schniter, L. C.Potter et J.Ziniel : Fast bayesian matching pursuit. *Information Theory and Applications Workshop*, pages 326–333, janvier 2008.

[SRS08] P.Smaragdis, B.Raj et M.Shashanka : Sparse and shift-invariant feature extraction from non-negative data. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2069–2072, 2008.

[SS97] E.Scheirer et M.Slaney : Construction And Evaluation Of A Robust Multifeature Speech/music Discriminator. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 2:1331–1334, 1997.

[SSDR08] B.Sturm, J.Shynk, L.Daudet et C.Roads : Dark Energy in Sparse Atomic Estimations. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(3):671–676, mars 2008.

[SW73] D.Slepian et J.Wolf : Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, I(X), 1973.

[SW96] T.Slobada et A.Waibel : Dictionary learning for spontaneous speech recognition. *Fourth International Conference on Spoken Language*, 4:2328–2331, 1996.

[Tem02] V. N.Temlyakov : A Criterion for Convergence of Weak Greedy Algorithms. *Adv. Comp. Math.*, 17(3):269–280, 2002.

[TG05] J.Tropp et A.Gilbert : Simultaneous sparse approximation via greedy pursuit. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 721–724, 2005.

[TG07] J.Tropp et A.Gilbert : Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, décembre 2007.

[TGS06] J. a.Tropp, A. C.Gilbert et M. J.Strauss : Algorithms for simultaneous sparse approximation. Part I : Greedy pursuit. *Signal Processing*, 86(3):572–588, mars 2006.

[Tib94] R.Tibshirani : Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[Tro04] J.Tropp : Greed is good : Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, (February):1–21, 2004.

[VG08] D.Varodayan et B.Girod : Audio authentication based on distributed source coding. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 225–228, 2008.

[VGF06] E.Vincent, R.Gribonval et C.Fevotte : Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech and Language Processing*, 14(4):1462–1469, juillet 2006.

[Wan03] A.Wang : An Industrial-strength Audio Search Algorithm. *In International Society for Music Information Retrieval Conference*, pages 7–13, 2003.

[Wan06] A.Wang : The Shazam Music Recognition Service. *Communications of the ACM*, 49(8), août 2006.

[WCL09] L.Wang, E. S.Chng et H.Li : Efficient sparse self-similarity matrix construction for repeating sequence detection. *IEEE International Conference on Multimedia and Expo*, pages 458–461, 2009.

[WG97] K.Wang et D. M.Goblirsch : Extracting dynamic features using the stochastic matching pursuit algorithm for speech event detection. *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 132–139, 1997.

[WLMJ97] K.Wang, C.-h.Lee, S.Member et B.-h.Juang : Selective feature extraction via signal decomposition. *IEEE Signal Processing Letters*, 4(1):8–11, janvier 1997.

[WZ76] A.Wyner et J.Ziv : The rate-distortion function for source coding with side information at the decoder. *IEEE Transactions on Information Theory*, 22(1):1–10, janvier 1976.

[Yeu99] R.Yeung : Distributed source coding for satellite communications. *IEEE Transactions on Information Theory*, 45(4):1111–1120, mai 1999.

[YL06] M.Yuan et Y.Lin : Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society : Series B*, 68:49–67, 2006.

[YR04] O.Yilmaz et S.Rickard : Blind Separation of Speech Mixtures via Time-Frequency Masking. *IEEE Transactions on Signal Processing*, 52(7):1830–1847, juillet 2004.

[ZBZJ09] H.Zayyani, M.Babaie-Zadeh et C.Jutten : Bayesian Pursuit algorithm for sparse representation. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1549–1552, 2009.

[ZF92] R.Zamir et M.Feder : On universal quantization by randomized uniform/lattice quantizers. *IEEE Transactions on Information Theory*, 38(2):428–436, mars 1992.

[ZGLM12] Z.Zhang, A.Ganesh, X.Liang et Y.Ma : TILT : Transform Invariant Low-rank Textures. *International Journal of Computer Vision*, 99:1–24, décembre 2012.

[ZKG10] J.Zepeda, E.Kijak et C.Guillemot : Approximate nearest neighbors using sparse representations. *In IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2370–2373, 2010.

[ZL77] J.Ziv et A.Lempel : A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, 23(3):337–343, mai 1977.

[ZMW+10] W.Zhang, C.Ma, W.Wang, Y.Liu et L.Zhang : Side information based orthogonal matching pursuit in distributed compressed sensing. *In IEEE International Conference on Network Infrastructure and Digital Content*, pages 80–84, 2010.

[ZS98] R.Zamir et S.Shamai : Nested linear/lattice codes for Wyner-Ziv encoding. *In Information Theory Workshop*, pages 92–93. IEEE, 1998.

[ZT11] T.Zhou et D.Tao : Godec : Randomized low-rank & sparse matrix decomposition in noisy case. *In International Conference on Machine Learning*, 2011.

Index