



HAL
open science

Représentation et traitement temps-fréquence des signaux audionumériques pour des applications de design sonore

Florent Jaillet

► **To cite this version:**

Florent Jaillet. Représentation et traitement temps-fréquence des signaux audionumériques pour des applications de design sonore. Traitement du signal et de l'image [eess.SP]. Université de la Méditerranée Aix-Marseille II, 2005. Français. NNT : 2005AIX22045 . tel-04124238

HAL Id: tel-04124238

<https://theses.hal.science/tel-04124238>

Submitted on 9 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

Université de la Méditerranée - Aix-Marseille II
Faculté des Sciences de Luminy

THÈSE

pour obtenir le grade de

Docteur de l'Université de la Méditerranée - Aix-Marseille II

École Doctorale 353 : Physique, Modélisation et Sciences pour l'Ingénieur
Spécialité : Acoustique, Traitement du signal et Informatique Appliqués à la Musique

présentée et soutenue publiquement le 17 juin 2005 par

Florent JAILLET

Représentation et traitement temps-fréquence des signaux audionumériques pour des applications de design sonore

Jury :

| | |
|------------------------------|--------------------|
| M. Patrick BOUSSARD | Invité |
| M. Philippe DEPALLE | Rapporteur |
| M. Patrick FLANDRIN | Rapporteur |
| M. Rémi GRIBONVAL | Examineur |
| M. Richard KRONLAND-MARTINET | Examineur |
| M. Bruno TORRÉSANI | Directeur de thèse |

Remerciements

Je remercie profondément Bruno Torrèsani pour tout ce qu'il m'a appris et apporté durant ces quelques années de thèse et qu'il serait impossible de résumer ici en quelques lignes. Je tiens cependant à le remercier spécifiquement pour son soutien sans faille (très appréciable en période de vaches maigres...), pour sa disponibilité, et pour la clarté exceptionnelle de ses explications, qui, en restant très rigoureuses, ont toujours su faire appel à mon intuition. Je le remercie également pour les nombreuses qualités humaines dont il fait infailliblement preuve. Je lui exprime la très grande estime que j'ai pour lui.

Je remercie Patrick Boussard qui a rendu ce travail possible. Je le remercie en particulier pour la grande confiance qu'il m'a accordée tout au long du projet. Je le remercie également pour tout ce qu'il m'a appris sur le monde de l'entreprise. Je salue son esprit d'entreprise, son exceptionnelle persévérance et son optimisme.

Je remercie Patrick Flandrin et Philippe Depalle d'avoir accepté d'être les rapporteurs de ma thèse. Je remercie également Richard Kronland-Martinet et Rémi Gribonval pour leur participation au jury.

Je remercie tous ceux qui ont peuplé l'étage du LATP à l'IMT durant ma thèse, notamment pour les nombreuses discussions dans la salle commune et ailleurs. Parmi eux, je remercie particulièrement les membres du groupe de traitement du signal pour des aspects plus scientifiques de collaboration, avec une mention spéciale pour les membres du NUHAG venus nous rejoindre, Monika et Peter, pour l'intérêt qu'ils ont porté à mon travail et les réponses qu'ils ont données à mes questions.

Je remercie les membres anciens et actuels de GENESIS, pour l'agréable cadre de travail et l'esprit d'équipe qu'ils ont souvent su créer. Je remercie en particulier ceux avec qui j'ai directement collaboré dans le cadre de divers projets. Je remercie spécialement Florent d'avoir rendu notre vaste collaboration autour de LEA particulièrement agréable.

Je remercie les ingénieurs et chercheurs d'autres entreprises rencontrés dans le cadre du développement du logiciel LEA pour leur intérêt, leurs informations, leurs suggestions et leur collaboration. Je remercie en particulier Pierre Mollon pour son rôle déclencheur dans le développement du logiciel.

Je remercie les membres du réseau européen HASSIP (Harmonic Analysis and

Statistics for Signal and Image Processing) pour les rencontres enrichissantes qu'ils ont organisées et auxquelles j'ai pu participer. Je remercie en particulier le groupe de recherche en modélisation mathématique de l'IBB (Institut für Biomathematik und Biometrie, Neuherberg), qui m'a accueilli pour un séjour qui, malgré sa très courte durée, a été très fructueux grâce à Felix Friedrich que je remercie particulièrement.

Je remercie mes différents hôtes marseillais qui m'ont permis d'achever ma rédaction dans de bonnes conditions.

Je remercie tous ceux qui ont contribué à ce travail d'une manière ou d'une autre.

Table des matières

| | | |
|----------|--|-----------|
| 1 | Introduction | 13 |
| 1.1 | Cadre de la thèse | 13 |
| 1.2 | Le processus de design sonore industriel | 13 |
| 1.3 | Traitement temps-fréquence des sons | 18 |
| 1.4 | Structure du document | 20 |
| 1.5 | Exemple d'application | 22 |
| 2 | Représentations temps-fréquence | 27 |
| 2.1 | Les outils d'analyse temps-fréquence | 27 |
| 2.1.1 | La notion de fréquence | 27 |
| 2.1.2 | Les différentes approches temps-fréquence | 28 |
| 2.1.3 | Transformée de Gabor | 34 |
| 2.1.4 | Transformée en ondelettes | 47 |
| 2.2 | Représentations adaptées | 49 |
| 2.2.1 | Transformées covariantes par translation fréquentielle | 50 |
| 2.2.2 | Version semi-discrète | 53 |
| 2.2.3 | Transformées covariantes par translation temporelle | 54 |
| 2.2.4 | Version semi-discrète | 57 |
| 2.2.5 | Remarque sur la construction des atomes | 58 |
| 2.2.6 | Transformée de Gabor instationnaire | 60 |
| 2.2.7 | Paquets d'atomes temps-fréquence | 64 |
| 2.3 | Représentations adaptatives supervisées | 71 |
| 2.3.1 | Critères d'optimalité | 71 |
| 2.3.2 | Limites de l'optimisation globale | 74 |
| 2.3.3 | Optimisation locale supervisée | 77 |
| 2.4 | Représentations adaptatives non supervisées | 79 |
| 2.4.1 | Objectif : une représentation multifenêtre | 79 |
| 2.4.2 | Puzzle temps-fréquence | 82 |
| 2.4.3 | Étude numérique de l'algorithme | 89 |
| 2.4.4 | Puzzle temps-fréquence avec seuillage sur les valeurs d'entropie | 94 |

| | | |
|----------|---|------------|
| 3 | Détection et estimation de composantes | 101 |
| 3.1 | Composantes localement harmoniques | 101 |
| 3.1.1 | Définition et caractérisation | 101 |
| 3.1.2 | Approches courantes pour la détection | 104 |
| 3.1.3 | Critère pour la détection de composantes localement harmoniques | 105 |
| 3.1.4 | Discrétisation du critère | 107 |
| 3.1.5 | Résolution par optimisation stochastique | 108 |
| 3.1.6 | Résolution par programmation dynamique | 117 |
| 3.1.7 | Reconstruction de composantes localement harmoniques | 120 |
| 3.2 | Composantes transitoires | 122 |
| 3.2.1 | Caractérisation | 122 |
| 3.2.2 | Détection dans le cadre du puzzle temps-fréquence | 123 |
| 3.2.3 | Détection dans un cadre général | 124 |
| 3.2.4 | Reconstruction | 128 |
| 4 | Application | 131 |
| 4.1 | Logiciel d'Expertise Acoustique (LEA) | 132 |
| 4.1.1 | Genèse du projet LEA | 132 |
| 4.1.2 | Description générale du logiciel LEA | 133 |
| 4.1.3 | Fonctionnalités du module temps-fréquence | 137 |
| 4.2 | Exemples d'utilisations industrielles et académiques | 138 |
| 4.2.1 | Traitements de sons automobiles | 139 |
| 4.2.2 | Design sonore des sons de démarreurs | 140 |
| 4.2.3 | Étude de la perception des sons de plaques | 140 |
| 5 | Conclusion | 143 |
| 5.1 | Résultats | 143 |
| 5.1.1 | Résultats théoriques | 143 |
| 5.1.2 | Résultats pratiques | 145 |
| 5.2 | Perspectives | 146 |
| 5.2.1 | Perspectives théoriques | 146 |
| 5.2.2 | Perspectives pratiques | 148 |
| A | Détails de calculs | 149 |
| A.1 | Calcul des moments des paquets d'atomes temps-fréquence | 149 |
| A.1.1 | Paquets construits à partir de la transformée de Gabor | 149 |
| A.1.2 | Paquets construits à partir de la transformée en ondelettes | 151 |
| A.2 | Calcul des entropies pour un et deux atomes gaussiens | 152 |
| A.2.1 | Pour un atome gaussien | 152 |
| A.2.2 | Pour deux atomes gaussiens éloignés | 154 |

Table des figures

| | | |
|------|--|----|
| 1.1 | Décomposition du lien mécanique/perception dans le cadre du design sonore industriel | 15 |
| 1.2 | Boucle de traitement des sons par modification de leurs représentations temps-fréquence | 19 |
| 1.3 | Transformée de Gabor du signal original | 22 |
| 1.4 | Traitement du son d'oiseau | 24 |
| 1.5 | Traitement du son de flûte | 25 |
| 2.1 | Classification des approches temps-fréquence | 28 |
| 2.2 | Exemple de représentation temps-fréquence obtenue à partir de la distribution de Wigner-Ville | 31 |
| 2.3 | Pavage temps-fréquence pour la transformée de Gabor | 37 |
| 2.4 | Exemple de transformée de Gabor pour une impulsion | 40 |
| 2.5 | Exemple de transformée de Gabor pour un sinus | 41 |
| 2.6 | Exemple de transformée de Gabor pour une sinusoïde d'amplitude et de fréquence variant lentement dans le temps | 42 |
| 2.7 | Influence de la taille de fenêtre sur la représentation obtenue avec la transformée de Gabor | 43 |
| 2.8 | Effet du lissage par le noyau reproduisant | 45 |
| 2.9 | Pavage temps-fréquence pour la transformée en ondelettes | 49 |
| 2.10 | Exemple de pavage temps-fréquence pour la transformée de Gabor instationnaire | 52 |
| 2.11 | Exemple de pavage temps-fréquence pour la transformée K_x^g | 56 |
| 2.12 | Influence de l'évolution de la taille des fenêtres au cours du temps sur la forme des fenêtres utilisées pour la transformée de Gabor instationnaire | 62 |
| 2.13 | Comparaison des représentations obtenues avec la transformée de Gabor instationnaire et la transformée de Gabor classique | 63 |
| 2.14 | Comparaison des représentations obtenues par transformée de Gabor, transformée en ondelettes et transformée sur une échelle en Barks | 70 |
| 2.15 | Influence du critère sur le choix de la fenêtre optimale | 76 |

| | | |
|------|---|-----|
| 2.16 | Exemple d'adaptation locale supervisée | 78 |
| 2.17 | Principe de l'adaptation locale non supervisée | 81 |
| 2.18 | Structure de l'algorithme de puzzle temps-fréquence (variante 1) utilisant deux repères de Gabor | 85 |
| 2.19 | Décomposition obtenue avec l'algorithme de puzzle temps-fréquence (variante 1) | 86 |
| 2.20 | Module au carré de la transformée de Fourier d'un sinus à 1000 Hz fenêtré pour deux tailles de fenêtres | 87 |
| 2.21 | Décomposition obtenue avec l'algorithme de puzzle temps-fréquence (variante 2) | 89 |
| 2.22 | Influence du type de signal sur la convergence de l'algorithme de puzzle temps-fréquence (variante 1) | 91 |
| 2.23 | Influence de la taille des rectangles sur la convergence de l'algorithme de puzzle temps-fréquence (variante 1) | 92 |
| 2.24 | Comparaison de la convergence des deux variantes de l'algorithme de puzzle temps-fréquence | 93 |
| 2.25 | Convergence de l'algorithme de puzzle temps-fréquence (variante 1) utilisant trois tailles de fenêtres | 94 |
| 2.26 | Décomposition obtenue avec l'algorithme de puzzle temps-fréquence (variante 1) avec seuillage des valeurs d'entropie | 97 |
| 2.27 | Application de l'algorithme de puzzle temps-fréquence (variante 1) avec seuillage des valeurs d'entropie sur un signal de parole | 98 |
| 2.28 | Évolution du rapport signal à bruit en fonction du nombre d'itérations pour l'algorithme de puzzle temps-fréquence (variante 1) avec seuillage | 99 |
| 3.1 | Exemple de spectrogramme pour une composante localement harmonique | 103 |
| 3.2 | Exemple de détection de composante localement harmonique avec l'algorithme de recuit simulé | 111 |
| 3.3 | Exemple de détection de composante localement harmonique avec l'algorithme 2 | 116 |
| 3.4 | Influence de paramètre c sur la détection | 119 |
| 3.5 | Exemple de détection en présence d'un fort bruit | 120 |
| 3.6 | Évolution au cours du temps du nombre de rectangles temps-fréquence sélectionnés pour la petite fenêtre par l'algorithme de puzzle temps-fréquence | 125 |
| 3.7 | Évolution au cours du temps du nombre de rectangles temps-fréquence dont l'entropie est inférieure à un seuil fixé | 126 |
| 3.8 | Évolution au cours du temps de la somme des valeurs d'entropie obtenues pour un pavage rectangulaire du plan temps-fréquence sur un son de glockenspiel | 127 |

| | | |
|-----|---|-----|
| 3.9 | Exemple de détection de composantes transitoires | 129 |
| 4.1 | Exemple d'affichage obtenu avec le module temporel de LEA | 133 |
| 4.2 | Exemple d'affichage obtenu avec le module spectral de LEA | 134 |
| 4.3 | Exemple d'affichage obtenu avec le module temps-fréquence de LEA . | 135 |
| 4.4 | Exemple d'affichage des valeurs d'indices globaux obtenu avec le module psychoacoustique de LEA | 136 |
| 4.5 | Exemple de traitement de montée en régime automobile | 141 |

Table des algorithmes

| | | |
|---|---|-----|
| 1 | Algorithme de recuit simulé appliqué au problème d'optimisation du critère C | 110 |
| 2 | Algorithme déterministe similaire à un recuit simulé à température nulle appliqué au problème d'optimisation du critère C | 115 |

Chapitre 1

Introduction

1.1 Cadre de la thèse

J'ai réalisé le travail de thèse présenté dans ce document en tant qu'ingénieur au sein de la société GENESIS dans le cadre d'une convention industrielle de formation par la recherche (CIFRE). Ce cadre de travail a été déterminant dans l'étude qui a été menée. En effet, la thèse s'est inscrite dans le projet de recherche de GENESIS sur l'utilisation des représentations temps-fréquence pour le design sonore industriel dont le but est le développement et la commercialisation d'outils logiciels. Ainsi, le cadre applicatif de la thèse a été fixé de façon très précise dès le début et il est nécessaire de bien le cerner pour comprendre les contraintes qu'il implique et qui sont déterminantes dans les choix faits dans l'étude des aspects théoriques apparaissant dans ce travail.

En conséquence, cette partie introductive détaille l'application visée dans cette thèse en présentant dans un premier temps le processus de design sonore industriel, puis en précisant le mode d'utilisation des représentations temps-fréquence pour la transformation des sons retenu dans le cadre du projet de recherche.

1.2 Le processus de design sonore industriel

Le design sonore est une discipline en émergence aux limites encore mal définies. Le terme de design sonore peut en particulier désigner des activités variées, comme par exemple celle du concepteur de l'habillage sonore d'un site web, d'une sonnerie de téléphone, ou encore d'un jingle sonore pour une marque. Le design sonore recouvre également l'activité de celui qui va déterminer le son que doit émettre le pot d'échappement d'un nouveau véhicule automobile pour qu'il respecte un ensemble de contraintes externes, comme la limitation de la gêne perçue, les contraintes réglementaires, ou encore les demandes de constructeurs souhaitant que le son per-

sonnalise la marque ou le modèle.

Cependant, à travers ces différentes facettes, le design sonore poursuit toujours le même objectif : s'assurer que le son créé répondra bien à un ensemble de critères souhaités. Ces critères seront de natures variées selon le type de design sonore pratiqué et les sons mis en jeu. Il s'agira par exemple de critères perceptifs, comme la limitation de la gêne ou le jeu sur le timbre du son, de critères physiques, comme la limitation de la puissance acoustique rayonnée, de critères esthétiques, ou encore de critères de fonctionnalité, lorsque le son doit transmettre un message (par exemple, un signal d'urgence doit bien être perçu comme tel), ou de tout autre critère que le designer souhaite prendre en compte.

Les outils créés dans cette thèse s'adressent à un type de design sonore bien spécifique que nous qualifierons de design sonore industriel.

Dans les exemples de design sonore tels que l'habillage d'un site web ou la création d'un jingle sonore, les contraintes physiques sur la production sonore sont assez faibles puisque le son va être créé par un système dédié à la reproduction du son et capable de reproduire fidèlement tout type de son. Dans ce type de cas, la liberté de création dans le design des sons est quasi-totale en ce qui concerne les contraintes physiques de production du son.

Dans le cas du design sonore industriel, l'industriel conçoit un produit (au moins partiellement) mécanique, et c'est le fonctionnement mécanique du produit qui est à l'origine du son. Ainsi les caractéristiques mécaniques déterminent le son émis et les contraintes liées à la conception mécanique du produit ont donc une grande influence sur les possibilités de design sonore.

Dans la démarche de design sonore industriel, on cherche idéalement à établir les liens entre les paramètres mécaniques et la perception. Cependant, pour la plupart des produits industriels, compte tenu de la complexité et de la finesse de la perception humaine, et des limitations des simulations mécaniques à l'heure actuelle, il n'est pas possible d'étudier directement ces liens. En effet, les simulations ne permettant en général pas de synthétiser des sons suffisamment réalistes pour recueillir des jugements d'auditeurs conformes aux conditions normales d'utilisation des produits, il serait nécessaire de réaliser directement une série de prototypes aux caractéristiques mécaniques variées afin de corréliser ces caractéristiques à la perception qu'en ont les auditeurs. Une telle démarche demanderait un temps de réalisation et un coût trop élevé pour être employée en pratique.

En conséquence, on est amené à introduire une étape intermédiaire d'analyse et de traitement du signal pour pouvoir étudier le lien entre la mécanique et la perception. Comme on peut le voir schématisé sur la figure 1.1, l'étude se décompose alors en deux sous études distinctes. D'une part on étudie le lien entre les caractéristiques mécaniques du produit et les caractéristiques du signal acoustique qu'il émet, et, d'autre part, on étudie le lien entre les caractéristiques physiques du signal et le jugement perceptif des utilisateurs.

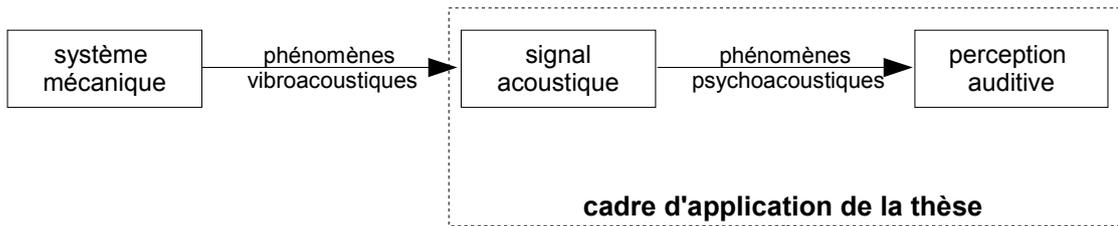


FIG. 1.1 – Décomposition du lien mécanique/perception dans le cadre du design sonore industriel

L'exemple suivant permet de bien comprendre comment cette démarche peut être mise en place et comment elle permet d'établir le lien entre la mécanique et la perception.

Exemple 1.1 *Un constructeur automobile souhaite concevoir un nouveau véhicule qui soit clairement identifié comme un véhicule sportif. Il sait que le son du moteur est déterminant dans l'appréciation du caractère sportif par ses clients et souhaite en conséquence lors de la conception s'assurer que le son sera conforme à leur attente.*

On peut considérer que le son dans l'habitacle du véhicule est constitué de deux composantes principales :

- *une composante harmonique dont la fréquence fondamentale est directement liée au régime de rotation du moteur,*
- *une composante bruitée, regroupant le bruit aérodynamique et le bruit de roulement.*

Comme on l'a déjà indiqué, pour étudier la perception du son du moteur à l'intérieur du véhicule, il serait trop coûteux de construire une batterie de moteurs prototypes en modifiant les caractéristiques mécaniques puis de recueillir un jugement perceptif d'auditeurs pour établir directement le lien entre la mécanique et la perception. Cependant, par simulation numérique, le constructeur est souvent capable d'établir le lien entre la mécanique du moteur qu'il est en train de concevoir et certains aspects du signal acoustique. Il peut par exemple estimer l'amplitude relative des partiels de la composante harmonique créée par le moteur. Cependant la connaissance de ce seul paramètre du signal ne lui permet pas de faire aboutir son étude jusqu'à la perception de l'utilisateur. En effet, les sons qu'il pourrait reconstruire en synthétisant la composante harmonique seule ne sont pas réalistes, puisqu'ils n'intègrent pas la composante bruitée.

Le constructeur a cependant la possibilité d'étudier le lien entre les caractéristiques du signal et la perception en partant d'enregistrements de signaux réels et en les modifiant avec des outils de traitement du signal. Dans le cas de notre exemple, le constructeur, pourra effectuer un enregistrement sonore d'une phase de

conduite à l'intérieur de l'habitacle d'un véhicule existant, puis modifier l'amplitude des partiels sur le son enregistré en conservant les autres composantes du son (bruit de roulement et bruit aérodynamique). Il peut ainsi faire varier très précisément les caractéristiques des partiels tout en conservant le réalisme des sons. À partir de sons créés de cette manière, le constructeur peut recueillir les jugements perceptifs d'utilisateurs. Le constructeur peut ainsi déterminer un son cible ayant les caractéristiques perceptives souhaitées pour le véhicule sportif qu'il est en train de concevoir. Les niveaux relatifs des partiels de ce son sont alors utilisés comme référence dans la conception mécanique, permettant au constructeur de faire aboutir sa démarche de design sonore.

Remarque 1.1 *Il est à noter que la procédure de recueil des jugements perceptifs est à la fois complexe, longue et coûteuse. En effet, elle fait intervenir les méthodes de la psychoacoustique et nécessite en conséquence tout un groupe d'auditeurs écoutant et jugeant un ensemble de sons créés par le constructeur en faisant varier les caractéristiques du signal. Une analyse statistique des jugements émis par les auditeurs est ensuite effectuée pour corrélérer les différentes caractéristiques du jugement aux différents paramètres des signaux. Cet aspect fait ressortir un des intérêts de la décomposition en deux phases de l'étude du lien entre mécanique et perception. L'industriel peut n'effectuer qu'une seule fois l'étude de lien signal/perception et peut la réutiliser pour divers modèles d'un même produit. De plus, une fois l'étude du lien signal/perception réalisée, l'industriel a la possibilité d'exprimer des spécifications perceptives directement à partir de grandeurs mesurées sur le signal, ce qui permet d'intégrer beaucoup plus facilement les contraintes perceptives dans le cahier des charges du produit et dans l'ensemble du processus de conception.*

Les outils qui vont être développés dans cette thèse sont destinés à la phase d'étude du lien signal/perception et plus précisément à la transformation de signaux réels pour créer de nouveaux sons qui pourront servir pour le recueil du jugement perceptif. Il s'agira en conséquence d'outils de traitement des signaux audio numériques dont on peut relever quelques spécificités :

- Ces outils étant destinés à l'étude de la perception de sons naturels, ils doivent permettre de créer des sons réalistes. En particulier ils doivent permettre de limiter les artefacts de traitement pour que le traitement ne puisse pas être repéré par l'auditeur.

Pour utiliser facilement les traitements dans ce contexte, on doit également pouvoir leur attribuer un certain "sens perceptif" afin de traiter de manière globale un ensemble de composantes qui ont une unité perceptive et de pouvoir anticiper l'effet du traitement. Par exemple, on souhaite qu'une composante harmonique, perçue comme un élément unique, puisse être traitée comme un unique objet, plutôt que d'avoir à traiter individuellement chacun de ses partiels.

On peut noter que cette spécificité de traitement marque une différence avec l'utilisation des traitements audionumériques dans le contexte musical, cadre le plus courant de leur étude. En effet, le musicien s'inscrivant dans une démarche de création peut utiliser les traitements comme un outil de création de son à part entière et il peut même être amené à utiliser spécifiquement les artefacts de traitement pour créer de nouveaux sons, sans tenir compte d'une quelconque contrainte de réalisme.

- Même si le lien mécanique/signal et les outils pour l'analyser n'entrent pas dans le cadre de cette thèse, il est nécessaire de tenir compte du fait que cette phase devra être étudiée dans une démarche globale de design sonore industriel. On doit en conséquence pouvoir attribuer aux traitements un "sens mécanique". Il faut donc éviter de proposer des traitements trop "exotiques" qu'il serait impossible de relier à des effets mécaniques. Il s'agit là aussi d'une limitation qui n'apparaît pas dans les applications musicales.
- Compte tenu de l'application visée, le contexte d'utilisation peut être précisé et les contraintes associées dégagées. Les outils sont destinés aux ingénieurs travaillant dans l'acoustique industrielle, ils doivent en conséquence pouvoir s'intégrer facilement dans leurs méthodes de travail.

D'un point de vue matériel, ceci implique qu'il doivent n'utiliser que des supports standards. En l'occurrence, la discussion avec des industriels a fait ressortir que les traitements doivent pouvoir être réalisés sur un ordinateur PC standard pour des sons de 5 minutes maximum avec une fréquence d'échantillonnage autour de 50 kHz en conservant des temps de calcul raisonnables.

Les outils doivent proposer un niveau de confort d'utilisation professionnel. Ceci implique de limiter la difficulté d'utilisation des outils, afin de ne pas requérir un niveau d'expertise trop important. Pour cela, l'ergonomie d'utilisation doit être spécialement étudiée et les outils conçus doivent être suffisamment intuitifs pour pouvoir s'adresser à des utilisateurs non-spécialistes en traitement du signal.

Les outils doivent cependant rester suffisamment généralistes pour répondre aux besoins des différentes applications industrielles. On pourra ainsi par exemple être amené à préférer construire un outil nécessitant l'intervention de l'utilisateur mais lui permettant d'effectuer des traitements dans des conditions difficiles, plutôt que proposer des outils totalement automatiques qui ne fonctionneront de façon satisfaisante que pour une classe trop limitée de signaux.

Après avoir précisé l'objectif de cette thèse, c'est à dire construire des outils d'analyse et de transformation des signaux audionumériques adaptés au design sonore industriel, nous allons maintenant préciser l'a priori fait dans le cadre du projet de recherche associé à la thèse : utiliser la représentation graphique temps-fréquence

des signaux comme base des outils de traitement.

1.3 Traitement temps-fréquence des sons

Les outils de traitement audionumérique pour des applications de design sonore industriel doivent permettre de modifier les sons de manière fine, et ce pour des signaux réels qui peuvent être complexes. En particulier, les sons étudiés seront très souvent non-stationnaires, c'est à dire que leur contenu évoluera en fonction du temps, éventuellement rapidement. C'est par exemple le cas des partiels du bruit de moteur signalé dans l'exemple de design sonore précédent, puisqu'ils varient en fonction du temps avec le régime du moteur, c'est également le cas de tous les sons transitoires comme les bruits de chocs.

Les éditeurs de sons commerciaux actuellement disponibles, développés pour des applications musicales et audiovisuelles, ne permettent en général de ne réaliser que des traitements purement temporels (copie, coupe, collage, mixage de segments temporels de signaux) ou purement fréquentiels (filtrage linéaire). Ces traitements restent trop limités pour les applications de design sonore industriel, en particulier pour le traitement des sons non-stationnaires. Pour proposer des traitements plus puissants, GENESIS a été amené à chercher des outils mieux adaptés et s'est naturellement tourné vers la représentation temps-fréquence des signaux qui est bien adaptée à l'analyse des signaux non-stationnaires et permet un lien naturel avec la perception auditive à travers la notion de fréquence.

Le mode d'utilisation des représentations temps-fréquence retenu pour le projet de recherche est décrit sur la figure 1.2 tirée de [24]. Il s'agit de mettre en place une boucle d'analyse/modification/synthèse des signaux qui se décompose de la façon suivante :

- l'analyse correspond au calcul de la transformée temps-fréquence d'un signal original,
- la modification se fait directement sur les coefficients de la transformée; une représentation construite à partir des coefficients de cette transformée est présentée à l'utilisateur qui peut agir sur la transformée à l'aide d'outils spécifiques contrôlés à la souris,
- La synthèse correspond à la reconstruction du signal à partir de la transformée modifiée.

L'utilisateur peut à la suite de sa modification écouter le signal reconstruit pour juger si la transformation correspond à ses attentes. S'il est satisfait, il conserve la modification et peut poursuivre son traitement en recommençant la procédure à partir du son modifié; s'il ne l'est pas, il annule la modification et reprend le traitement à partir du son avant modification. Ce fonctionnement par essai/correction permet de converger rapidement vers le traitement souhaité et facilite l'exploration

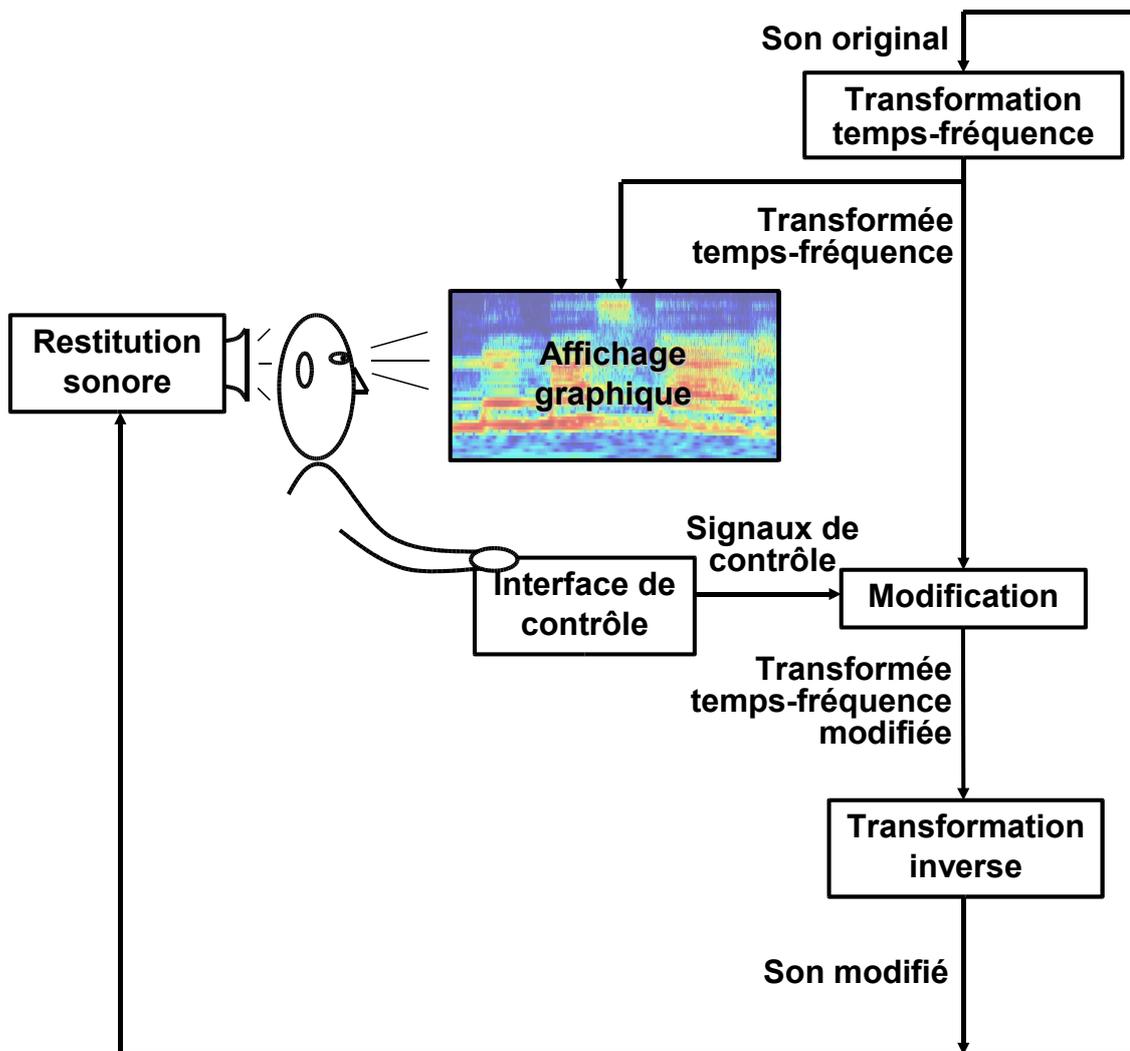


FIG. 1.2 – Boucle de traitement des sons par modification de leurs représentations temps-fréquence

du lien signal/perception, l'utilisateur étant en mesure à chaque instant d'écouter l'effet de sa transformation.

1.4 Structure du document

L'objet de cette thèse est l'étude des éléments théoriques et pratiques nécessaires à la mise en place de la boucle de traitement temps-fréquence décrite sur la figure 1.2, le but étant de proposer un cadre cohérent permettant de construire un outil complet et effectif de traitement.

En considérant la boucle de traitement temps-fréquence, on peut identifier deux points-clefs concernant l'étude théorique :

Comment “bien” représenter le signal dans le plan temps-fréquence ?

Il est nécessaire de bien maîtriser les étapes de transformation temps-fréquence (calcul de la représentation temps-fréquence) et de son inversion, en assurant en particulier que la représentation soit bien adaptée à notre cadre d'application. Ce problème fait l'objet du chapitre 2.

Comment détecter et estimer des composantes structurées ?

Pour permettre à l'utilisateur d'effectuer des traitements puissants et facilement interprétables perceptivement, il est nécessaire de lui donner des outils pour agir directement sur des composantes structurées. Se pose alors le problème de la détection et de l'estimation de ces composantes sur la représentation temps-fréquence. Ce problème fait l'objet du chapitre 3.

Pour la mise en place effective de la boucle de traitement il est nécessaire de préciser comment ces éléments théoriques peuvent être appliqués en pratique, ce qui est l'objet du chapitre 4.

La structure du document est la suivante :

chapitre 2 Ce chapitre décrit les représentations temps-fréquence utilisées dans la boucle de traitement, en précisant comment certaines limitations des représentations classiques peuvent être dépassées et en proposant une approche pour assurer la bonne qualité de la représentation.

Nous identifions, parmi les approches temps-fréquence courantes, celle qui paraît la mieux adaptée à notre cadre de travail : l'approche atomique. Nous présentons alors les décompositions atomiques courantes (transformée de Gabor, transformée en Ondelettes).

Nous étudions ensuite comment outrepasser certaines limitations auxquelles sont soumises ces transformées. Pour cela, nous construisons des décompositions atomiques dans lesquelles la résolution temps-fréquence peut évoluer très librement en fonction du temps ou de la fréquence.

Puis nous étudions comment optimiser de manière automatique la qualité de la représentation temps-fréquence en introduisant des éléments d'adaptativité dans la décomposition. Pour cela, on introduit une notion de "meilleure" représentation basée sur des critères entropiques mesurant la parcimonie des représentations.

On étudie alors deux cas de figure :

- d'abord dans un contexte supervisé, on propose une méthode permettant à l'utilisateur de déterminer les composantes pour lesquelles il souhaite obtenir une bonne représentation,
- puis dans un contexte non-supervisé, en systématisant l'approche précédente, on met en place un algorithme numérique de décomposition atomique adaptative désigné sous le nom de puzzle temps-fréquence. Plusieurs variantes de l'algorithme sont décrites et étudiées.

Cet algorithme permet d'obtenir une décomposition de Gabor multifenêtre qui peut également être formulée sous forme de décomposition multicouche. On peut ainsi notamment obtenir une décomposition du signal sous la forme d'une somme d'une composante tonale, d'une composante transitoire et d'un résidu.

chapitre 3 Ce chapitre présente des méthodes de détection et d'estimation de composantes structurées à partir des représentations temps-fréquence des signaux. On s'intéresse à deux types de composantes : les composantes localement harmoniques et les composantes transitoires.

En ce qui concerne les composantes localement harmoniques, après les avoir définies puis caractérisées dans le plan temps-fréquence, nous décrivons un critère permettant par optimisation de détecter ces composantes. Nous étudions ensuite deux approches pour réaliser l'optimisation :

- d'une part, une approche stochastique, utilisant l'algorithme de recuit simulé, en précisant les limites de cette approche dans notre contexte,
- d'autre part, une approche par programmation dynamique, qui permet une résolution déterministe efficace du problème.

En ce qui concerne la détection de composantes transitoires, nous explorons comment les éléments mis en place au chapitre précédent dans l'algorithme de puzzle temps-fréquence peuvent être utilisés pour la détection de transitoires.

chapitre 4 Ce chapitre précise la manière dont les éléments théoriques présentés dans les deux chapitres précédents ont été mis en pratique. Pour cela, nous précisons l'application logiciel qui en a été faite à travers le développement du logiciel d'expertise acoustique (LEA) développé et commercialisé par GENESIS. Puis nous décrivons des traitements réalisés avec ces outils pour des applications de design sonore.

chapitre 5 Ce chapitre conclut le document en résumant les résultats obtenus dans

cette thèse et en évoquant les perspectives qu'ils ouvrent.

Remarque 1.2 *Pour des raisons de confidentialité, certains compléments sur l'implémentation des transformées et des algorithmes présentés dans cette thèse ne sont pas décrits dans ce document. Ceci ne devrait cependant pas nuire à la compréhension des éléments présentés, ni empêcher de juger de la véracité des résultats y apparaissant.*

Ces compléments sur l'implémentation ont été regroupés dans un rapport interne de GENESIS [25].

1.5 Exemple d'application

Pour conclure cette introduction, nous présentons au lecteur un exemple de traitement complet réalisé avec le logiciel LEA (présenté en 4.1) partiellement développé dans le cadre de cette thèse. Cet exemple permet de bien identifier le type de traitement que l'on souhaite réaliser et également de voir comment s'articulent les divers éléments présentés dans la thèse.

La figure 1.3 montre le spectrogramme du signal original obtenue à partir de la transformée de Gabor (décrite en 2.1.3). Le signal est constitué du mixage d'une note de flûte traversière avec un son d'oiseau. On peut noter sur la représentation les structures horizontales qui correspondent aux partiels harmoniques du son de flûte, et dans la zone de 1s à 2,5s et de 2500Hz à 4000Hz, les cris d'oiseau qui émergent du son de flûte.



FIG. 1.3 – Transformée de Gabor du signal original

En utilisant l'outil d'adaptation automatique de la taille de fenêtre (basé sur les éléments présentés dans 2.3.3), on choisit la taille la mieux adaptée à la représentation des cris d'oiseau.

On obtient ainsi la représentation apparaissant en haut de la figure 1.4. On voit que sur cette représentation, l'énergie correspondant aux cris d'oiseau est effectivement mieux localisée et peut facilement être sélectionnée, à l'aide des outils de sélection disponible dans LEA. On peut alors extraire le son d'oiseau pour obtenir le signal dont la représentation apparaît au milieu de la figure 2.3.3, ou encore supprimer l'énergie correspondant aux cris d'oiseau pour ne conserver que le son de flûte et obtenir la représentation apparaissant en bas de la même figure.

On peut alors continuer le traitement sur le son de flûte ainsi extrait. Pour cela, on applique l'outil d'adaptation de taille de fenêtre sur les partiels harmoniques du son. On obtient ainsi la représentation visible en haut de la figure 1.5, sur laquelle les partiels sont bien localisés. On peut alors utiliser l'outil de détection des composantes localement harmoniques (basé sur les éléments présentés en 3.1). On sélectionne de cette manière les régions de la transformée correspondant aux partiels harmoniques, qui peuvent être extraits, en obtenant la représentation au milieu de la figure 1.5, ou supprimés, pour ne conserver que la composante bruitée du son de flûte, correspondant au bruit de souffle, apparaissant en bas de la figure 1.5.

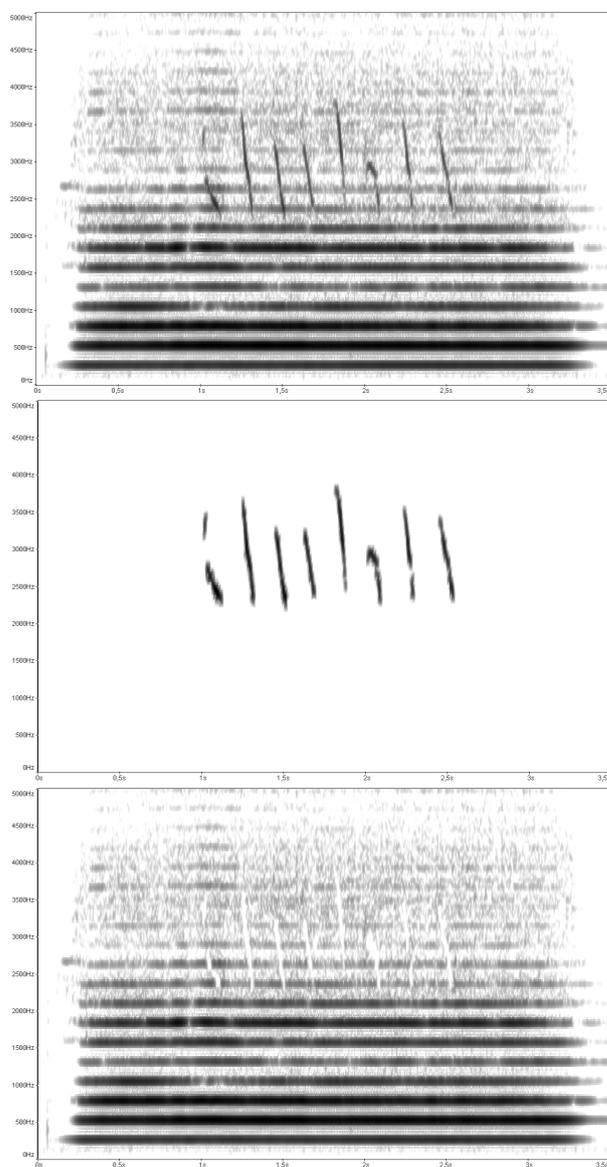


FIG. 1.4 – Traitement du son d’oiseau : en haut, transformée de Gabor du signal original avec une taille de fenêtre optimisée pour la représentation des cris d’oiseau, au milieu, transformée de Gabor des cris d’oiseau extraits, en bas, transformée de Gabor du son de flûte extrait en supprimant le son d’oiseau

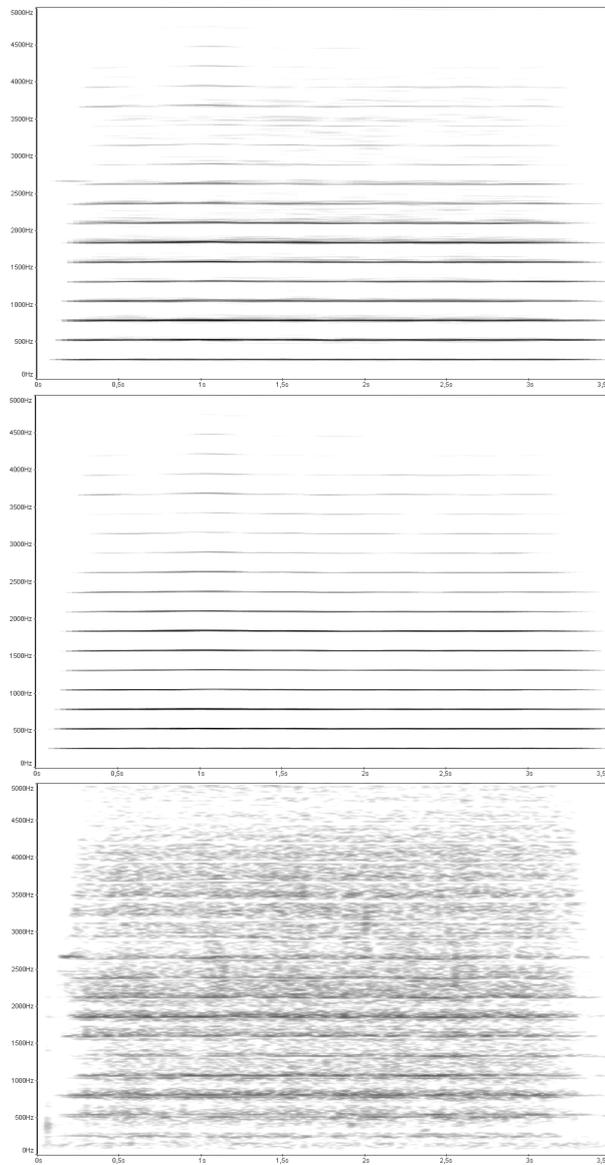


FIG. 1.5 – Traitement du son de flûte : en haut, transformée de Gabor du son de flûte extrait avec une taille de fenêtre optimisée pour la représentation des partiels harmoniques, au milieu, transformée de Gabor de la composante harmonique extraite, en bas, transformée de Gabor de la composante bruitée du son de flûte

Chapitre 2

Représentations temps-fréquence

2.1 Les outils d'analyse temps-fréquence

2.1.1 La notion de fréquence

Le signal temporel représentant l'évolution de la pression acoustique recueillie lors d'une prise de son ne donne pas directement une bonne représentation de l'information pour une interprétation perceptive. Ainsi, pour pouvoir exploiter le signal, une étape d'analyse est nécessaire pour structurer l'information de manière mieux adaptée.

L'oreille est sensible à la façon dont évolue la pression acoustique au cours du temps. En particulier, la notion de fréquence de cette évolution apparaît comme un des attributs principaux de la perception auditive. On est ainsi naturellement amené à construire des notions mathématiques qui pourront facilement être reliées à cet attribut perceptif pour structurer l'information du signal.

Mathématiquement, la notion de fréquence apparaît naturellement pour les signaux périodiques comme l'inverse de la période du signal. Cependant pour traiter des signaux réels, il est nécessaire d'étendre la classe des signaux pour lesquels la notion est définie. Pour un signal $x \in L^2(\mathbb{R})$, la généralisation de la notion de fréquence est obtenue en définissant la transformée de Fourier par :

$$\hat{x}(\nu) = \int x(t) e^{-i2\pi\nu t} dt. \quad (2.1)$$

La transformée de Fourier permet de représenter le signal sous forme d'une somme infinie de composantes sinusoidales complexes correspondant chacune à une fréquence pure et s'étendant dans le temps de $-\infty$ à $+\infty$, selon la relation d'inversion :

$$x(t) = \int \hat{x}(\nu) e^{i2\pi\nu t} d\nu. \quad (2.2)$$

La notion de fréquence introduite dans cette relation n'est cependant facilement interprétable que pour les sons stationnaires. En effet, on peut voir dans la formule d'inversion de la transformée de Fourier que le signal est représenté comme une somme de composantes fréquentielles élémentaires à support temporel infini, c'est à dire qu'implicitement, on suppose que la composante est toujours présente dans le signal au cours du temps.

Cette notion de fréquence permet donc difficilement de rendre compte de l'évolution temporelle du contenu spectral du signal, qui est pourtant naturelle d'un point de vue perceptif, comme en témoigne par exemple la notation musicale occidentale dont le propos est justement de décrire cette évolution.

Pour obtenir une représentation qui rende compte du contenu fréquentiel du signal, mais également de son évolution au cours du temps, on utilise des représentations temps-fréquence, qui font à la fois intervenir des variables temporelles liées à l'évolution du signal au cours du temps et des variables fréquentielles liées au contenu spectral.

2.1.2 Les différentes approches temps-fréquence

Nous proposons sur la figure 2.1 **une**¹ classification possible des différentes approches temps-fréquence. En parcourant cette classification, nous allons rapidement présenter les différentes approches temps-fréquence les plus courantes. Ceci permettra de bien situer le cadre d'étude retenu dans la suite du chapitre et de préciser pourquoi ce cadre a été retenu.

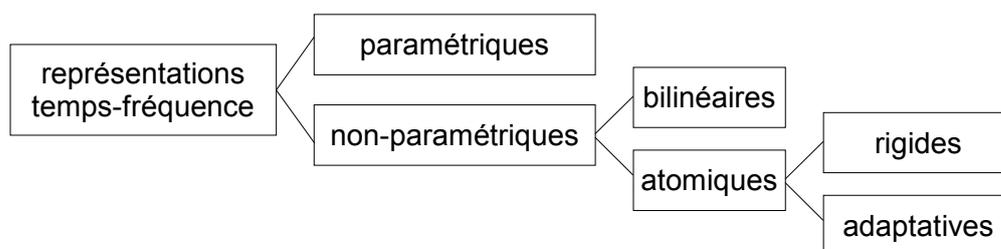


FIG. 2.1 – Classification des approches temps-fréquence

Approche paramétrique

L'approche paramétrique suppose une connaissance a priori des caractéristiques du signal permettant de déterminer un modèle du signal. L'analyse consiste alors

¹D'autres classifications sont bien-sûr possibles.

à estimer, à partir du signal, les paramètres du modèle correspondant. C'est alors la connaissance du modèle et de ses paramètres qui caractérisent le contenu spectral du signal et son évolution au cours du temps. On peut signaler par exemple la modélisation du signal sous forme de somme de sinusoides variant lentement dans le temps souvent utilisée pour le signal audionumérique dans les applications de traitement de la parole et d'informatique musicale [33] [42] [11] [10] (voir la bibliographie de [43] pour plus de références).

L'intérêt de l'approche paramétrique, pour un modèle bien choisi, est de permettre la représentation de l'information sur un faible nombre de coefficients qui sont les paramètres du modèle. La représentation du signal obtenue peut ainsi être très concise. De plus, l'interprétation physique de ces coefficients est en général très facile car elle est directement liée à la modélisation choisie. Les représentations paramétriques peuvent en conséquence constituer des outils puissants pour l'analyse et la transformation des signaux.

Cependant la représentation paramétrique, comme elle s'appuie sur un modèle de signal, n'est bien adaptée qu'aux signaux qui correspondent effectivement à ce modèle. Compte tenu de l'universalité recherchée pour les outils développés dans cette thèse, cette caractéristique sera trop limitative pour concevoir des outils de traitements audionumériques suffisamment génériques pour notre application.

De plus, l'estimation des paramètres du modèle à partir du signal est en général un problème difficile, en particulier lorsque les signaux analysés s'écartent du modèle ou que le modèle est complexe. Or pour pouvoir traiter des signaux audionumériques de nature variée, on est amené à mettre en place un modèle très générique pour lequel l'estimation des paramètres est complexe. La mise en place d'un tel modèle s'appuie généralement sur le modèle de signal déjà évoqué plus haut dans lequel le signal est représenté par une somme de sinusoides variant dans le temps, auquel il faut au minimum adjoindre un modèle spécifique pour les composantes transitoires, et un autre pour les composantes bruitées, afin d'obtenir une représentation convenable pour une classe de sons suffisamment étendue. Pour un tel modèle, compte tenu de la complexité du problème, l'estimation des paramètres se fait généralement à partir de représentations temps-fréquence non-paramétriques. Ainsi la qualité de l'estimation dépend de la qualité des représentations non-paramétriques utilisées. Dans ces conditions, l'étude des représentations non-paramétriques apparaît comme un préalable à l'utilisation des représentations paramétriques.

Compte tenu de cette constatation et des limitations des méthodes paramétriques, nous avons choisi de fonder notre étude sur la deuxième approche, l'approche non-paramétrique, pour construire nos outils de traitement du son.

Approche non-paramétrique

L'approche dite non-paramétrique ne présuppose pas de connaissance a priori sur les signaux. Dans cette approche on cherche à structurer l'information présente dans le signal de manière mieux adaptée à son analyse. Plus précisément, le but est de construire à partir du signal une fonction de deux variables, le temps et la fréquence (ou des variables qui sont directement liées à ces deux variables), qui va décrire le contenu du signal pour l'instant et la fréquence donnés.

Dans l'approche non-paramétrique, on peut distinguer deux types de représentations :

- d'une part, les distributions bilinéaires temps-fréquence d'énergie, construites en représentant l'énergie de signal dans le plan temps-fréquence,
- d'autre part, les décompositions atomiques du signal, dans lesquelles le signal est représenté sous la forme d'une somme de composantes bien localisées en temps et en fréquence, les coefficients de la décomposition fournissant une représentation alternative du signal.

Approche non-paramétrique bilinéaire

Les distributions bilinéaires temps-fréquence, sont des fonctions $\rho_x(\tau, \nu)$ de deux variables, le temps τ et la fréquence ν , qui permettent de représenter l'énergie E_x du signal x dans le plan temps-fréquence. Elle vérifient pour cela :

$$E_x = \int |x(t)|^2 dt = \iint \rho(\tau, \nu) d\tau d\nu. \quad (2.3)$$

En imposant certaines propriétés aux fonctions ρ_x , on détermine la classe des distributions bilinéaires temps-fréquence, appelée la classe de Cohen [5] [20], dans laquelle la distribution de Wigner-Ville tient une place centrale. La définition de cette distribution est :

$$W_x(\tau, \nu) = \int x\left(\tau + \frac{t}{2}\right) \bar{x}\left(\tau - \frac{t}{2}\right) e^{-i2\pi\nu t} dt. \quad (2.4)$$

La présence de termes d'interférences dans les distributions bilinéaires limite fortement les possibilités d'utilisation de ces dernières pour la modification des signaux audionumérique. Plus précisément, en l'écrivant pour la distribution de Wigner-Ville définie plus haut, la transformée de la somme de deux signaux x et y n'est pas simplement la somme des transformées des deux signaux, mais on a :

$$W_{x+y}(\tau, \nu) = W_x(\tau, \nu) + W_y(\tau, \nu) + 2 \Re(W_{xy}(\tau, \nu)) \quad (2.5)$$

où

$$W_{xy}(\tau, \nu) = \int x\left(\tau + \frac{t}{2}\right) \bar{y}\left(\tau - \frac{t}{2}\right) e^{-i2\pi\nu t} dt. \quad (2.6)$$

Le terme d'interférence $2\Re(W_{xy}(\tau, \nu))$ fait apparaître des valeurs souvent non-négligeables dans la représentations entre les composantes bien localisées du signal, ce qui limite fortement la lisibilité des représentations pour les signaux complexes, comme les signaux audionumériques. Pour des signaux complexes, il devient difficile de savoir si la distribution fait apparaître de l'énergie en raison de la présence d'une composante ou à cause des interférences entre deux composantes. L'interprétation de ces représentations peut demander en conséquence une expertise que n'ont en général pas les utilisateurs visés dans cette thèse.

Exemple 2.1 *La figure 2.2 illustre la présence des interférences pour un signal de synthèse constitué d'une somme de deux sinus de fréquence 3000 Hz et 6000 Hz modulés par une enveloppe gaussienne. Sur la représentation, on voit effectivement apparaître les composantes sinusoïdales aux fréquences 3000 Hz et 6000 Hz, mais il apparaît également entre les deux composantes, autour de 4500 Hz, une zone oscillante de niveau élevé qui est due aux interférences entre les deux composantes.*

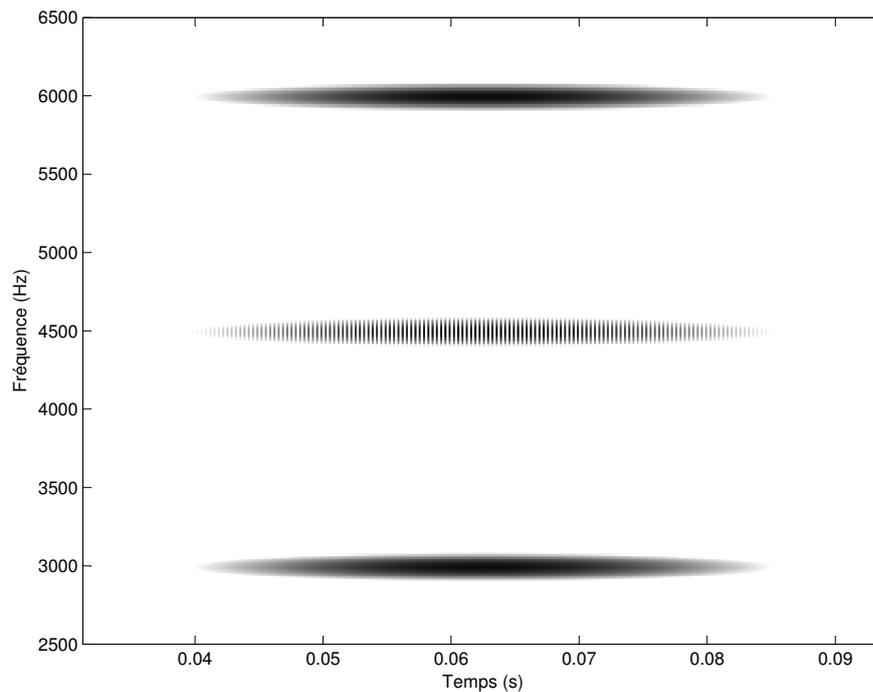


FIG. 2.2 – Exemple de représentation temps-fréquence obtenue à partir de la distribution de Wigner-Ville

Si l'on souhaite utiliser la distribution pour faire des modifications sur le signal, comme par exemple supprimer une composantes bien localisée, il est nécessaire de

supprimer à la fois les valeurs de la distribution le long de la composante, mais également les interférences entre cette composante et toutes les autres composantes du signal, ce qui rend le traitement très difficile à mettre en place.

L'importance des termes d'interférence dans les distributions bilinéaires peut être limitée en introduisant un lissage des distributions. Cependant on perd alors la propriété d'inversibilité de la distribution qui est nécessaire pour notre application, puisqu'on doit s'assurer que, lors d'un traitement, le signal est reconstruit exactement dans les zones non-modifiées de la transformée. Si la reconstruction n'est qu'approximative, l'erreur de reconstruction s'ajoute à chaque itération du traitement, et, comme le nombre d'itération peut être important, l'erreur risque de devenir trop importante.

Pour ces différentes raisons, nous n'avons par retenu les distributions bilinéaires temps-fréquence pour construire les outils pour le traitement des sons dans cette thèse. Les décompositions atomiques qui vont maintenant être décrites leur ont été préférées. Il faut cependant noter que pour l'affichage visuel des représentations, ce sont effectivement des distributions bilinéaires temps-fréquence construites à partir des décompositions atomiques qui sont représentées, comme cela sera précisé dans la suite.

Approche non-paramétrique atomique

Le principe des décompositions atomiques est de représenter le signal sous la forme d'une combinaison linéaire de fonctions localisées dans le plan temps-fréquence. La décomposition obtenue pour le signal x sera donc de la forme :

$$x = \sum_k a_k f_k, \quad (2.7)$$

où $(f_k)_{k \in \mathbb{Z}}$ est la famille² de fonctions de $L^2(\mathbb{R})$ sur laquelle est effectuée la décomposition et $(a_k)_{k \in \mathbb{Z}}$ une famille de nombres complexes qui constituent les coefficients de la décomposition.

Les coefficients a_k donnent accès à une autre représentation du signal qui contient toute l'information puisqu'elle permet la reconstruction du signal. L'interprétation temps-fréquence de cette information se fait grâce à la connaissance des fonctions f_k et en particulier grâce à la connaissance de leurs localisations temps-fréquence.

La modification des signaux se fait facilement à partir de la décomposition en modifiant directement les coefficients a_k .

²La famille choisie peut être indénombrable, la somme étant alors remplacée par une intégrale.

Approche non-paramétrique atomique “rigide”

Nous utilisons le terme “rigide” pour distinguer l’approche classique, dans laquelle la construction des atomes se fait de manière assez contrainte à partir d’atomes de base, des approches adaptatives présentées dans la suite, et qui permettent une plus grande liberté dans le choix des atomes utilisés.

Pour construire les familles de fonctions utilisées dans les décomposition atomiques, l’approche la plus courante est de rechercher des familles de fonctions pouvant être utilisées de la même manière que des bases de l’espace des signaux considérés. Le calcul des coefficients est alors simple et se fait par projection sur les éléments de la famille pour obtenir une décomposition de la forme :

$$x = \sum_k \langle x, f_k \rangle f_k. \quad (2.8)$$

Dans l’approche classique, les atomes sont de plus construits à partir d’une fonction de base, un atome père, en appliquant des transformations spécifiques, de type translation temporelle, modulation fréquentielle, ou encore dilatation, et en s’assurant que la famille obtenue puisse être utilisée pour effectuer une décomposition de la forme (2.8). C’est la démarche utilisée pour la transformée de Gabor et la transformée en ondelettes qui sont les décompositions atomiques les plus couramment utilisées, et qui sont présentées en 2.1.3 et 2.1.4.

L’intérêt de cette démarche, même si elle implique une construction assez rigide des atomes, vient du fait que grâce à la structure des familles, l’étude théorique de ces décompositions peut se faire de manière poussée. Ainsi, les nombreux résultats obtenus pour ce type de décompositions permettent de bien contrôler les propriétés des décompositions dans les applications pratiques et d’assurer des mises en oeuvre efficaces des décompositions à l’aide d’algorithmes rapides, ce qui explique leur utilisation courante.

La méthode de construction des atomes permet également d’imposer des propriétés de covariance à la décomposition, ce qui facilite son interprétation. Par exemple, lorsque la décomposition est covariante par translation temporelle, la transformée d’une copie translatée du signal est une version translatée de la transformée du signal.

Les décompositions atomiques “rigides” peuvent ainsi être utilisées de façon satisfaisante pour de nombreuses applications, et sont en particulier bien adaptées à l’application qui nous intéresse dans cette thèse : le traitement des signaux audio-numériques.

Pour ces raisons, **nous avons retenu cette approche pour construire les représentations temps-fréquence** dans notre boucle de traitement temps-fréquence des sons.

Cependant, afin de dépasser certaines limitations des approches classiques, nous allons étudier en 2.2 comment les représentations peuvent être améliorées en utilisant des constructions moins rigides des atomes.

Approche non-paramétrique atomique adaptative

Pour permettre une plus grande souplesse dans la décomposition et éliminer certaines des limitations apparaissant dans les décompositions rigides, des méthodes de décomposition adaptatives ont été proposées.

Ces méthodes utilisent généralement de vastes dictionnaires d'atomes temps-fréquence qui, associés à une procédure adaptative de choix des atomes utilisés en fonction du signal, permettent d'obtenir des représentations plus concises du signal en améliorant la parcimonie de la représentation, c'est à dire en minimisant le nombre de coefficients significatifs dans la décomposition.

Parmi ces décompositions, on peut citer le matching pursuit [32], le matching pursuit orthogonal [9], le basis pursuit [4] qui permettent d'obtenir des décompositions bien adaptées aux applications de traitement des sons [22]. Cependant ces différentes méthodes nécessitent une capacité de calcul qui reste aujourd'hui trop importante pour une application dans les conditions visées pour cette thèse. Nous n'avons en conséquence pas étudié ces méthodes pour construire nos outils de traitement.

Cependant l'introduction d'une dose limitée d'adaptativité dans les décompositions atomiques permet à la fois d'améliorer les représentations obtenues et de faciliter l'utilisation en réduisant généralement le nombre de paramètre d'analyse à régler par l'utilisateur. Aussi, nous allons proposer des méthodes pour introduire l'adaptativité dans les décompositions rigides en section 2.3 et 2.4. Cette approche permet de profiter des bénéfices de l'adaptativité des représentations obtenues en conservant un coût de calcul compatible avec une application dans des conditions réalistes. Ces méthodes s'appuient sur l'idée de ne pas effectuer une recherche exhaustive dans le dictionnaire, trop coûteuse en temps de calcul, mais de mettre en place une recherche plus structurée.

2.1.3 Transformée de Gabor

Définition

Les atomes de la transformée de Gabor³ sont obtenus par translation temporelle et fréquentielle (modulation) d'une unique fonction que l'on désigne sous le nom de

³Nous avons choisi de limiter la présentation de la transformée de Gabor dans ce document aux éléments nécessaires aux constructions qui sont faites dans les chapitres suivants. Nous renvoyons le lecteur aux références [3] [18] [23] pour des présentations plus détaillées.

fenêtre d'analyse, notée ici g . La construction des atomes se fait en conséquence de la façon suivante :

soit g une fonction de $L^2(\mathbb{R})$ vérifiant la condition d'admissibilité :

$$\int |g(t)|^2 dt = 1, \quad (2.9)$$

la famille d'atomes $(g_{\tau\nu})_{(\tau,\nu)\in\mathbb{R}^2}$ est définie par

$$g_{\tau\nu}(t) = g(t - \tau) e^{i2\pi\nu(t-\tau)}. \quad (2.10)$$

La transformée de Gabor continue G_x^g d'un signal x de $L^2(\mathbb{R})$ est obtenue par produit scalaire du signal sur les atomes de la décomposition, selon la relation :

$$G_x^g(\tau, \nu) = \langle x, g_{\tau\nu} \rangle = \int x(t) \bar{g}(t - \tau) e^{-i2\pi\nu(t-\tau)} dt. \quad (2.11)$$

Remarque 2.1 *La transformée de Gabor continue est plus communément appelée transformée de Fourier à court terme (en anglais short-time Fourier transform) ou transformée de Fourier glissante. Certains auteurs réservent le nom de transformée ou décomposition de Gabor à la version discrétisée avec une fenêtre quelconque de la transformée continue, d'autres à la version discrétisée de la transformée utilisant une fenêtre gaussienne. Ceci s'explique par le fait que l'article de Gabor [21] à l'origine de cette dénomination propose effectivement une version discrétisée de cette transformée avec fenêtre gaussienne.*

Nous avons choisi de retenir le nom de transformée de Gabor car il est associé à l'interprétation de la transformée comme décomposition atomique et que cette interprétation est particulièrement bien adaptée à la compréhension dans le cadre du traitement temps-fréquence.

Remarque 2.2 *Nous avons choisi ici une définition de la transformée de Gabor continue pour les signaux de $L^2(\mathbb{R})$, ce qui permet d'assurer l'existence de la formule d'inversion de la transformée présentée dans la suite. Cependant on voit dans l'équation 2.11 que la transformée peut être définie sur une classe plus large de fonctions ou de distributions. Nous ne détaillons pas ce point qui dépasse notre cadre de travail et nous renvoyons le lecteur intéressé à [15] [16].*

Interprétations

Plusieurs interprétations de la transformée de Gabor sont possibles :

Décomposition atomique : La première est l'interprétation comme décomposition atomique sur les atomes $g_{\tau\nu}$. On utilise dans la pratique une fenêtre g dont l'énergie est bien localisée autour de l'instant 0 et dont la transformée

de Fourier concentre elle aussi l'énergie autour de la fréquence nulle, ainsi la fonction g est bien localisée autour de l'origine dans le plan temps-fréquence. Compte-tenu de la définition des atomes, qui sont obtenus par translation temporelle et fréquentielle de g , les caractéristiques de localisation temporelle et fréquentielle sont identiques pour tous les atomes, à une simple translation dans le plan temps-fréquence près. Ainsi la décomposition se fait sur une famille d'atomes ayant les mêmes caractéristiques de localisation dans tout le plan temps-fréquence. La figure 2.3 présente une représentation symbolique des caractéristiques de localisation temps-fréquence des atomes.

Analyse de Fourier à court terme : La deuxième interprétation est à l'origine du nom de transformée de Fourier à court terme ou glissante également donné à la transformée. En définissant le signal $x_\tau(t) = x(t + \tau)\bar{g}(t)$, c'est à dire en fenêtrant le signal x autour de l'instant τ avec la fenêtre \bar{g} , la transformée de Gabor s'écrit comme la transformée de Fourier du signal fenêtré, selon la relation :

$$\hat{x}_\tau(\nu) = \int x_\tau(t)e^{-i2\pi\nu t} dt \quad (2.12)$$

$$= \int x(t + \tau)\bar{g}(t)e^{-i2\pi\nu t} dt \quad (2.13)$$

$$= \int x(t')\bar{g}(t' - \tau)e^{-i2\pi\nu(t' - \tau)} dt' \quad (2.14)$$

$$= G_x^g(\tau, \nu). \quad (2.15)$$

Pour un instant τ la transformée peut donc être interprétée comme un spectre à court terme du signal, la transformée de Gabor permettant ainsi d'analyser l'évolution spectrale du signal au cours du temps.

Remarque 2.3 *C'est cette interprétation et la formulation associée qui sont généralement utilisées pour l'implémentation pratique de la transformée. En effet, elle permet de tirer profit des algorithmes rapides de calcul de la transformée de Fourier discrète, identifiés en anglais sous le terme de fast Fourier transform (FFT), pour obtenir des implémentations performantes de la transformée.*

Banc de filtres : Enfin, on peut interpréter la transformée comme la sortie d'un banc de filtres continu. Pour chaque fréquence ν , en définissant $r_\nu(t) = \bar{g}(-t)e^{i2\pi\nu t}$, on peut écrire la transformée sous la forme du signal x filtré par le filtre de réponse impulsionnelle r_ν , selon la relation :

$$G_x^g(\tau, \nu) = \int x(t)r_\nu(\tau - t)dt. \quad (2.16)$$

La réponse en fréquence du filtre vaut :

$$\hat{r}_\nu(\xi) = -\overline{\hat{g}(f - \nu)}. \quad (2.17)$$

La transformée peut donc être interprétée comme une analyse en banc de filtres utilisant des filtres de largeur de bande identique pour toutes les fréquences et dont les réponses en fréquence sont obtenues par translation fréquentielle d'un même gabarit.

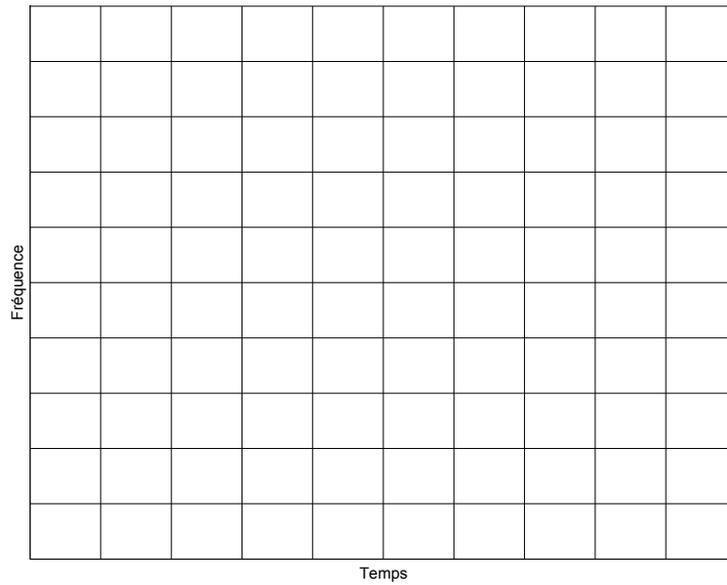


FIG. 2.3 – Pavage temps-fréquence pour la transformée de Gabor

Propriétés

Propriété 2.1 (Linéarité) La transformée de Gabor continue est linéaire, c'est à dire :

$$\forall(\lambda, \mu) \in \mathbb{C}^2, \forall(x, y) \in (L^2(\mathbb{R}))^2, \quad G_{\lambda x + \mu y}^g = \lambda G_x^g + \mu G_y^g. \quad (2.18)$$

Propriété 2.2 (Covariance) La transformée de Gabor continue est covariante par translation fréquentielle (modulation), c'est à dire :

en définissant, pour une translation fréquentielle d'une fréquence $\nu_0 \in \mathbb{R}$ d'un signal $x \in L^2(\mathbb{R})$, le signal modulé x_{ν_0} tel que $x_{\nu_0}(t) = x(t) e^{i2\pi\nu_0 t}$, on a

$$\forall(\tau, \nu) \in \mathbb{R}^2, \quad G_{x_{\nu_0}}^g(\tau, \nu) = G_x^g(\tau, \nu - \nu_0). \quad (2.19)$$

La transformée de Gabor continue est covariante par translation temporelle, c'est à dire :

en définissant, pour une translation temporelle d'une durée $\tau_0 \in \mathbb{R}$ d'un signal $x \in L^2(\mathbb{R})$, le signal translaté x_{τ_0} tel que $x_{\tau_0}(t) = x(t - \tau_0)$, on a

$$\forall(\tau, \nu) \in \mathbb{R}^2, \quad G_{x_{\tau_0}}^g(\tau, \nu) = G_x^g(\tau - \tau_0, \nu). \quad (2.20)$$

Propriété 2.3 (Isométrie) La transformée de Gabor continue est une isométrie, c'est à dire qu'elle conserve le produit scalaire par la relation :

$$\forall(x, y) \in (L^2(\mathbb{R}))^2, \quad \langle G_x^g, G_y^g \rangle = \langle x, y \rangle. \quad (2.21)$$

En effet, pour x et y dans $L^2(\mathbb{R})$, on a :

$$\langle G_x^g, G_y^g \rangle = \iint G_x^g(\tau, \nu) \overline{G_y^g(\tau, \nu)} d\tau d\nu \quad (2.22)$$

$$= \iiint x(t) \overline{y(u)} g(u - \tau) \overline{g(t - \tau)} e^{i2\pi\nu(u-t)} dt du d\tau d\nu \quad (2.23)$$

$$= \iiint x(t) \overline{y(u)} g(u - \tau) \overline{g(t - \tau)} \delta(u - t) dt du d\tau \quad (2.24)$$

$$= \int x(t) \overline{y(t)} \left(\int |g(t - \tau)|^2 d\tau \right) dt. \quad (2.25)$$

Comme g vérifie $\int |g(t)|^2 dt = 1$, la propriété d'isométrie est bien vérifiée.

Propriété 2.4 (Inversion) La transformée de Gabor continue est inversible selon la relation suivante, dans laquelle l'égalité est prise au sens du produit scalaire⁴ dans $L^2(\mathbb{R})$:

$$\forall x \in L^2(\mathbb{R}), \quad x = \iint G_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, \quad (2.26)$$

ce qui signifie :

$$\forall(x, y) \in (L^2(\mathbb{R}))^2, \quad \langle x, y \rangle = \left\langle \iint G_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, y \right\rangle \quad (2.27)$$

En effet, pour $(x, y) \in (L^2(\mathbb{R}))^2$, en remplaçant G_y^g par sa définition, on a

$$\langle G_x^g, G_y^g \rangle = \iiint G_x^g(\tau, \nu) g_{\tau\nu}(t) \overline{y(t)} dt d\tau d\nu \quad (2.28)$$

$$= \left\langle \iint G_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, y \right\rangle \quad (2.29)$$

Comme G est une isométrie, on a bien l'égalité cherchée.

⁴L'égalité au sens du produit scalaire implique l'égalité presque partout. Pour plus de détails sur cette notion, voir par exemple [7].

Propriété 2.5 (Conservation de l'énergie) *La propriété d'isométrie implique la conservation d'énergie par la transformée, selon la relation*

$$\int |x(t)|^2 dt = \iint |G_x^g(\tau, \nu)|^2 d\tau d\nu. \quad (2.30)$$

Remarque 2.4 *Compte tenu de la relation de conservation de l'énergie, $|G_x^g|^2$ peut être interprétée comme une densité temps-fréquence d'énergie, puisqu'elle répartit l'énergie du signal dans tout le plan temps-fréquence. Elle est nommée spectrogramme et fait partie de la classe de Cohen évoquée en section 2.1.2.*

On peut noter que le problème lié aux termes d'interférences pour les distributions bilinéaires de la classe de Cohen évoqué précédemment a un effet limité dans le cas du spectrogramme. Comme la transformée de Gabor est linéaire, on a pour deux signaux $(x, y) \in (L^2(\mathbb{R}))^2$

$$|G_{x_1+x_2}^g|^2 = |G_{x_1}^g|^2 + |G_{x_2}^g|^2 + 2\Re\left(G_{x_1}^g \overline{G_{x_2}^g}\right). \quad (2.31)$$

Pour le spectrogramme, les interférences sont donc négligeables lorsque les composantes sont suffisamment éloignées dans le plan temps-fréquence. En revanche elle deviennent importantes lorsque les composantes sont très proches, ce qui se traduit dans la pratique par une limitation de la résolution temps-fréquence.

C'est en fait le spectrogramme qui est habituellement représenté lorsque l'on construit des visualisations sonographiques à partir de la transformée de Gabor. Il est de plus en général représenté en utilisant une échelle en décibel (dB) afin de faire apparaître les énergies des composantes de manière mieux adaptée à la perception auditive (la perception de niveau subjectif en fonction du niveau de pression acoustique suit approximativement une loi logarithmique qu'une loi linéaire).

Pour les applications visées dans cette thèse, c'est ce mode de visualisation qui a été retenu. La visualisation graphique présentée à l'utilisateur est construite à partir du spectrogramme sur une échelle en dB, c'est à dire à partir de la fonction $10 \log_{10}(|G_x^g|^2)$. L'information de phase de la transformée n'est donc pas représentée. Les transformations des sons sont cependant effectuées en utilisant directement les coefficients complexes de la transformée de Gabor.

Exemples de transformées de Gabor

Pour étudier le comportement de la transformée, on donne quelques exemples de transformées pour des signaux académiques.

Remarque 2.5 *Certains signaux dans les exemples suivants ne font pas partie de $L^2(\mathbb{R})$, cependant comme indiqué dans la remarque 2.2, il est possible de définir la transformée de Gabor pour ces signaux. Nous avons choisi de les faire figurer car*

ils constituent des cas extrêmes permettant de bien illustrer le comportement de la transformée.

Exemple 2.2 Pour une impulsion de Dirac $x(t) = \delta(t)$, on a la transformée :

$$G_x^g(\tau, \nu) = \bar{g}(-\tau) e^{i2\pi\nu\tau}. \quad (2.32)$$

Sur la transformée, l'énergie du signal est donc concentrée autour de l'instant nul dans une bande verticale couvrant toutes les fréquences et dont la forme est donnée par la fenêtre g . La figure 2.4 donne un exemple de représentation obtenue pour une impulsion avec une fenêtre gaussienne.

On voit donc que la localisation de l'énergie sur la représentation dépend directement de la forme de la fenêtre. En particulier plus la fenêtre est courte, plus l'énergie est bien localisée autour de l'instant nul pour l'impulsion.

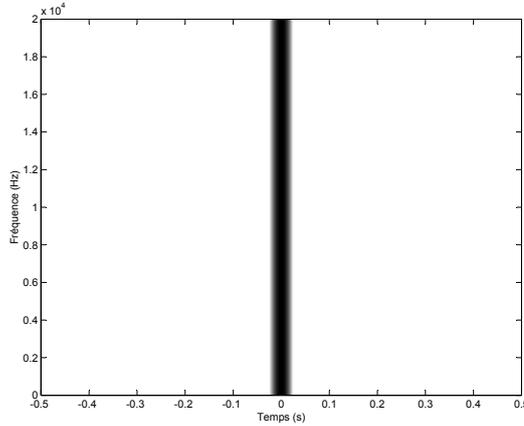


FIG. 2.4 – Exemple de transformée de Gabor pour une impulsion

Exemple 2.3 Pour un sinus, $x(t) = \sin(2\pi\nu_0 t + \varphi_0)$, on a :

$$G_x^g(\tau, \nu) = \frac{1}{2i} \left(e^{i(2\pi\nu_0\tau + \varphi_0)} \hat{g}(\nu_0 - \nu) - e^{-i(2\pi\nu_0\tau + \varphi_0)} \hat{g}(-\nu_0 - \nu) \right). \quad (2.33)$$

Pour une fréquence ν_0 suffisamment élevée, le terme $\hat{g}(-\nu_0 - \nu)$ est négligeable pour les fréquences positives, et l'énergie du signal apparaît dans cette région concentrée autour de la fréquence ν_0 dans une bande horizontale dont la forme est donnée par \hat{g} . La figure 2.5 donne un exemple de représentation obtenue pour un sinus avec une fenêtre gaussienne.

On voit donc que la localisation de l'énergie sur la représentation dépend directement de la transformée de Fourier de la fenêtre. En particulier plus la fenêtre est longue et plus l'énergie est bien localisée autour de la fréquence ν_0 pour le sinus. On a donc un comportement dual de celui obtenu pour une impulsion.

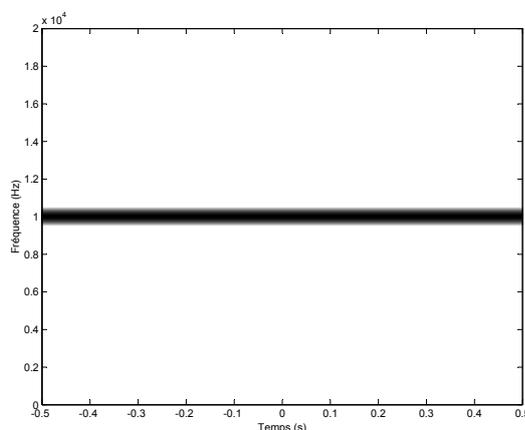


FIG. 2.5 – Exemple de transformée de Gabor pour un sinus

Exemple 2.4 Pour une fonction de la forme $x(t) = A(t) \cos \left(2\pi \int_0^t f(u) du + \phi \right)$, en supposant que l'amplitude $A(t)$ et la fréquence instantanée $f(t)$ varient suffisamment lentement avec t , on a l'approximation, pour ν positif (voir [3] pour plus de détails) :

$$G_x^g(\tau, \nu) \approx \frac{1}{2} A(\tau) e^{i(2\pi \int_0^\tau f(u) du + \phi)} \overline{\hat{g}(f(\tau) - \nu)}. \quad (2.34)$$

Lorsque les variations sont suffisamment lentes, la transformée de Gabor se comporte donc localement autour d'un instant τ comme pour un sinus de fréquence $f(\tau)$ et d'amplitude $A(\tau)$, et l'énergie sur la représentation sera localisée autour d'une crête (*ridge* en anglais) qui suit la fréquence instantanée. Un exemple de représentation pour ce type de composantes est donné sur la figure 2.6. L'amplitude $A(t)$ du signal est une gaussienne et la fréquence instantanée $f(t)$ varie sinusoidalement autour d'une fréquence de 10000Hz.

Les transformées obtenues pour l'impulsion et le sinus font apparaître que la localisation des composantes sur la représentation dépend fortement de la fenêtre obtenue, et en particulier de sa longueur (où un paramètre équivalent mesurant sa localisation temporelle si elle n'est pas à support fini). On voit en particulier que des fenêtres de courte durée permettent une meilleure localisation de composantes très localisées dans le temps, le cas extrême de localisation temporelle correspondant à l'impulsion. À l'inverse, une grande fenêtre permet une meilleure localisation des composantes bien localisées dans l'espace de Fourier, le cas extrême correspondant au sinus. Les fenêtres peuvent être en conséquence plus ou moins bien adaptées à la représentation d'un signal en fonction des caractéristiques du signal et de la fenêtre.

Le choix de la fenêtre d'analyse dans la transformée de Fourier apparaît ainsi comme un problème particulièrement important dans le cadre du traitement temps-

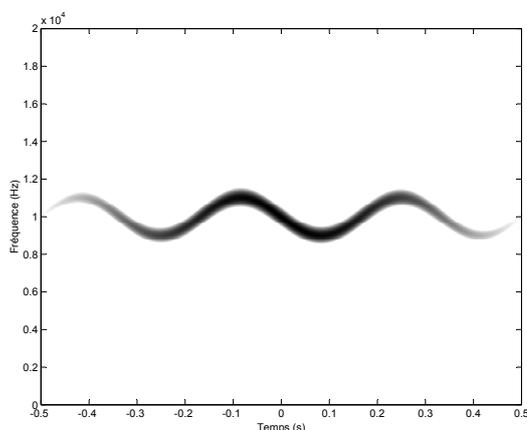


FIG. 2.6 – Exemple de transformée de Gabor pour une sinusoïde d’amplitude et de fréquence variant lentement dans le temps

fréquence des signaux. Le choix de la fenêtre conditionne très fortement le type de composantes bien identifiables sur la transformée et donc les traitements que peut effectuer l’utilisateur sur cette transformée.

Exemple 2.5 *L’exemple représenté sur la figure 2.7 montre l’importance d’un choix bien adapté de la fenêtre sur un signal constitué d’une somme de deux chirps linéaires. On voit qu’il existe une taille optimale permettant de bien résoudre les deux composantes, alors qu’avec une taille de fenêtre trop petite ou trop grande, les deux chirps ne sont pas résolus.*

Modification de signaux à partir de leur transformée de Gabor

La transformée de Gabor est très couramment utilisée pour le traitement des signaux, et en particulier dans le domaine audionumérique, et ceci depuis de nombreuses années, comme en témoignent les références [39] [35]. Elle est généralement désigné dans ce contexte sous son nom de transformée de Fourier à court terme, et également sous le nom de vocodeur de phase, qui indique en général que la transformée est utilisée pour réaliser un type spécifique de traitement, l’élongation temporelle du signal sans changement perceptif de fréquence (ou le traitement “dual”, la transposition fréquentielle sans changement de durée).

Il ne s’agit pas du type de traitement que nous souhaitons étudier dans notre contexte. Nous nous intéressons à un autre type de traitement, qui peut être interprété comme un “filtrage variant dans le temps” et qui consiste à utiliser un type particulier d’opérateur : les multiplicateurs de Gabor [17]. Il s’agit en fait simplement

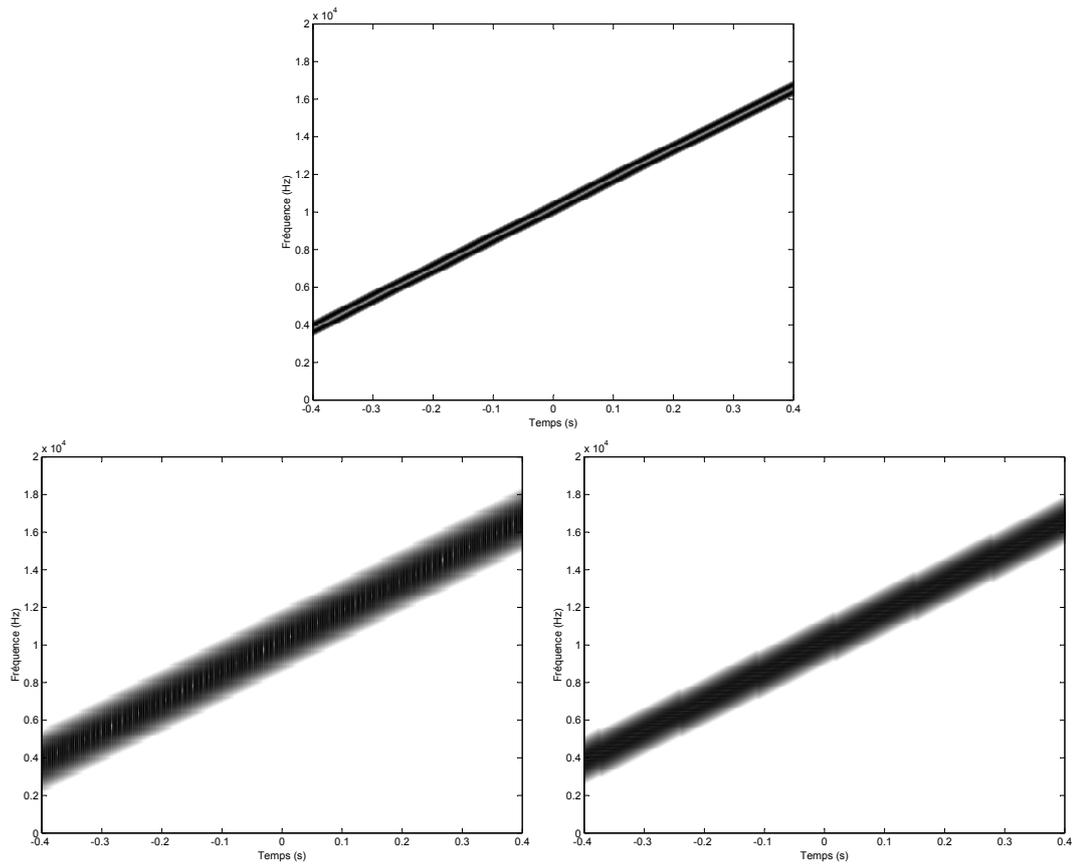


FIG. 2.7 – Influence de la taille de fenêtre sur les représentations obtenues avec la transformée de Gabor : pour un signal composé d'une somme de deux chirps, en haut, transformée obtenue avec une taille de fenêtre moyenne, en bas à gauche, transformée obtenue avec une petite taille de fenêtre, en bas à droite, transformée obtenue avec une grande taille de fenêtre

de multiplier les coefficients de la transformée de Gabor par une fonction du temps et de la fréquence, puis de reconstruire le signal à partir des coefficients modifiés avec la formule d'inversion de la transformée.

Avec la formulation continue, pour modifier un signal $x \in L^2(\mathbb{R})$, on introduit donc une fonction bornée $M \in L^\infty(\mathbb{R}^2)$ et on reconstruit le signal :

$$x_{mod} = \iint M(\tau, \nu) G_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu. \quad (2.35)$$

Il faut noter que les possibilités de traitement en utilisant les multiplicateurs de Gabor sont contraintes par la redondance de la transformée de Gabor.

Remarque 2.6 *En effet, l'espace des fonctions de $L^2(\mathbb{R}^2)$ qui sont des transformées de Gabor d'un signal de $L^2(\mathbb{R})$ ne constitue qu'un sous-espace de $L^2(\mathbb{R}^2)$ qui est "plus petit" (mais qui reste en dimension infinie) que $L^2(\mathbb{R}^2)$. Ainsi, la fonction $M(\tau, \nu) G_x^g(\tau, \nu)$ n'est généralement pas la transformée de Gabor d'un signal et la transformée de Gabor du signal reconstruit x_{mod} ne vaut pas exactement $M(\tau, \nu) G_x^g(\tau, \nu)$ (mais elle s'en approche, pour des choix "raisonnables" de M).*

Pour qu'une fonction $H \in L^2(\mathbb{R}^2)$ soit la transformée d'un signal de $L^2(\mathbb{R})$, elle doit satisfaire la relation à noyau reproduisant traduisant la redondance de la transformée, c'est à dire : il existe $g \in L^2(\mathbb{R})$ tel que

$$\forall (\tau, \nu) \in \mathbb{R}^2, \quad H(\tau, \nu) = \iint e^{i2\pi\nu'(\tau-\tau')} G_g^g(\tau - \tau', \nu - \nu') H(\tau', \nu') d\tau' d\nu'. \quad (2.36)$$

On ne peut ainsi pas obtenir n'importe quelle modification de la transformée de Gabor du signal, les transformées obtenues après reconstruction étant lissées par le noyau reproduisant. L'exemple suivant permet de voir l'effet du noyau reproduisant en pratique.

Exemple 2.6 *Sur un bruit blanc, noté b , dont la transformée de Gabor G_b^g est représentée en haut de la figure 2.8, on souhaite appliquer un multiplicateur M qui annule les coefficients de la transformée dans un rectangle donné. On souhaite ainsi obtenir la représentation du milieu de la figure 2.8 qui représente le module au carré de la fonction $M(\tau, \nu) G_b^g(\tau, \nu)$. En calculant la transformée de Gabor du signal reconstruit à partir de $M(\tau, \nu) G_b^g(\tau, \nu)$, on obtient la représentation du bas de la figure 2.8. On voit que la transformée obtenue est une version lissée de la transformée souhaitée. Les modifications qui pourront effectivement être réalisées à l'aide de la transformée de Gabor seront limitées par les caractéristiques du noyau reproduisant (qui dépend lui-même uniquement de la fenêtre d'analyse).*

Les modifications des sons à partir des transformées présentées dans la suite se font avec le même type d'approche (la multiplication des coefficients de la transformée). On obtient également des limitations sur les modifications du même type, car les transformées proposées sont redondantes.

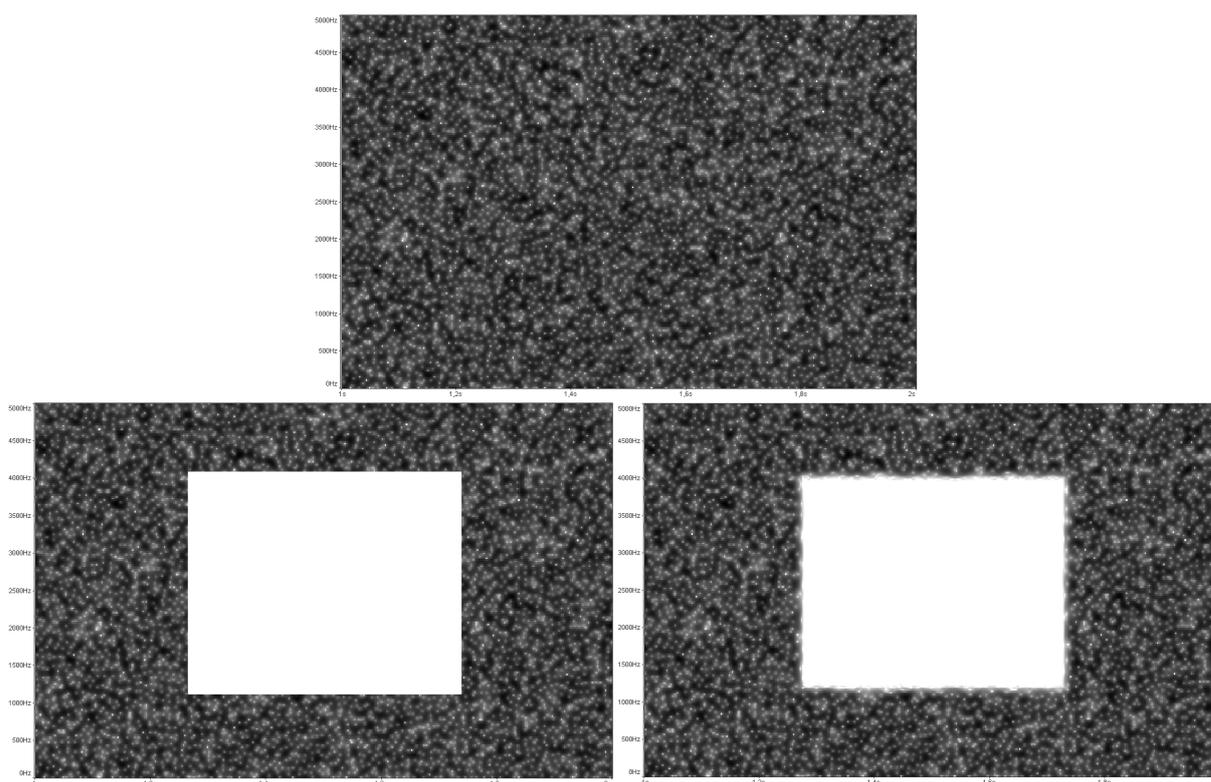


FIG. 2.8 – Effet du lissage par le noyau reproduisant : en haut, transformée de Gabor du bruit blanc original, au milieu, transformée souhaitée, obtenue en annulant les coefficients de la transformée dans une zone rectangulaire, en bas, transformée de Gabor du signal reconstruit à partir de la transformée souhaitée

Discrétisation

Pour une utilisation pratique, il est nécessaire d'utiliser une version discrétisée de la transformée. Le cadre d'étude de la discrétisation de la transformée de Gabor est la théorie des repères⁵ (*frame* en anglais).

La définition d'un repère est la suivante : une famille de fonctions $(\phi_n)_{n \in \mathcal{N}}$ d'un espace de Hilbert \mathcal{E} est un repère de \mathcal{E} s'il existe des constantes A et B réelles vérifiant $0 < A \leq B$ telles que

$$\forall x \in \mathcal{E}, \quad A\|x\|^2 \leq \sum_{n \in \mathcal{N}} |\langle x, \phi_n \rangle|^2 \leq B\|x\|^2. \quad (2.37)$$

L'intérêt des repères est qu'ils permettent la reconstruction du signal à partir de sa décomposition sur les éléments du repère. En effet si (2.37) est satisfaite, il existe un repère $(\tilde{\phi}_n)_{n \in \mathcal{N}}$ de \mathcal{E} (qui n'est généralement pas unique), appelé repère dual, tel que :

$$\forall x \in \mathcal{E}, \quad x = \sum_{n \in \mathcal{N}} \langle x, \phi_n \rangle \tilde{\phi}_n. \quad (2.38)$$

On a également la relation :

$$x = \sum_{n \in \mathcal{N}} \langle x, \tilde{\phi}_n \rangle \phi_n. \quad (2.39)$$

Si $A = B$, on dit que le repère est strict (*tight frame*). Dans ce cas on peut utiliser le même repère pour la décomposition et la reconstruction avec la relation :

$$x = \frac{1}{A} \sum_{n \in \mathcal{N}} \langle x, \phi_n \rangle \phi_n, \quad (2.40)$$

et on retrouve pour $A = 1$ la forme de l'équation (2.8).

On montre, par exemple dans [8], que sous certaines conditions, la discrétisation de la transformée de Gabor aboutit à la construction de repères qui peuvent être utilisés pour le traitement de signaux discrets. L'étude du lien entre la version continue et la version discrète de la transformée est décrit dans [28].

Une présentation détaillée de la théorie des repères de Gabor est donnée dans [23] [18] [19].

Une description des repères de Gabor dans le cadre discret pour les applications au traitement du signal audionumérique est présentée dans [13]. J'ai également décrit la version discrète de la transformée de Gabor pour les signaux discrets de dimension finie dans [25].

⁵On désigne également les repères sous le nom de structures obliques.

Dans le cas discret en dimension finie, on peut calculer explicitement le repère dual qui est obtenu par translation en temps et en fréquence d'une fenêtre duale unique. On assure ainsi la reconstruction exacte du signal à partir de sa décomposition. On peut également construire des repères stricts et donc utiliser des décompositions de Gabor discrètes de la forme présentés dans l'équation (2.8), bien appropriées aux applications de modification des signaux.

2.1.4 Transformée en ondelettes

Nous ne donnons ici que la définition de la transformée en ondelettes, car nous ne l'utiliserons pas directement pour le traitement des sons. Elle sera cependant utilisée pour la construction d'autres transformées dans la partie 2.2.7. Pour plus de détail sur la transformée en ondelettes, on peut consulter par exemples les références [7] [31].

Pour simplifier la présentation de la transformée en ondelettes, nous nous plaçons dans cette section dans le cadre de l'étude des signaux analytiques, c'est à dire des signaux de l'espace de Hardy $H^2(\mathbb{R})$ défini par :

$$H^2(\mathbb{R}) = \{f \in L^2(\mathbb{R}) : \hat{f}(\xi) = 0, \xi \leq 0\}. \quad (2.41)$$

Cette limitation n'est pas gênante car les signaux qui nous intéressent sont à valeur réelle, et il est possible de faire facilement le lien entre les signaux à valeur réelle de $L^2(\mathbb{R})$ et $H^2(\mathbb{R})$ en utilisant la transformée de Hilbert H . Elle permet d'associer à un signal $x \in L^2(\mathbb{R})$ à valeur réelle un signal analytique $Z_x \in H^2(\mathbb{R})$ vérifiant $Z_x(t) = [I + iH]f(t)$, avec I l'opérateur identité. Le signal x peut être facilement obtenu à partir de Z_x selon la relation $x(t) = \Re(Z_x(t))$.

Définition

Les atomes de la transformée en ondelettes sont obtenus par translation temporelle et dilatation d'une unique fonction que l'on désigne sous le nom d'ondelette mère ou d'ondelette d'analyse, notée ici h . La construction des atomes se fait en conséquence de la façon suivante :

soit h une fonction de $H^2(\mathbb{R})$ vérifiant la condition d'admissibilité

$$\int_0^{+\infty} |\hat{h}(\nu)|^2 \frac{d\nu}{\nu} = 1, \quad (2.42)$$

la famille d'atomes $(h_{\tau a})_{\tau, a \in \mathbb{R} \times \mathbb{R}_+^*}$ est définie par

$$h_{\tau a} = \frac{1}{a} h\left(\frac{t - \tau}{a}\right). \quad (2.43)$$

La transformée en ondelettes continue T_x^h d'un signal x de $H^2(\mathbb{R})$ est obtenue par produit scalaire du signal sur les atomes de la décomposition, selon la relation :

$$T_x^h(\tau, a) = \langle x, h_{\tau a} \rangle = \frac{1}{a} \int_{-\infty}^{+\infty} x(t) \bar{h}\left(\frac{t-\tau}{a}\right) dt \quad (2.44)$$

Interprétations

Comme pour la transformée de Gabor, plusieurs interprétations de la transformée en ondelettes sont possibles :

Décomposition atomique : La première est l'interprétation comme décomposition atomique sur les atomes $h_{\tau a}$. On utilise dans la pratique une ondelette h dont l'énergie est bien localisée autour de l'instant 0 et dont la transformée de Fourier concentre l'énergie autour d'une fréquence centrale ν_0 avec une largeur de bande $\Delta\nu$. Compte tenu de la définition des atomes $h_{\tau a}$, temporellement, l'énergie de l'atome est localisée autour de l'instant τ , en se concentrant dans une région qui fait a fois la taille de celle obtenue pour h . Ainsi, plus la valeur d'échelle a est grande, plus l'énergie de l'atome est étalée dans le temps. En revanche, on a un effet inverse en fréquence. En effet, la transformée de Fourier de l'atome $h_{\tau a}$ vaut :

$$\hat{h}_{\tau a}(\xi) = e^{-i2\pi\xi\tau} \hat{h}(a\xi). \quad (2.45)$$

Pour une échelle a , la fréquence centrale vaut donc $\frac{\nu_0}{a}$ et la largeur de bande $\frac{\Delta\nu}{a}$. Ainsi, plus la fréquence centrale de l'atome est faible, c'est à dire plus la valeur d'échelle a est grande, plus la concentration de l'énergie en fréquence est grande. La décomposition se fait donc sur une famille d'atomes dont les caractéristiques de localisation dépendent de la fréquence : tous les atomes occupent la même "aire" temps-fréquence, mais lorsque la fréquence augmente, leur localisation temporelle s'améliore alors que leur localisation fréquentielle diminue. La figure 2.9 présente une représentation symbolique des caractéristiques de localisation temps-fréquence des atomes.

Banc de filtres : On peut également interpréter la transformée comme la sortie d'un banc de filtres continu. Pour chaque échelle a , en définissant $r_a(t) = \frac{1}{a} \bar{h}\left(-\frac{t}{a}\right)$, on peut écrire la transformée sous la forme du signal x filtré par le filtre de réponse impulsionnelle r_a , selon la relation :

$$G_x^g(\tau, \nu) = \int x(t) r_a(\tau - t) dt. \quad (2.46)$$

La réponse en fréquence du filtre vaut :

$$\hat{r}_\nu(\xi) = -\overline{\hat{h}(a\nu)}. \quad (2.47)$$

Donc pour une échelle a , la fréquence centrale du filtre vaut $\frac{\nu_0}{a}$ et la largeur de bande $\frac{\Delta\nu}{a}$. La transformée peut donc être interprétée comme une analyse en banc de filtres utilisant des filtres à facteur de qualité (rapport de la largeur de bande sur la fréquence centrale) constant.

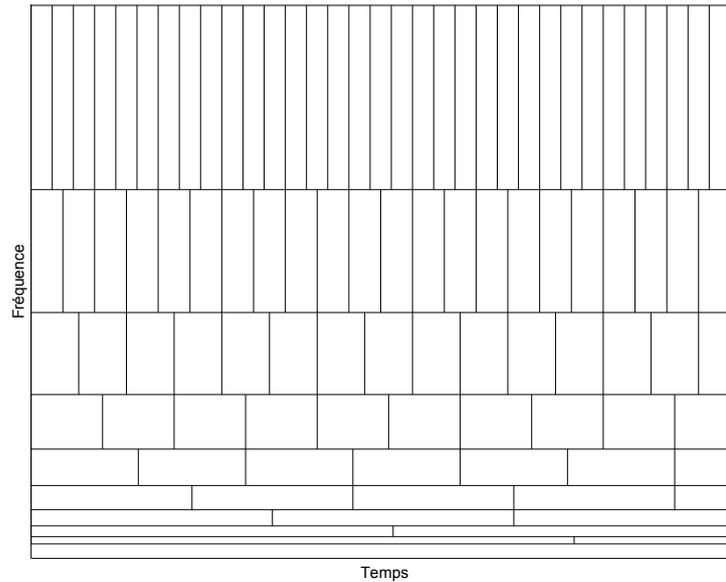


FIG. 2.9 – Pavage temps-fréquence pour la transformée en ondelettes

2.2 Représentations adaptées

Les deux exemples de représentations atomiques que nous avons décrits en 2.1.3 et 2.1.4 sont basés sur une construction simple et relativement rigide : une série de transformations géométriques simples est appliquée à un atome père. Cette construction a pour effet de fixer des propriétés de localisation temps-fréquence communes à tous les atomes de la famille.

Or les signaux réels possèdent souvent une forme d’“instationnarité” (temporelle ou fréquentielle) qui rend cette rigidité pénalisante.

Nous allons voir dans ce qui suit qu’il est possible d’échapper (dans une certaine mesure) à cette rigidité, et d’introduire des familles d’atomes temps-fréquence de résolution variable, tout en préservant la propriété essentielle : l’existence d’une formule d’inversion pour la transformation.

2.2.1 Transformées covariantes par translation fréquentielle

Pour le traitement des signaux présentant une “instationnarité” temporelle, il est souhaitable, pour obtenir une bonne représentation, de pouvoir adapter la résolution temporelle des atomes au cours du temps. La transformée de Gabor instationnaire présentée ici permet cette adaptation en assurant l’inversibilité de la transformée.

Définition

Pour permettre aux atomes de varier au cours du temps, on n’utilise plus une unique fenêtre mère comme dans la transformée de Gabor, mais on introduit une famille de fenêtres de $L^2(\mathbb{R})$ $(g_\tau)_{\tau \in \mathbb{R}}$ qui dépendent du paramètre temporel τ en vérifiant la condition d’admissibilité :

$$\forall t \in \mathbb{R}, \quad \int |g_\tau(t)|^2 d\tau = 1. \quad (2.48)$$

Les atomes $g_{\tau\nu}$ sont alors obtenus par translation fréquentielle des fenêtres g_τ , selon la relation :

$$g_{\tau\nu}(t) = g_\tau(t) e^{i2\pi\nu(t-\tau)}. \quad (2.49)$$

La transformée de Gabor instationnaire H_x^g d’un signal x de $L^2(\mathbb{R})$ est obtenue par produit scalaire du signal sur les atomes de la décomposition, selon la relation :

$$H_x^g(\tau, \nu) = \langle x, g_{\tau\nu} \rangle = \int x(t) \overline{g_\tau(t)} e^{-i2\pi\nu(t-\tau)} dt = e^{i2\pi\nu\tau} \int \hat{x}(\xi) \overline{\hat{g}_\tau(\xi - \nu)} d\xi. \quad (2.50)$$

Remarque 2.7 *Il existe bien des familles de fenêtres vérifiant la condition 2.48. En effet, on peut par exemple remarquer que la transformée de Gabor classique est un cas particulier de la transformée de Gabor instationnaire, puisqu’en définissant pour la fenêtre $g \in L^2(\mathbb{R})$ vérifiant $\int |g(t)|^2 dt = 1$ les fonctions g_τ par :*

$$\forall t \in \mathbb{R}, \quad g_\tau(t) = g(t - \tau), \quad (2.51)$$

on a bien

$$\forall t \in \mathbb{R}, \quad \int |g_\tau(t)|^2 d\tau = \int |g(t - \tau)|^2 d\tau = 1. \quad (2.52)$$

Une méthode de construction de familles de fonctions vérifiant la condition est proposée en partie 2.2.5.

Interprétations

Décomposition atomique : Pour que la variable temporelle τ conserve une interprétation comparable au paramètre de translation temporelle de la transformée de Gabor classique, on choisit pour chaque valeur de τ une fonction g_τ dont l'énergie est bien localisée dans le temps autour de l'instant τ .

On choisit également d'utiliser des fonctions g_τ passe-bas, afin de localiser l'énergie des fenêtres en fréquence autour de l'origine.

Typiquement, g_τ sera une fenêtre d'analyse spectrale translatée temporellement d'une valeur τ , comme dans le cas de la transformée de Gabor, avec cependant maintenant la possibilité d'avoir pour les différentes valeurs de τ des tailles et des formes de fenêtres différentes.

Les caractéristiques de localisation temps-fréquence des atomes varient donc avec le temps, mais ne varient pas, pour un instant τ donné, avec la fréquence. La figure 2.10 présente une représentation symbolique des caractéristiques de localisation temps-fréquence des atomes.

Analyse de Fourier à court terme : En définissant le signal $x_\tau(t) = x(t + \tau) \overline{g_\tau}(t + \tau)$, c'est à dire en fenêtrant le signal x avec la fenêtre $\overline{g_\tau}$, la transformée de Gabor s'écrit comme la transformée de Fourier du signal fenêtré, selon la relation :

$$\hat{x}_\tau(\nu) = \int x_\tau(t) e^{-i2\pi\nu t} dt \quad (2.53)$$

$$= \int x(t + \tau) \overline{g_\tau}(t + \tau) e^{-i2\pi\nu t} dt \quad (2.54)$$

$$= \int x(t') \overline{g_\tau}(t') e^{-i2\pi\nu(t'-\tau)} dt' \quad (2.55)$$

$$= H_x^g(\tau, \nu). \quad (2.56)$$

Comme dans la transformée de Gabor classique, pour un instant τ la transformée peut donc être interprétée comme un spectre à court terme du signal, la transformée de Gabor instationnaire permettant ainsi d'analyser l'évolution spectrale du signal au cours du temps, avec cependant une analyse dont les caractéristiques varient dans temps, d'où la dénomination instationnaire de la transformée.

Propriétés

Propriété 2.6 (Linéarité) *La transformée de Gabor instationnaire est linéaire, c'est à dire :*

$$\forall(\lambda, \mu) \in \mathbb{C}^2, \forall(x, y) \in (L^2(\mathbb{R}))^2, \quad H_{\lambda x + \mu y}^g = \lambda H_x^g + \mu H_y^g. \quad (2.57)$$

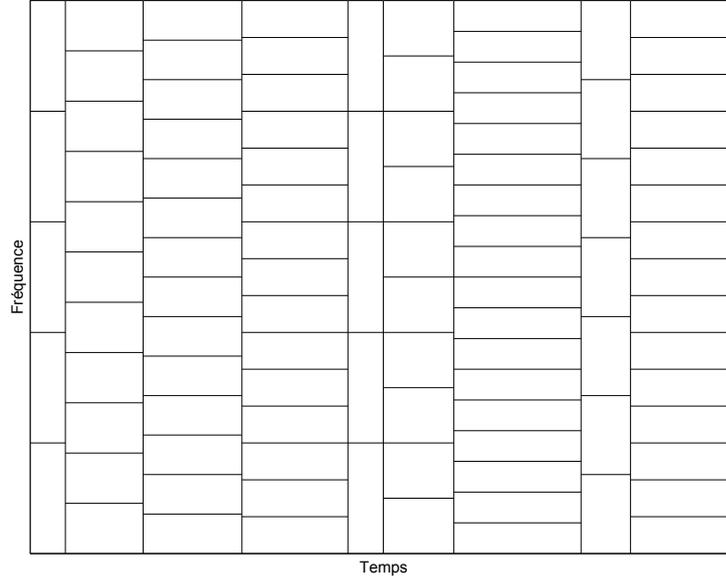


FIG. 2.10 – Exemple de pavage temps-fréquence pour la transformée de Gabor instationnaire

Propriété 2.7 (Covariance) *La transformée de Gabor instationnaire est covariante par translation fréquentielle (modulation), c'est à dire :*

en définissant, pour une translation fréquentielle d'une fréquence $\nu_0 \in \mathbb{R}$ d'un signal $x \in L^2(\mathbb{R})$, le signal modulé x_{ν_0} tel que $x_{\nu_0}(t) = x(t) e^{i2\pi\nu_0 t}$, on a

$$\forall(\tau, \nu) \in \mathbb{R}^2, \quad H_{x_{\nu_0}}^g(\tau, \nu) = H_x^g(\tau, \nu - \nu_0). \quad (2.58)$$

Propriété 2.8 (Isométrie) *La transformée de Gabor instationnaire est une isométrie, c'est à dire qu'elle conserve le produit scalaire, par la relation :*

$$\forall(x, y) \in (L^2(\mathbb{R}))^2, \quad \langle H_x^g, H_y^g \rangle = \langle x, y \rangle. \quad (2.59)$$

En effet, pour x et y dans $L^2(\mathbb{R})$, on a :

$$\langle H_x^g, H_y^g \rangle = \iint H_x^g(\tau, \nu) \overline{H_y^g(\tau, \nu)} d\tau d\nu \quad (2.60)$$

$$= \iiint x(t) \overline{y(u)} g_\tau(u) \overline{g_\tau(t)} e^{i2\pi\nu(u-t)} dt du d\tau d\nu \quad (2.61)$$

$$= \iiint x(t) \overline{y(u)} g_\tau(u) \overline{g_\tau(t)} \delta(u-t) dt du d\tau \quad (2.62)$$

$$= \int x(t) \overline{y(t)} \left(\int |g_\tau(t)|^2 d\tau \right) dt. \quad (2.63)$$

Comme la famille $(g_\tau)_{\tau \in \mathbb{R}}$ vérifie la condition d'admissibilité 2.48, la propriété d'isométrie est bien vérifiée.

Propriété 2.9 (Inversion) *La transformée de Gabor instationnaire est inversible selon la relation suivante, dans laquelle l'égalité est prise au sens du produit scalaire dans $L^2(\mathbb{R})$:*

$$\forall x \in L^2(\mathbb{R}), \quad x = \iint H_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu. \quad (2.64)$$

En effet, pour $(x, y) \in (L^2(\mathbb{R}))^2$, en remplaçant H_y^g par sa définition, on a

$$\langle H_x^g, H_y^g \rangle = \iiint H_x^g(\tau, \nu) g_{\tau\nu}(t) \bar{y}(t) dt d\tau d\nu \quad (2.65)$$

$$= \left\langle \iint H_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, y \right\rangle \quad (2.66)$$

Comme H est une isométrie, on a

$$\forall (x, y) \in (L^2(\mathbb{R}))^2, \quad \langle x, y \rangle = \left\langle \iint H_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, y \right\rangle \quad (2.67)$$

Propriété 2.10 (Conservation de l'énergie) *La propriété d'isométrie implique la conservation d'énergie par la transformée, selon la relation*

$$\int |x(t)|^2 dt = \iint |H_x^g(\tau, \nu)|^2 d\tau d\nu. \quad (2.68)$$

Discrétisation

J'ai décrit une version discrète en dimension finie de la transformée dans [25]. Cette version est utilisable pour une implémentation effective.

Il est également possible de construire une version semi-discrète de la transformée en conservant des fonctions de variables continues, comme présenté ci-dessous.

2.2.2 Version semi-discrète

L'élément clé dans la démonstration des propriétés de la transformée est la covariance par rapport à la translation suivant la variable continue ν . L'autre variable, la variable temporelle τ , peut être discrétisée en conservant les bonnes propriétés de la transformée.

On introduit alors une famille dénombrable de fenêtres $(h_m)_{m \in \mathbb{Z}}$ à laquelle on impose cette fois de vérifier la condition d'admissibilité :

$$\forall t \in \mathbb{R}, \quad \sum_{m \in \mathbb{Z}} |h_m(t)|^2 = 1. \quad (2.69)$$

On note τ_m le centre de gravité de h_m , et on définit la famille semi-continue d'atomes $(h_{m\nu})_{(m,\nu)\in\mathbb{Z}\times\mathbb{R}}$ par translation fréquentielle des fenêtres h_m , selon la relation :

$$\forall t \in \mathbb{R}, \quad h_{m\nu}(t) = h_m(t) e^{i2\pi\nu(t-\tau_m)}. \quad (2.70)$$

On introduit l'espace \mathcal{E} vérifiant :

$$\mathcal{E} = \{X : \mathbb{Z} \times \mathbb{R} \rightarrow \mathbb{C} : \sum_{m \in \mathbb{Z}} \int |X(m, \nu)|^2 d\nu < +\infty\}. \quad (2.71)$$

L'espace \mathcal{E} a une structure d'espace de Hilbert, grâce au produit scalaire suivant :

$$\forall (X, Y) \in \mathcal{E}, \quad \langle X, Y \rangle = \sum_{m \in \mathbb{Z}} \int X(m, \nu) \overline{Y(m, \nu)} d\nu. \quad (2.72)$$

On définit alors la transformée du signal x de $L^2(\mathbb{R})$ comme la fonction I_x^h vérifiant :

$$I_x^h(m, \nu) = \int x(t) \overline{h_m(t)} e^{-i2\pi\nu(t-\tau_m)} dt, \quad (2.73)$$

Si (2.69) est vérifiée, alors $I_x^h \in \mathcal{E}$, et les propriétés de la transformée sont conservées, les démonstrations s'effectuant de manière similaire en remplaçant l'intégrale par rapport à la variable τ par une somme sur l'indice m .

En particulier, la formule d'inversion s'écrit :

$$\forall x \in L^2(\mathbb{R}), \quad x = \sum_{m \in \mathbb{Z}} \int I_x^h(m, \nu) h_{m\nu} d\nu. \quad (2.74)$$

On verra en 2.2.5 comment des familles vérifiant les conditions (2.48) et (2.69) peuvent être construites, mais on va d'abord comparer le cas des transformées covariantes par translation temporelle.

2.2.3 Transformées covariantes par translation temporelle

Définition

Pour permettre aux caractéristiques des atomes de varier avec la fréquence, on introduit une famille de fonctions $L^2(\mathbb{R})$ $(g_\nu)_{\nu \in \mathbb{R}}$ qui dépendent du paramètre fréquentiel ν et dont les transformées de Fourier vérifient la condition d'admissibilité :

$$\forall \xi \in \mathbb{R}, \quad \int |\hat{g}_\nu(\xi)|^2 d\nu = 1. \quad (2.75)$$

Les atomes $g_{\tau\nu}$ sont alors obtenus par translation temporelle des fenêtres g_ν , selon la relation :

$$g_{\tau\nu}(t) = g_\nu(t - \tau). \quad (2.76)$$

On définit la transformée K_x^g d'un signal x de $L^2(\mathbb{R})$ par produit scalaire du signal sur les atomes de la décomposition, selon la relation :

$$K_x^g(\tau, \nu) = \langle x, g_{\tau\nu} \rangle = \int x(t) \overline{g_\nu(t - \tau)} dt = \int \hat{x}(\xi) \overline{\hat{g}_\nu(\xi)} e^{i2\pi\xi\tau} d\xi. \quad (2.77)$$

Interprétations

Décomposition atomique : Pour que la variable temporelle ν conserve une interprétation comparable au paramètre de translation fréquentielle de la transformée de Gabor classique, on choisit pour chaque valeur de ν une fonction g_ν telle que sa transformée de Fourier concentre l'énergie autour de la fréquence ν .

On choisit également d'utiliser des fonctions g_ν bien localisée dans le temps autour de l'instant nul.

Les caractéristiques de localisation temps-fréquence des atomes varient donc avec la fréquence, mais ne varient pas, pour une fréquence τ donné, avec le temps. La figure 2.11 présente une représentation symbolique des caractéristiques de localisation temps-fréquence des atomes.

Banc de filtres : On peut interpréter la transformée comme la sortie d'un banc de filtres continu. Pour chaque fréquence ν , en définissant $r_\nu(t) = \overline{g_\nu(-t)}$, on peut écrire la transformée sous la forme du signal x filtré par le filtre de réponse impulsionnelle r_ν , selon la relation :

$$G_x^g(\tau, \nu) = \int x(t) r_\nu(\tau - t) dt. \quad (2.78)$$

La transformée peut donc être interprétée comme une analyse en banc de filtres utilisant des filtres de largeur de bande variable avec les fréquences.

Propriétés

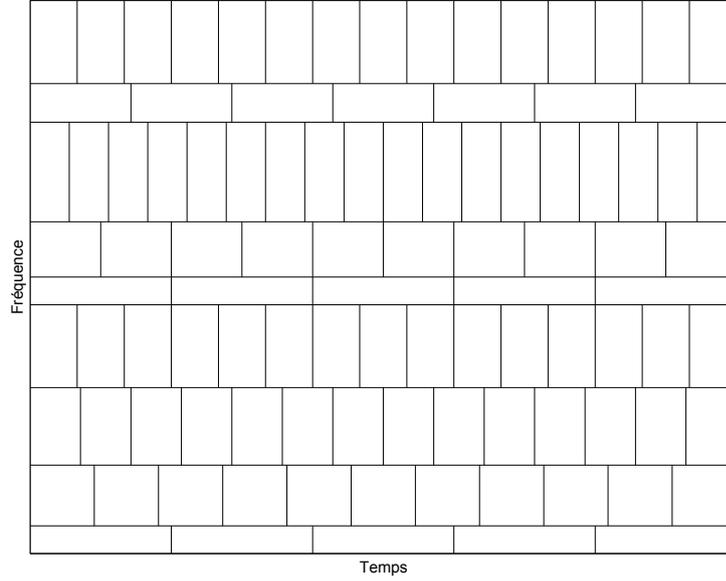
Propriété 2.11 (Linéarité) *La transformée est linéaire, c'est à dire :*

$$\forall(\lambda, \mu) \in \mathbb{C}^2, \forall(x, y) \in (L^2(\mathbb{R}))^2, \quad K_{\lambda x + \mu y}^g = \lambda K_x^g + \mu K_y^g. \quad (2.79)$$

Propriété 2.12 (Covariance) *La transformée est covariante par translation temporelle, c'est à dire :*

en définissant, pour une translation temporelle d'une durée $\tau_0 \in \mathbb{R}$ d'un signal $x \in L^2(\mathbb{R})$, le signal translaté x_{τ_0} tel que $x_{\tau_0}(t) = x(t - \tau_0)$, on a

$$\forall(\tau, \nu) \in \mathbb{R}^2, \quad K_{x_{\tau_0}}^g(\tau, \nu) = K_x^g(\tau - \tau_0, \nu). \quad (2.80)$$

FIG. 2.11 – Exemple de pavage temps-fréquence pour la transformée K_x^g

Propriété 2.13 (Isométrie) *La transformée est une isométrie, c'est à dire qu'elle conserve le produit scalaire, par la relation :*

$$\forall (x, y) \in (L^2(\mathbb{R}))^2, \quad \langle K_x^g, K_y^g \rangle = \langle x, y \rangle. \quad (2.81)$$

En effet, pour x et y dans $L^2(\mathbb{R})$, en utilisant la formulation de la définition à partir des transformées de Fourier des fonctions, on a :

$$\langle K_x^g, K_y^g \rangle = \iiint \hat{x}(\xi) \bar{\hat{y}}(\zeta) \hat{g}_\nu(\zeta) \bar{\hat{g}_\nu}(\xi) e^{i2\pi\tau(\xi-\zeta)} d\xi d\zeta d\tau d\nu, \quad (2.82)$$

et on voit apparaître une formulation “duale” de l'expression obtenue pour la transformée de Gabor instationnaire, dans laquelle les fonctions sont remplacées par leurs transformées de Fourier et où les variables temporelles et fréquentielles sont interchangées.

On obtient alors :

$$\langle K_x^g, K_y^g \rangle = \int \hat{x}(\xi) \bar{\hat{y}}(\xi) \left(\int |\hat{g}_\nu(\xi)|^2 d\nu \right) d\xi. \quad (2.83)$$

Comme la famille $(g_\nu)_{\nu \in \mathbb{R}}$ vérifie :

$$\forall \xi \in \mathbb{R}, \quad \int |\hat{g}_\nu(\xi)|^2 d\nu = 1, \quad (2.84)$$

et comme la transformée de Fourier est isométrique, la propriété d'isométrie est bien vérifiée.

Propriété 2.14 (Inversion) *La transformée est inversible selon la relation suivante, dans laquelle l'égalité est prise au sens du produit scalaire dans $L^2(\mathbb{R})$:*

$$\forall x \in L^2(\mathbb{R}), \quad x = \iint K_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu. \quad (2.85)$$

En effet, pour $(x, y) \in (L^2(\mathbb{R}))^2$, en remplaçant G_y^g par sa définition, on a

$$\langle K_x^g, K_y^g \rangle = \iiint K_x^g(\tau, \nu) g_{\tau\nu}(t) \bar{y}(t) dt d\tau d\nu \quad (2.86)$$

$$= \left\langle \iint K_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, y \right\rangle \quad (2.87)$$

Comme K est une isométrie, on a

$$\forall (x, y) \in (L^2(\mathbb{R}))^2, \quad \langle x, y \rangle = \left\langle \iint K_x^g(\tau, \nu) g_{\tau\nu} d\tau d\nu, y \right\rangle \quad (2.88)$$

Propriété 2.15 (Conservation de l'énergie) *La propriété d'isométrie implique la conservation d'énergie par la transformée, selon la relation*

$$\int |x(t)|^2 dt = \iint |K_x^g(\tau, \nu)|^2 d\tau d\nu. \quad (2.89)$$

Discrétisation

J'ai décrit une version discrète en dimension finie de la transformée dans [25]. Cette version est utilisable pour une implémentation effective.

Il est également possible de construire une version semi-discrète de la transformée en conservant des fonctions de variables continues, comme présenté ci-dessous.

2.2.4 Version semi-discrète

L'élément clé dans la démonstration des propriétés de la transformée est la covariance par rapport à la translation suivant la variable continue τ . L'autre variable, la variable fréquentielle ν , peut être discrétisée en conservant les bonnes propriétés de la transformée.

On introduit alors une famille dénombrable de fenêtres $(h_n)_{n \in \mathbb{Z}}$ à laquelle on impose cette fois de vérifier la condition d'admissibilité :

$$\forall \xi \in \mathbb{R}, \quad \sum_{n \in \mathbb{Z}} |\hat{h}_n(\xi)|^2 = 1, \quad (2.90)$$

et on obtient la famille semi-continue d'atomes $(h_{\tau n})_{(\tau, n) \in \mathbb{R} \times \mathbb{Z}}$ par translation temporelle des fonctions h_n , selon la relation :

$$\forall t \in \mathbb{R}, \quad h_{\tau n}(t) = h_n(t - \tau). \quad (2.91)$$

On introduit l'espace \mathcal{F} vérifiant :

$$\mathcal{F} = \left\{ X : \mathbb{R} \times \mathbb{Z} \rightarrow \mathbb{C} : \sum_{n \in \mathbb{Z}} \int |X(\tau, n)|^2 d\tau < +\infty \right\}. \quad (2.92)$$

L'espace \mathcal{F} a une structure d'espace de Hilbert, grâce au produit scalaire suivant :

$$\forall (X, Y) \in \mathcal{F}, \quad \langle X, Y \rangle = \sum_{n \in \mathbb{Z}} \int X(\tau, n) Y(\tau, n) d\tau. \quad (2.93)$$

On définit alors la transformée du signal x de $L^2(\mathbb{R})$ comme la fonction L_x^h vérifiant :

$$L_x^h(\tau, n) = \int x(t) \overline{h_n}(t - \tau) dt, \quad (2.94)$$

Si (2.90) est vérifiée, alors $L_x^h \in \mathcal{F}$, et les propriétés de la transformée sont conservées, les démonstrations s'effectuant de manière similaire en remplaçant l'intégrale par rapport à la variable ν par une somme sur l'indice n .

En particulier, la formule d'inversion s'écrit :

$$\forall x \in L^2(\mathbb{R}), \quad x = \sum_{n \in \mathbb{Z}} \int L_x^h(\tau, n) h_{\tau n} d\tau. \quad (2.95)$$

On voit donc que les deux cas étudiés (conservation de la covariance par translation fréquentielle ou par translation temporelle) sont similaires, la translation fréquentielle étant remplacée par la translation temporelle dans le second cas et les conditions similaires s'appliquant sur la transformée de Fourier des atomes plutôt que sur les atomes eux-mêmes.

En conséquence, un seul problème reste maintenant à résoudre pour traiter les deux cas : comment construire des familles de fonctions vérifiant les conditions obtenues précédemment ?

2.2.5 Remarque sur la construction des atomes

Les deux types de transformations présentant des caractéristiques communes, nous prenons un peu de recul en présentant une méthode⁶ de construction des atomes

⁶Il s'agit d'une des approches possibles pour la construction des atomes. D'autres approches sont facilement envisageables.

applicables aux deux types de transformées. Dans le cas des transformées covariantes par translation fréquentielle, la méthode s'applique directement sur les atomes, la variable λ étant alors une variable temporelle. Dans le cas des transformées covariantes par translation temporelle, la méthode s'applique sur les transformées de Fourier des atomes et la variable λ est alors une variable fréquentielle.

Dans le cas d'une famille indénombrable, le problème étudié est la construction d'une famille de fonctions $(g_\lambda)_{\lambda \in \mathbb{R}}$ vérifiant la condition

$$\forall u \in \mathbb{R}, \quad \int |g_\lambda(u)|^2 d\lambda = 1. \quad (2.96)$$

Les atomes peuvent être facilement construits à partir de la famille $(\gamma_\lambda)_{\lambda \in \mathbb{R}}$ vérifiant la condition moins restrictive suivante : il suffit de supposer qu'il existe des constantes positives A et B vérifiant $0 < A \leq B < \infty$ telles que :

$$\forall u \in \mathbb{R}, \quad A \leq \int |\gamma_\lambda(u)|^2 d\lambda \leq B. \quad (2.97)$$

En effet, il suffit alors de considérer la famille $(g_\lambda)_{\lambda \in \mathbb{R}}$ telle que

$$\forall \lambda \in \mathbb{R}, \forall u \in \mathbb{R}, \quad g_\lambda(u) = \frac{\gamma_\lambda(u)}{\sqrt{\int |\gamma_\lambda(u)|^2 d\lambda}}, \quad (2.98)$$

qui vérifie la condition 2.96.

Dans le cas d'une famille dénombrable, le problème étudié est très similaire, il s'agit de la construction d'une famille de fonctions $(h_k)_{k \in \mathbb{Z}}$ vérifiant la condition

$$\forall u \in \mathbb{R}, \quad \sum_{k \in \mathbb{Z}} |h_k(u)|^2 = 1. \quad (2.99)$$

On effectue le même type de construction en partant alors d'une famille $(\eta_k)_{k \in \mathbb{Z}}$ vérifiant : il existe des constantes positives A et B vérifiant $0 < A \leq B < \infty$ telles que

$$\forall u \in \mathbb{R}, \quad A \leq \sum_{k \in \mathbb{Z}} |\eta_k(u)|^2 \leq B. \quad (2.100)$$

C'est alors la famille $(h_k)_{k \in \mathbb{Z}}$ telle que

$$\forall k \in \mathbb{Z}, \forall u \in \mathbb{R}, \quad h_k(u) = \frac{\eta_k(u)}{\sqrt{\sum_{k \in \mathbb{Z}} |\eta_k(u)|^2}} \quad (2.101)$$

qui vérifie la condition 2.99.

Pour une utilisation pratique, la condition 2.100 peut par exemple être remplie en utilisant des fonctions η_k à supports $\text{supp}(\eta_k)$ finis et strictement positives sur

leurs supports, telles que l'ensemble des fonctions couvre tout l'axe réel, c'est à dire $\bigcup \text{supp}(\eta_k) = \mathbb{R}$ et que en chaque point, il y ait un nombre borné de fenêtres qui se chevauchent, c'est à dire qu'il existe $N \in \mathbb{N}^*$ tel que

$$\forall u \in \mathbb{R}, \quad \text{card} \{k; u \in \text{supp}(\eta_k)\} < N. \quad (2.102)$$

Nous verrons dans la partie 2.2.7 qu'une méthode alternative peut être mise en place pour construire des familles dénombrables vérifiant la condition (2.99) à partir de familles indénombrables vérifiant (2.96).

Avant cela, l'application de cette construction dans le cas des transformées covariantes par translation fréquentielle va être illustré à travers une étude plus approfondie de la transformée de Gabor instationnaire.

2.2.6 Transformée de Gabor instationnaire

La construction précédente peut être utilisée pour définir une transformée de Gabor instationnaire, c'est à dire une transformée de Gabor dans laquelle la fenêtre d'analyse évolue dans le temps. Comme on l'a vu en 2.1.3 avec la transformée de Gabor, l'influence de la fenêtre utilisée sur la représentation obtenue est importante et pour un signal dont les caractéristiques fréquentielles évoluent fortement dans le temps, on peut souhaiter modifier la fenêtre, et en particulier sa taille, au cours du temps afin de l'adapter à chaque instant aux caractéristiques du signal.

Pour cela on utilise la méthode de construction des atomes à partir d'une famille de fenêtres vérifiant (2.100) et qui évoluent en fonction du temps de manière adaptée au signal, selon des caractéristiques définies par l'utilisateur.

Le problème avec cette construction vient du fait que les fenêtres effectivement utilisées pour l'analyse ne sont pas celles qui sont choisies par l'utilisateur, puisqu'elles sont déformées afin que la famille de fonctions vérifie la condition (2.69). Cependant, sous certaines conditions, la déformation des fenêtres peut rester suffisamment limitée pour que l'interprétation de la représentation reste naturelle. Pour cela, il faut que la forme de la fenêtre choisie par l'utilisateur varie lentement au cours du temps par rapport à la taille de la fenêtre. On peut alors localement considérer que la fenêtre ne varie pas, et la transformée de Gabor instationnaire se comporte localement comme un repère de Gabor classique, dans lequel la déformation de la fenêtre dépend de la finesse d'échantillonnage temporel.

En revanche, lorsque la fenêtre varie rapidement, elle est fortement déformée autour de la zone de variation, comme le montre la figure 2.12. Sur cette figure, on représente la forme des fenêtres autour d'une zone de changement de taille en représentant seulement une fenêtre toutes les douze fenêtres afin de faciliter la lisibilité de la figure. De plus toutes les fenêtres sont normalisées en amplitude afin qu'elles atteignent toutes un maximum de 1, ce qui facilite la comparaison des formes

des différentes fenêtres et permet de bien apprécier la déformation. En bas de la figure est représentée la taille de fenêtre utilisée en fonction de l'indice temporel. On voit à gauche que lorsque le changement de taille est brusque, les fenêtres chevauchant la zone de changement de taille sont fortement déformées, alors qu'à droite, en réalisant un changement de taille progressif sur un nombre d'échantillon qui reste cependant limité, on obtient une déformation limitée des fenêtres dans la zone de changement de taille.

Exemple 2.7 *La figure 2.13 illustre l'intérêt de la transformée de Gabor instationnaire pour la représentation de signaux réels variant fortement au cours du temps. Le signal analysé est un son de glockenspiel dont les caractéristiques sont très différentes selon la portion de signal considérée : dans les zones correspondant aux attaques des notes, on a un signal transitoire fortement instationnaire sur une très courte durée qui est bien représenté en utilisant une petite fenêtre temporelle ; dans les zones correspondant à la résonance des lames, on a un signal quasi-stationnaire composé de partiels sinusoidaux sur une plus longue durée, qui est bien représenté en utilisant une grande fenêtre permettant de bien localiser et séparer les différents partiels. On peut effectivement voir sur la représentation du bas de la figure 2.13 obtenue par transformée de Gabor classique avec une petite taille de fenêtre que les attaques des notes sont bien localisées sur la représentation. Cependant la faible résolution fréquentielle de la fenêtre implique un étalement des partiels dans la partie de résonance où on peut constater que les partiels ne sont pas résolus. En revanche sur la représentation du centre, obtenue par transformée de Gabor classique avec une grande fenêtre, les partiels sont bien résolus, mais les attaques sont maintenant "étalées". La représentation du haut est obtenue par transformée de Gabor instationnaire, en choisissant une petite taille de fenêtre (celle utilisée pour la représentation du bas) dans les zones d'attaques et une grande fenêtre (celle utilisée pour la représentation du centre) dans les zones de résonance, et en faisant varier la taille de fenêtre suffisamment lentement pour éviter de trop fortes déformations des fenêtres, comme évoqué plus haut. On obtient alors une meilleure représentation du signal global, puisqu'on a dans les zones d'attaques une représentation très proche de celle obtenue avec petite fenêtre et dans les zones de résonance une représentation très proches de celle obtenue avec la grande fenêtre.*

Pour finir, signalons deux remarques sur la transformée de Gabor instationnaire :

Remarque 2.8 *La transformée de Gabor instationnaire permet d'améliorer la représentation pour des signaux contenant à chaque instant des composantes aux caractéristiques temps-fréquence proches. Par contre, lorsque le signal contient au même instant des composantes aux caractéristiques temps-fréquence très différentes à des fréquences différentes, des difficultés se posent toujours. Pour résoudre*

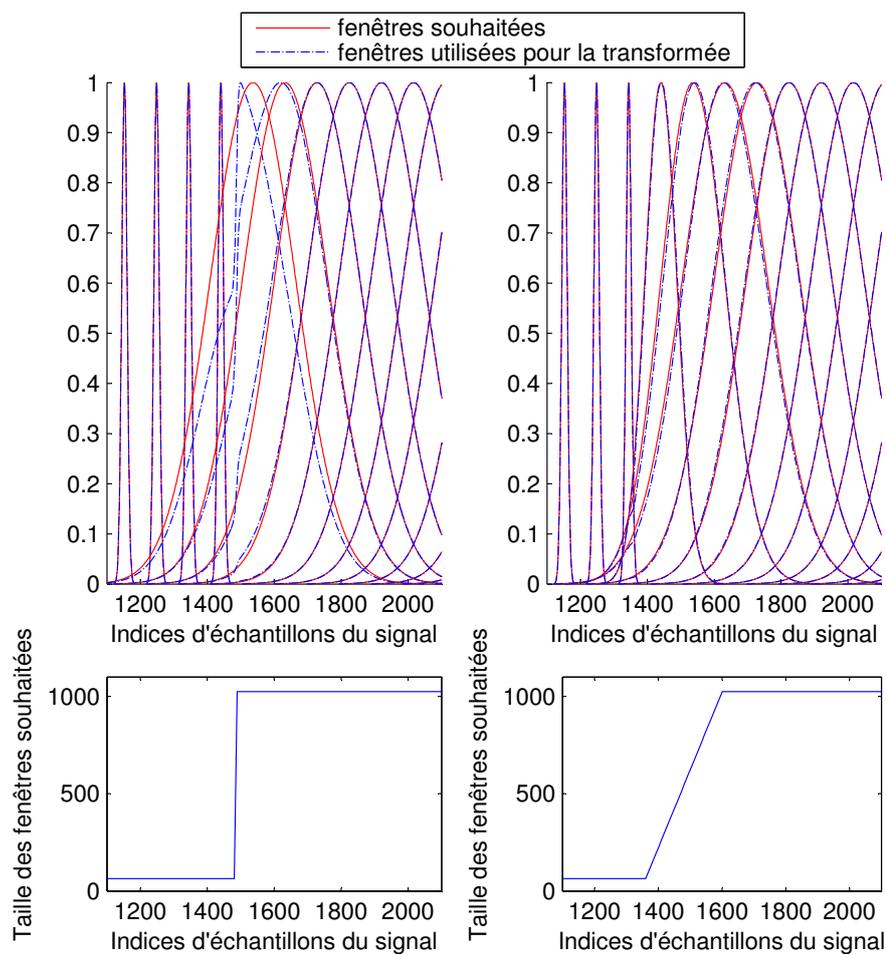


FIG. 2.12 – Influence de l'évolution de la taille des fenêtres au cours du temps sur la forme des fenêtres utilisées pour la transformée de Gabor instationnaire : à gauche, forte déformation dans le cas d'un passage brusque d'une taille de 64 à 1024 points, à droite, faible déformation dans le cas d'une passage d'une taille de 64 à 1024 points en 250 échantillons de signal

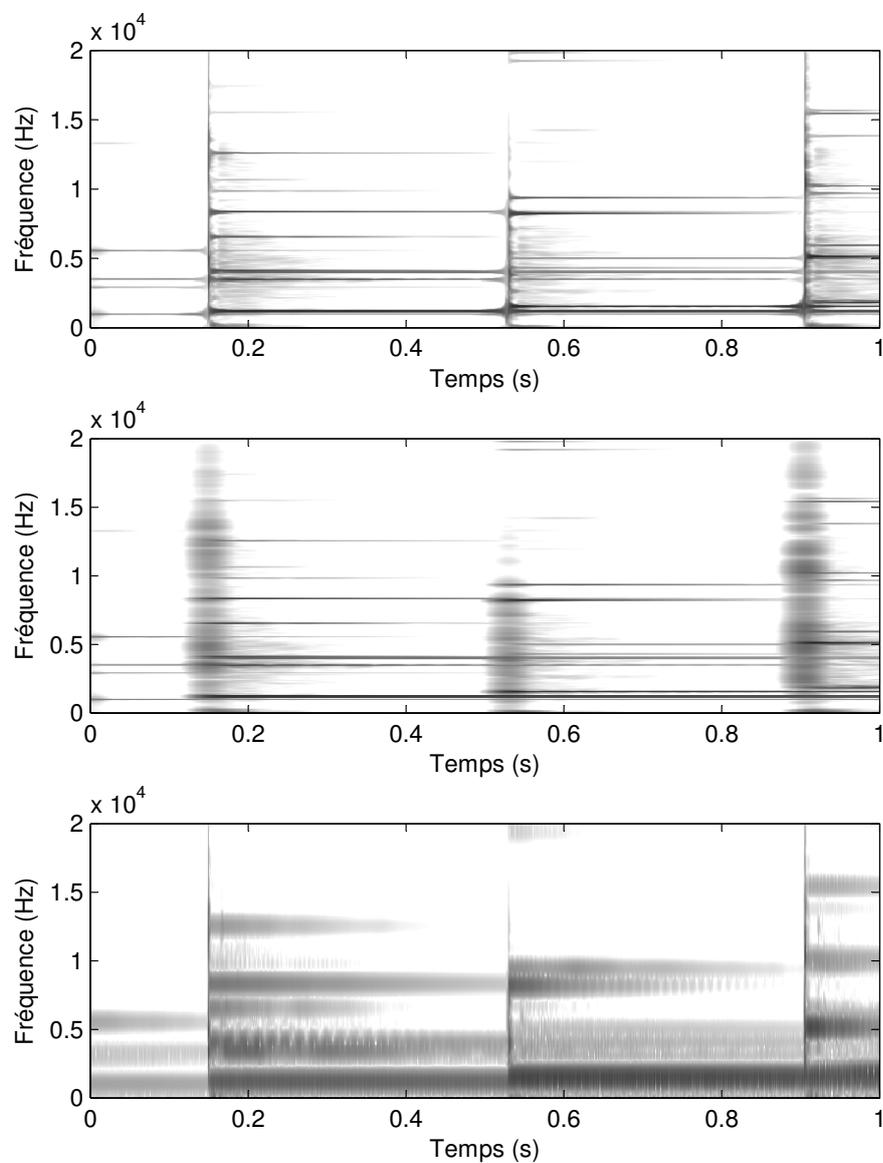


FIG. 2.13 – Comparaison des représentations obtenues avec la transformée de Gabor instationnaire et la transformée de Gabor classique : sur un son de glockenspiel, en haut transformée de Gabor instationnaire, au centre transformée de Gabor classique avec une grande fenêtre, en bas transformée de Gabor classique avec une petite fenêtre

ce problème, il faut aller plus loin en permettant d'utiliser des fenêtres d'analyse différentes dans différentes régions du plan temps-fréquence. Une méthode numérique pour mettre en place une analyse de ce type est présentée dans la partie 2.4.

Remarque 2.9 *L'utilisation de la transformée de Gabor instationnaire implique l'indication de l'évolution de la taille de fenêtre au cours de temps, ce qui augmente fortement le nombre de paramètres que l'utilisateur doit régler. Pour faciliter ce réglage, la partie 2.3 propose une approche pour l'adaptation automatique de la taille de fenêtre.*

2.2.7 Paquets d'atomes temps-fréquence

Comme on l'a vu en 2.1.3 et 2.1.4, la transformée de Gabor et la transformée en ondelettes peuvent être interprétées comme des analyses par des bancs de filtres. La relation entre la fréquence centrale et la largeur de bande de ces filtres est complètement déterminée par le type de représentation et l'atome générateur de la transformée : pour la transformée de Gabor, la largeur de bande est constante pour toutes les fréquences et fixée par les caractéristiques de la fenêtre d'analyse, pour la transformée en ondelettes, la largeur de bande est proportionnelle à la fréquence centrale, et le coefficient de proportionnalité dépend des caractéristiques de l'ondelette mère.

Pour pouvoir aller plus loin dans l'analyse des signaux, on peut souhaiter construire des bancs de filtres plus flexibles dans lesquelles la largeur de bande peut varier en fonction de la fréquence centrale de manière plus souple, par exemple en suivant une "loi" fréquentielle fixée à l'avance. Il est ainsi possible d'adapter l'analyse au signal, où encore de construire une analyse utilisant une échelle fréquentielle perceptive comme dans [6].

Des transformées de ce type peuvent être mises en place à partir de transformée de Gabor ou de la transformée en ondelettes, en utilisant une construction spécifique des atomes que nous allons maintenant décrire. Dans le cas où la construction est réalisée à partir de la transformée en ondelettes, on retrouve les paquets d'ondelettes continus proposés dans [14] [36] et repris pour des applications au traitement audio dans [24] [26].

Méthode de construction des paquets

Pour construire les paquets d'atomes temps-fréquence, on utilise une méthode de construction des atomes dans le cas dénombrable alternative à celle décrite dans 2.2.5.

Pour cela, on part d'une famille $(g_\lambda)_{\lambda \in \mathbb{R}}$ indénombrable vérifiant la condition (2.96) et on définit une suite de réels $(\lambda_k)_{k \in \mathbb{Z}}$ vérifiant $\lambda_k < \lambda_{k+1}$ et telle que $\lim_{k \rightarrow +\infty} \lambda_k = +\infty$ et $\lim_{k \rightarrow -\infty} \lambda_k = -\infty$, la condition (2.96) peut s'écrire :

$$\forall u \in \mathbb{R}, \quad 1 = \int |g_\lambda(u)|^2 d\lambda = \sum_{k \in \mathbb{Z}} \left(\int_{\lambda_{k-1}}^{\lambda_k} |g_\lambda(u)|^2 d\lambda \right). \quad (2.103)$$

En introduisant pour tout $k \in \mathbb{Z}$ une fonction h_k telle que

$$\forall u \in \mathbb{R}, \quad |h_k(u)|^2 = \int_{\lambda_{k-1}}^{\lambda_k} |g_\lambda(u)|^2 d\lambda, \quad (2.104)$$

on construit donc une famille de fonctions $(h_k)_{k \in \mathbb{Z}}$ vérifiant la condition (2.99).

Cette méthode de construction peut en particulier être utilisée en prenant comme famille indénombrable de départ celle utilisée dans la transformée de Gabor ou la transformée en ondelettes.

Construction à partir de la transformée de Gabor

Dans le cas de la transformée de Gabor, la condition d'admissibilité se ramène uniquement à la relation de normalisation la fenêtre d'analyse $g : \int |g(t)|^2 dt = 1$.

Comme la transformée de Fourier est isométrique, on a pour la transformée de Fourier \hat{g} de la fenêtre g :

$$\int |\hat{g}(\xi)|^2 d\xi = 1, \quad (2.105)$$

et donc, en définissant pour tout ν la fonction g_ν par

$$\forall t \in \mathbb{R}, g_\nu(t) = g(t)e^{i2\pi\nu t}, \quad (2.106)$$

on a :

$$\forall \xi \in \mathbb{R}, \quad \int |\hat{g}(\xi - \nu)|^2 d\nu = \int |\hat{g}_\nu(\xi)|^2 d\nu = 1. \quad (2.107)$$

De là, on va pouvoir réaliser la construction que nous venons de décrire.

Pour une suite de nombres réels $(\nu_n)_{n \in \mathbb{Z}}$ vérifiant pour tout $n \in \mathbb{Z}$, $\nu_n < \nu_{n+1}$ et telle que $\lim_{n \rightarrow +\infty} \nu_n = +\infty$ et $\lim_{n \rightarrow -\infty} \nu_n = -\infty$, on considère les fonctions Γ_n telles que leurs transformées de Fourier vérifient :

$$\forall \xi \in \mathbb{R}, \quad |\hat{\Gamma}_n(\xi)|^2 = \int_{\nu_{n-1}}^{\nu_n} |\hat{g}_\nu(\xi)|^2 d\nu = \int_{\nu_{n-1}}^{\nu_n} |\hat{g}(\xi - \nu)|^2 d\nu. \quad (2.108)$$

D'après la relation (2.107), la famille de fonctions $(\Gamma_n)_{n \in \mathbb{Z}}$ vérifie la condition

$$\forall \xi \in \mathbb{R}, \quad \sum_{n \in \mathbb{Z}} |\hat{\Gamma}_n(\xi)|^2 = 1, \quad (2.109)$$

et permet, d'après ce que l'on a vu en 2.2.3, de définir la décomposition du signal isométrique suivante :

$$L_x^\Gamma(\tau, n) = \int x(t) \overline{\Gamma_n}(t - \tau) dt, \quad (2.110)$$

à partir de laquelle le signal peut-être reconstruit selon la relation

$$x(t) = \sum_{n \in \mathbb{Z}} \int L_x^\Gamma(\tau, n) \Gamma_n(t - \tau) d\tau. \quad (2.111)$$

Remarque 2.10 On désigne les fonctions Γ_n par le terme de paquets d'atomes temps-fréquence car chaque fonction Γ_n est construite en regroupant par intégration dans l'espace de Fourier des atomes de la transformée de Gabor.

Construction à partir de la transformée en ondelettes

Comme dans 2.1.4, on restreint dans cette section la construction au cas des signaux de $H^2(\mathbb{R})$ pour simplifier la formulation.

Pour la transformée en ondelettes, la condition d'admissibilité prend la forme

$$\int_0^{+\infty} \frac{|\hat{h}(\nu)|^2}{\nu} d\nu = 1. \quad (2.112)$$

Pour $a > 0$, en effectuant le changement de variable $a = \frac{\nu}{\xi}$ on a donc

$$\forall \xi \in \mathbb{R}_+^*, \quad \int_0^{+\infty} \frac{|\hat{h}(a\xi)|^2}{a} da = 1. \quad (2.113)$$

En introduisant les fonctions h_a telle que

$$\forall t \in \mathbb{R}, \quad h_a(t) = a^{-\frac{3}{2}} h\left(\frac{t}{a}\right) \quad (2.114)$$

qui vérifient

$$\forall \xi \in \mathbb{R}, \quad \hat{h}_a(\xi) = \frac{\hat{h}(a\xi)}{\sqrt{a}}, \quad (2.115)$$

on peut donc écrire

$$\forall \xi \in \mathbb{R}_+^*, \quad \int_0^{+\infty} |\hat{h}_a(\xi)|^2 da = 1. \quad (2.116)$$

En introduisant la suite de nombres réels $(a_n)_{n \in \mathbb{Z}}$ vérifiant pour tout $n \in \mathbb{Z}$ $a_{n+1} > a_n$ et telle que $\lim_{n \rightarrow +\infty} a_n = +\infty$ et $\lim_{n \rightarrow -\infty} a_n = 0$, et en définissant les fonctions Ψ_n telles que leurs transformées de Fourier vérifient

$$|\hat{\Psi}_n(\nu)|^2 = \int_{a_{n-1}}^{a_n} |\hat{h}_a(\nu)|^2 da = \int_{a_{n-1}}^{a_n} |\hat{h}(a\nu)|^2 \frac{da}{a}, \quad (2.117)$$

on obtient une famille de paquets d'ondelettes $(\Psi_n)_{n \in \mathbb{Z}}$ vérifiant la condition

$$\forall \nu \in \mathbb{R}_+, \quad \sum_{n \in \mathbb{Z}} |\hat{\Psi}_n(\nu)|^2 = 1, \quad (2.118)$$

et permettant de définir la décomposition du signal isométrique suivante :

$$L_x^\Psi(\tau, n) = \int x(t) \overline{\Psi_n(t - \tau)} dt, \quad (2.119)$$

à partir de laquelle le signal peut-être reconstruit selon la relation

$$x(t) = \sum_{n \in \mathbb{Z}} \int L_x^\Psi(\tau, n) \Psi_n(t - \tau) d\tau. \quad (2.120)$$

En introduisant une fonction ϕ , appelée fonction d'échelle⁷, vérifiant :

$$|\hat{\phi}(\nu)|^2 = \int_1^{+\infty} |\hat{h}(u\nu)|^2 \frac{du}{u}, \quad (2.121)$$

et en notant Φ_n la fonction telle que

$$\hat{\Phi}_n(\nu) = \hat{\phi}(a_n \nu), \quad (2.122)$$

on a :

$$|\hat{\Phi}_{n-1}(\nu)|^2 - |\hat{\Phi}_n(\nu)|^2 = \int_1^{+\infty} |\hat{h}(u a_{n-1} \nu)|^2 \frac{du}{u} - \int_1^{+\infty} |\hat{h}(u a_n \nu)|^2 \frac{du}{u} \quad (2.123)$$

$$= \int_{a_{n-1}}^{+\infty} |\hat{h}(a\nu)|^2 \frac{da}{a} - \int_{a_n}^{+\infty} |\hat{h}(a\nu)|^2 \frac{da}{a} \quad (2.124)$$

$$= \int_{a_{n-1}}^{a_n} |\hat{h}(a\nu)|^2 \frac{da}{a} \quad (2.125)$$

$$= |\hat{\Psi}_n(\nu)|^2. \quad (2.126)$$

Ainsi, seule la connaissance de la fonction d'échelle est nécessaire pour construire les paquets d'ondelettes.

⁷Le terme "fonction d'échelle" est choisi par analogie avec les analyses multi-résolution (voir par exemple [31] pour une définition), où la construction est précisément basée sur une fonction d'échelle qui joue le même rôle.

Caractéristiques fréquentielles des paquets

Compte tenu du mode de construction des paquets, leurs caractéristiques fréquentielles dépendent de la fenêtre g , ou de l'ondelette h selon le type de paquet, et des bornes d'intégration. Pour pouvoir utiliser les paquets en suivant une "loi" fixée donnant la largeur de bande en fonction de la fréquence, il est nécessaire de contrôler ces caractéristiques.

Pour cela, on définit les notions de fréquence centrale et de largeur de bande des paquets à partir des moments de leurs transformées de Fourier. On peut alors calculer explicitement les relations liant la fréquence centrale et la largeur de bande d'un paquet à celles de la fenêtre g , ou de l'ondelette h selon le type de paquets, et aux bornes d'intégration. Le détail des définitions et des calculs est présenté dans l'annexe A.1 de ce document.

Pour les paquets construits à partir de la transformée de Gabor, on obtient ainsi pour le paquet Γ_n la fréquence centrale :

$$f_{\Gamma_n} = \frac{\nu_{n-1} + \nu_n}{2}. \quad (2.127)$$

La largeur de bande Δ_{Γ_n} vérifie :

$$\Delta_{\Gamma_n}^2 = \Delta_g^2 + \frac{(\nu_n - \nu_{n-1})^2}{12}. \quad (2.128)$$

Pour les paquets construits à partir de la transformée en ondelettes, en notant $\mu_n = \frac{1}{a_n}$, la fréquence centrale du paquet Ψ_n vaut

$$f_{\Psi_n} = \frac{\mu_n + \mu_{n-1}}{2} f_h, \quad (2.129)$$

et largeur de bande vérifie

$$\Delta_{\Psi_n}^2 = \frac{\mu_n^2 + \mu_n \mu_{n-1} + \mu_{n-1}^2}{3} \Delta_h^2 + \frac{(\mu_n - \mu_{n-1})^2}{12} f_h^2. \quad (2.130)$$

On peut utiliser ces expressions, pour construire une analyse dans laquelle la largeur de bande suit une fonction F de la fréquence centrale fixée par l'utilisateur.

Pour cela, dans le cas où les paquets sont obtenus à partir de la transformée de Gabor, on fixe une valeur ν_0 , puis on résout (analytiquement ou numériquement, selon la fonction F utilisée) de proche en proche pour $n \in \mathbb{Z}$, $\Delta_{\Gamma_n} = F(f_{\Gamma_n})$, ce qui permet de fixer successivement des valeurs de ν_n , et de construire les paquets correspondants.

Dans le cas où les paquets sont obtenus à partir de la transformée en ondelettes, on utilise le même principe de construction de proche en proche, en partant d'une échelle initiale a_0 que l'on fixe.

Remarque 2.11 *On voit que cette méthode induit certaines limitations sur le choix de F . En effet, par exemple dans le cas où on part de la transformée de Gabor, le choix de F est limité par le fait que l'équation $\Delta_{\Gamma_n} = F(f_{\Gamma_n})$ doit avoir une solution pour tout $n \in \mathbb{Z}$ avec la valeur de ν_0 choisie.*

On peut déjà noter dans ce cas une première limitation simple sur F dans l'expression de la largeur de bande des paquets, puisqu'on a $\Delta_{\Gamma_n} \geq \Delta_g$, il faut donc avoir $\forall f \in \mathbb{R}, F(f) \geq \Delta_g$.

De la même manière, en partant d'ondelettes, on peut noter que la largeur de bande minimale pour un paquet dont une des bornes d'intégration est a_n est cette fois $\frac{\Delta_h}{a_n}$, c'est à dire la largeur de bande de l'ondelette à l'échelle a_n .

Remarque 2.12 *Il faut également remarquer qu'avec cette construction, l'utilisateur ne choisit pas directement la valeur de la fréquence centrale des paquets, puisqu'elle est imposée par la limite d'intégration du paquet précédent et la loi liant la largeur de bande à la fréquence centrale. En particulier, l'utilisateur ne peut pas imposer le nombre de paquets présents dans une bande de fréquence donnée. Ce dernier dépend à la fois de la loi utilisée et de la largeur de bande de l'atome père, ce qui détermine également la redondance de la représentation. Cependant, on peut augmenter la redondance de la représentation, en utilisant simultanément plusieurs constructions similaires réalisées à partir d'une valeur de ν_0 (ou a_0 selon le cas) différente.*

Exemple 2.8 *La figure 2.14 illustre l'utilisation des paquets d'ondelettes continus sur un extrait de parole (les mots "le masquage"). Elle permet de comparer les représentations obtenues pour la transformée de Gabor, la transformée en Ondelettes, et avec des paquets d'ondelettes ayant une largeur de bande constante sur l'échelle des Barks. Cette échelle est une échelle fréquentielle perceptive basé sur la mesure des bandes critiques de l'oreille. La construction de cette échelle est présentée dans [49].*

On voit que, la représentation la plus satisfaisante pour ce signal est celle obtenue avec les paquets d'ondelettes. En effet, sur la transformée de Gabor, les fréquences supérieures à 5000 Hz sont inutilement détaillées, alors que dans le signal, seul le son "s" crée une énergie significative dans cette région. Sur la transformée en ondelettes, les premiers partiels sont inutilement détaillés, alors que les partiels de rangs plus élevés ne sont plus résolus. Sur la transformée utilisant les paquets d'ondelettes, les partiels sont bien résolus et le nombre de coefficients consacrés aux fréquences aiguës reste limité tout en permettant de distinguer nettement le "s".

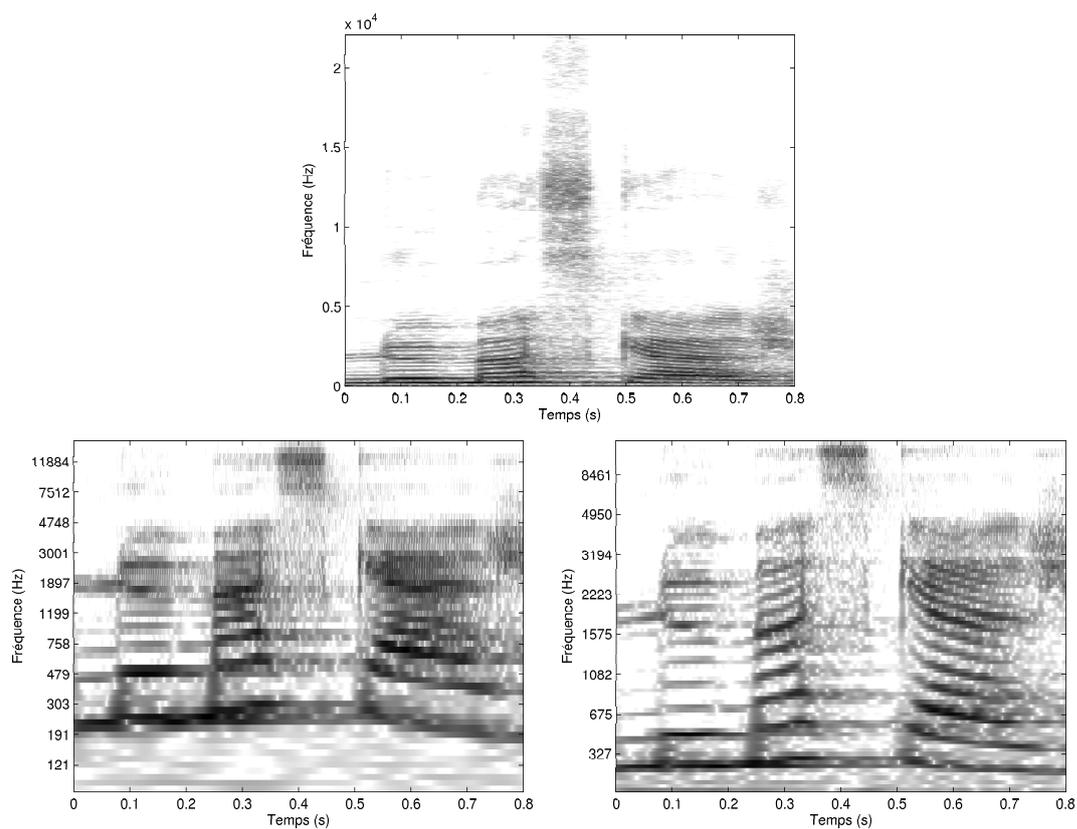


FIG. 2.14 – Comparaison des représentations obtenues par transformée de Gabor, transformée en ondelettes et transformée sur une échelle en Barks : sur un extrait de parole, en haut, transformée de Gabor, en bas à gauche, transformée en ondelettes, en bas à droite transformée suivant une échelle fréquentielle en Barks

2.3 Représentations adaptatives supervisées

On a vu en 2.1.3 l'importance du choix des paramètres d'analyse pour la qualité des représentations temps-fréquence obtenues. Pour faciliter l'utilisation de ces représentations, en particulier par des non-spécialistes, il est souhaitable de disposer de méthodes automatiques de choix des paramètres afin d'assurer la bonne qualité de la représentation.

Une stratégie possible est de se donner une famille paramétrique de représentations dans laquelle on recherche la "meilleure". Il faut pour cela se doter d'un critère définissant ce que l'on entend par "meilleur".

Dans ce travail, on se limite au cas de l'optimisation de la fenêtre d'une transformée de Gabor au signal. Cependant, les critères introduits sont définis à partir du spectrogramme associé à la transformée de Gabor et une extension de l'approche proposée à d'autres transformées bilinéaires dans lesquelles des paramètres doivent être réglés est envisageable.

Deux approches vont maintenant être évoquées pour définir les critères d'optimalité. La première est basée sur des critères d'adéquation qui mesurent la "ressemblance" de la fenêtre au signal. La deuxième repose sur des critères de parcimonie qui permettent d'identifier la représentation qui concentre le mieux l'énergie du signal sur un faible nombre de coefficients.

2.3.1 Critères d'optimalité

Dans cette section, le point de départ est un signal **fixé** x dont on considère la transformée de Gabor continue G_x^g . Le problème posé est de trouver la meilleure fenêtre $g \in L^2(\mathbb{R})$, $\|g\| = 1$.

Critères d'adéquation

Cette approche est proposée par W. Kozek dans [29]. Dans cette approche, la fenêtre optimale est définie comme celle dont l'"étalement" dans le plan temps-fréquence est le plus proche de celui du signal. La notion d'"étalement" temps-fréquence est mesurée à l'aide des fonctions d'ambiguïté A_x et A_g du signal x et de la fenêtre g , avec :

$$A_x(\tau, \nu) = \int x(t) \bar{x}(t - \tau) e^{-i2\pi\nu(t-\tau)} dt. \quad (2.131)$$

Le choix de la fenêtre optimale g_{opt} selon cette démarche prend, dans les formulations les plus simples, la forme :

$$g_{opt} = \arg \max_{\{g \in L^2(\mathbb{R}) : \|g\|=1\}} \langle |A_x|^2, |A_g|^2 \rangle, \quad (2.132)$$

Cette optimisation peut alors être réalisée numériquement, dès lors que l'on s'est donné une famille paramétrée de fenêtres. Les choix classiques incluent notamment des familles de copies d'une fenêtre de référence, à laquelle on fait subir des transformations simples, telles que des dilatations, ou parfois des "rotations" dans le plan temps-fréquence.

Cette approche paraît particulièrement intéressante lorsqu'une connaissance a priori permet de caractériser la forme de la fonction d'ambiguïté du signal, afin d'effectuer la recherche de la fenêtre optimale dans une classe de fonctions bien adaptées. Dans un cadre plus général, cette approche devrait faire apparaître le même type de limitations que celles que nous verrons dans la deuxième approche, en nécessitant un coût de calcul du critère d'optimalité plus important. Pour cette raison, nous n'avons pas retenu cette approche dans nos applications. Nous détaillons maintenant l'approche que nous avons développée.

Critères de parcimonie

Une approche alternative consiste à évaluer la qualité d'une représentation temps-fréquence par sa capacité à "concentrer" l'information sur un nombre limité de coefficients, ou sur des régions les plus compactes possibles du plan temps-fréquence. On parle alors d'approches parcimonieuses. C'est sur ces approches que nous nous sommes particulièrement focalisés.

Suivant Wickerhauser [44], on adapte les notions d'entropie issues de la théorie de l'information, dont on exploite la capacité à mesurer la "concentration" d'une densité de probabilité. Notre stratégie est la suivante : une normalisation bien choisie transforme le spectrogramme d'un signal fixé x en densité de probabilité. À chaque choix de fenêtre g correspond une telle densité, et la "meilleure" fenêtre au sens de la parcimonie est celle qui minimise l'entropie choisie. Avant de discuter les limitations de cette approche, commençons par en donner les détails.

Les entropies utilisées sont l'entropie de Shannon et les entropies de Rényi, qui sont définies selon les relations suivantes⁸ :

à partir du spectrogramme normalisé ρ_x^g du signal x défini par

$$\rho_x^g(\tau, \nu) = \frac{|G_x^g(\tau, \nu)|^2}{\|x\|^2}, \quad (2.133)$$

l'entropie de Shannon est définie par :

$$S(g) = - \iint \rho_x^g(\tau, \nu) \log_2(\rho_x^g(\tau, \nu)) d\tau d\nu, \quad (2.134)$$

⁸Pour simplifier les notations on ne fait pas apparaître la dépendance des valeurs d'entropie au signal x , car on travaille à signal fixé, en utilisant ces entropies pour comparer plusieurs transformées obtenues pour le même signal x .

et la famille des entropies de Rényi par :

$$R_\alpha(g) = \frac{1}{1-\alpha} \log_2 \left(\iint (\rho_x^g(\tau, \nu))^\alpha d\tau d\nu \right), \quad \alpha \in (0, 1). \quad (2.135)$$

Remarque 2.13 *Il est intéressant de noter que l'entropie de Shannon est la limite de l'entropie de Rényi, selon la relation :*

$$S(g) = \lim_{\alpha \rightarrow 1} R_\alpha(g). \quad (2.136)$$

Remarque 2.14 *Optimiser R_α revient à optimiser la norme $\|\rho_x^g\|_\alpha$ définie par :*

$$\|\rho_x^g\|_\alpha = \left(\iint (\rho_x^g(\tau, \nu))^\alpha d\tau d\nu \right)^{\frac{1}{\alpha}}. \quad (2.137)$$

L'optimisation de ce type de norme est un autre choix classique de critère de parcimonie. L'introduction de l'entropie de Rényi permet de faire le lien entre cette approche et l'utilisation de l'entropie de Shannon, autre choix classique.

Remarque 2.15 *Signalons que, dans notre contexte, on ne s'intéresse qu'à l'interprétation de ces entropies en tant que mesure de parcimonie. D'autres interprétations sont possibles et utilisées dans d'autres contextes. Par exemple, l'interprétation des entropies en tant que mesure de l'information dans les représentations temps-fréquence pour l'entropie de Rényi est utilisée dans [1] où le but est de mesurer la complexité de la représentation et non sa parcimonie.*

On ne s'intéresse en conséquence pas à des informations plus fines qui pourraient être donnée par les valeurs d'entropie sur les représentations. En revanche il est important que le choix de la fenêtre optimal soit dans une certaine mesure indépendant du choix de l'entropie utilisée, afin que la notion de meilleure représentation soit définie sans ambiguïté.

Remarque 2.16 *Pour une utilisation pratique, on n'utilise pas la transformée continue de Gabor, mais des repères de Gabor discrets. Pour effectuer le calcul des entropies à partir des coefficients discrets, on est alors amené à remplacer les intégrales apparaissant dans les définitions ci-dessus par des sommes discrètes. Pour que les valeurs d'entropie ainsi obtenues permettent effectivement de comparer deux représentations, il faut alors que les intégrales soient approchées par des sommes discrètes obtenues avec une même densité d'échantillonnage. Pour cela, on compare des représentations de Gabor qui ont la même redondance, en conservant constant le rapport entre la taille de fenêtre et le pas de calcul entre deux fenêtres. Ainsi, pour une fenêtre plus petite, on a un échantillonnage temporel plus serré et un échantillonnage fréquentiel plus lâche que pour une grande fenêtre, et on obtient pour un même signal des transformées contenant le même nombre de coefficients pour des tailles de fenêtres différentes.*

Nous allons maintenant étudier les problèmes apparaissant lorsque l'on utilise ces critères entropiques pour l'adaptation de la taille de fenêtre sur l'ensemble de signaux complexes.

2.3.2 Limites de l'optimisation globale

Le but de cette section est de mettre au "banc d'essai" les idées d'optimisation d'entropie que nous avons explicitées plus haut, et telles qu'elles sont souvent utilisées dans la littérature. On va voir que leur domaine de pertinence reste assez limité, ce qui motive les développements présentés plus loin.

On s'intéresse au cas de l'optimisation globale du critère entropique, c'est à dire le cas dans lequel la fenêtre optimale est choisie comme celle permettant d'obtenir la valeur d'entropie la plus faible pour la transformée calculée sur l'ensemble du signal.

Pour faire apparaître les limites de cette démarche, on étudie, sur des exemples simples, les résultats que l'on obtient en faisant les calculs analytiques des critères entropiques. Ces résultats (exacts ou approchés) corroborent les tests numériques effectués sur des signaux plus complexes.

Nous choisissons pour cela d'utiliser des signaux et des fenêtres gaussiennes qui permettent de mener les calculs explicites à leur terme et ainsi de disposer d'outils simples pour tester nos idées.

Pour un atome gaussien

Tout d'abord, on s'intéresse à un signal x constitué d'un atome temps-fréquence gaussien d'échelle σ_0 et de fréquence ν_0 de la forme suivante :

$$x(t) = e^{i2\pi\nu_0 t} e^{-\frac{t^2}{2\sigma_0^2}}, \quad (2.138)$$

et on réalise la transformée de Gabor en utilisant une fenêtre gaussienne g normalisée d'échelle σ :

$$g(t) = \frac{1}{\sqrt{\sigma}\sqrt{\pi}} e^{-\frac{t^2}{2\sigma^2}}. \quad (2.139)$$

On souhaite savoir quelle est la valeur du paramètre σ qui détermine la largeur de la fenêtre optimale pour les critères entropiques considéré.

Il s'avère qu'il est possible dans ce cas de mener à bien l'optimisation des entropies de façon explicite. Les calculs d'entropie correspondant sont détaillés dans l'annexe A.2.1.

On obtient l'expression suivante pour l'entropie de Rényi :

$$R_\alpha(g) = -1 - \frac{\log_2(\alpha)}{1-\alpha} + \log_2 \left(\frac{\sigma}{\sigma_0} + \frac{\sigma_0}{\sigma} \right) \quad (2.140)$$

et pour l'entropie de Shannon l'expression :

$$S(g) = \frac{1}{\ln(2)} - 1 + \log_2 \left(\frac{\sigma}{\sigma_0} + \frac{\sigma_0}{\sigma} \right) \quad (2.141)$$

On voit que l'expression $R_\alpha(g)$ est minimale indépendamment de la valeur de α lorsque $\sigma_0 = \sigma$, c'est à dire lorsque la largeur de la fenêtre est égale à celle de l'atome analysé. On obtient également le même résultat avec l'entropie de Shannon. Ce résultat est satisfaisant puisque le choix de la fenêtre optimale est indépendant du critère utilisé. Cependant, dès que le signal devient plus complexe, une dépendance de la valeur optimale au critère apparaît. Pour illustrer cela on peut effectuer les mêmes calculs en ajoutant au signal x un deuxième atome aux caractéristiques temps-fréquence différentes.

Pour deux atomes gaussiens éloignés

On s'intéresse donc maintenant au cas où le signal x est composé de la somme de deux atomes gaussiens x_1 et x_2 d'échelles σ_1 et σ_2 et de fréquences ν_1 et ν_2 très éloignés dans le plan temps-fréquence :

$$x(t) = x_1(t) + x_2(t) = e^{i2\pi\nu_1 t} e^{-\frac{t^2}{2\sigma_1^2}} + e^{i2\pi\nu_2 t} e^{-\frac{t^2}{2\sigma_2^2}}. \quad (2.142)$$

On a alors l'approximation (voir l'annexe A.2.2 pour le détail des calculs) :

$$\begin{aligned} R_\alpha(g) \approx & -1 - \frac{\log_2(\alpha)}{1-\alpha} - \frac{\alpha}{1-\alpha} \log_2(\sigma_1 + \sigma_2) \\ & + \frac{1}{1-\alpha} \log_2 \left(\sigma_1^\alpha \left(\frac{\sigma}{\sigma_1} + \frac{\sigma_1}{\sigma} \right)^{1-\alpha} + \sigma_2^\alpha \left(\frac{\sigma}{\sigma_2} + \frac{\sigma_2}{\sigma} \right)^{1-\alpha} \right). \end{aligned} \quad (2.143)$$

Le calcul direct du minimum de cette expression ne peut pas se faire facilement. Mais pour des valeurs de σ_1 et σ_2 fixées, on peut calculer numériquement le minimum du critère pour diverses valeurs du paramètre α et ainsi tracer la courbe de l'échelle optimale en fonction de α . Cette courbe est représentée sur la figure 2.15 dans le cas où $\sigma_1 = 1$ et $\sigma_2 = 10$. On voit que l'échelle optimale dépend maintenant fortement du critère choisi, pouvant presque passer du simple au triple selon la valeur de α , les petites valeurs de α favorisant le choix d'une plus petite fenêtre que les grandes valeurs de α .

La réalisation de tests numériques sur des signaux réels confirme la dépendance du choix de la taille de fenêtre optimale au critère dès que le signal contient des composantes aux caractéristiques temps-fréquences variées.

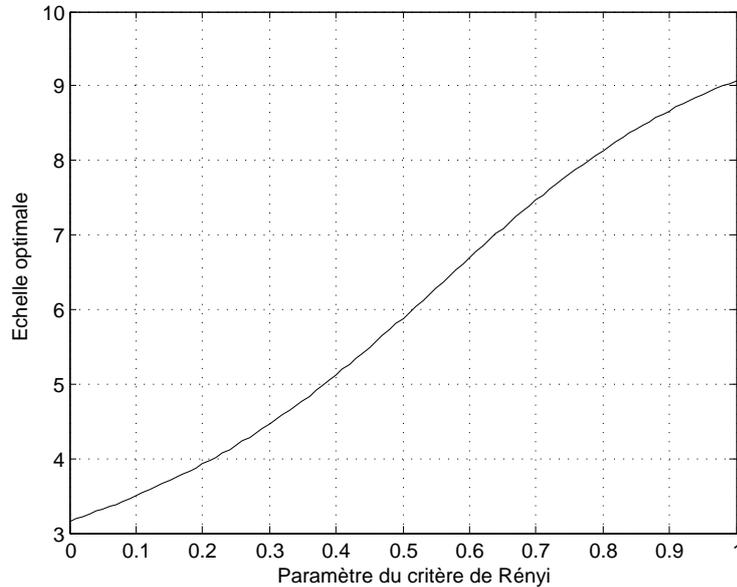


FIG. 2.15 – Influence du critère sur le choix de la fenêtre optimale : valeur de l'échelle σ de la fenêtre d'analyse optimale en fonction du paramètre α de l'entropie de Rényi pour un signal constitué de deux atomes d'échelles $\sigma_1 = 1$ et $\sigma_2 = 10$

Remarque 2.17 *On peut considérer ce phénomène comme un témoignage de l'ambiguïté qui paraît naturelle lorsque l'on cherche à définir une taille de fenêtre optimale pour des sons complexes. En effet, si l'on considère par exemple le cas extrême d'un signal composé d'une somme d'impulsions et de sinusoides, on sait qu'en utilisant une petite fenêtre, on obtient une représentation bien localisée des impulsions, mais une représentation peu localisée des sinusoides, et inversement, avec une grande fenêtre, la représentation des sinusoides est bien localisée, alors que celle des impulsions ne l'est pas. Ainsi, le choix d'une meilleure fenêtre dans un tel cas implique un compromis entre la qualité de représentation des deux types de composantes et la notion de meilleure représentation est relative à ce compromis. On peut ainsi considérer que le choix d'une valeur de α du critère de Rényi correspond à un choix particulier de compromis. Si, dans des cas particuliers, on peut imaginer identifier grâce à une connaissance a priori quel compromis paraît le plus intéressant, dans le cas général, aucun critère entropique ne semble plus justifié qu'un autre, et l'intérêt de l'optimisation globale pour les sons complexes paraît en conséquence très limité.*

Cependant pour certaines composantes temps-fréquence bien localisées, ou une somme de composantes de même caractéristiques temps-fréquence, l'influence du critère sur le choix optimal devient très faible (voir nulle dans le cas de l'atome gaussien unique, par exemple). Ainsi, la notion de meilleure représentation introduite

par les critères entropique ne semble présenter de réel intérêt que pour ce type de composantes. Comme les sons réels contiennent ce type de composantes, il reste possible d'utiliser avec profit l'adaptation automatique de la taille de fenêtre. Pour cela, une optimisation non plus globale mais locale dans le plan temps-fréquence doit être mise en place, afin d'isoler les composantes pour lesquelles la notion de meilleure représentation présente un intérêt. Dans cette situation, les calculs de cette section semblent indiquer que le choix du critère ne sera pas déterminant.

2.3.3 Optimisation locale supervisée

Pour isoler du signal une composante, ou un ensemble de composantes aux caractéristiques temps-fréquence communes, on va travailler localement dans le plan temps-fréquence, en permettant à l'utilisateur de définir sur la transformée de Gabor G_x^g d'un signal x une région Ω du plan temps-fréquence délimitant la où les composantes d'intérêt. Le signal correspondant à cette région est ensuite reconstruit selon la formule de reconstruction partielle suivante :

$$x_{g,\Omega}(t) = \iint_{\Omega} G_x^g(\tau, \nu) g(t - \tau) e^{2i\pi\nu(t-\tau)} d\tau d\nu. \quad (2.144)$$

La fenêtre est alors choisie en réalisant l'adaptation sur le signal reconstruit $x_{g,\Omega}$, c'est à dire que l'on sélectionne la fenêtre g_{opt} pour laquelle la transformée de Gabor de $x_{g,\Omega}$ minimise le critère entropique choisi.

La transformée de Gabor du signal original x est alors recalculée et affichée avec la taille de fenêtre adaptée à la région Ω sélectionnée, c'est à dire que l'on calcule et affiche $G_x^{g_{opt}}$.

Un exemple d'utilisation de cette méthode sur des signaux réels a déjà été présenté dans l'introduction de ce document, en 1.5. On donne ici un exemple sur un signal de synthèse.

Exemple 2.9 *La figure 2.16 montre, sur un exemple didactique constitué d'une somme d'atomes gaussiens, les représentations temps-fréquence obtenues à la suite de la procédure d'adaptation pour trois sélections temps-fréquence différentes. On voit que l'adaptation locale permet bien d'optimiser la parcimonie de la représentation dans la zone sélectionnée. En revanche la représentation peut, comme on s'y attend, devenir moins bonne dans d'autres régions temps-fréquence, puisque la représentation est recalculée avec la fenêtre sélectionnée pour l'ensemble du signal.*

Remarque 2.18 *A première vue, il paraîtrait plus simple de choisir la fenêtre optimale en calculant le critère entropique uniquement dans la région temps-fréquence sélectionnée, à partir d'une transformée de Gabor calculée sur l'ensemble du signal.*

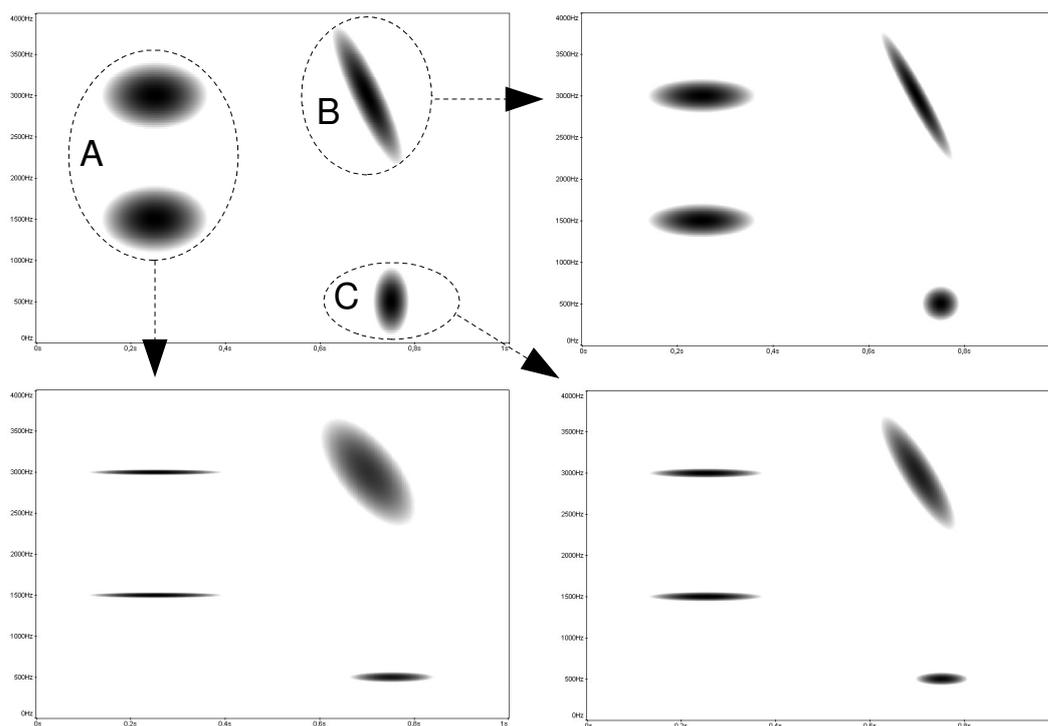


FIG. 2.16 – Exemple d'adaptation locale supervisée : en haut à gauche, transformée de Gabor initiale faisant apparaître trois sélections temps-fréquence notée A,B et C, en bas à gauche, transformée de Gabor obtenue après adaptation de la taille de fenêtre pour la région A, en haut à droite, transformée de Gabor obtenue après adaptation de la taille de fenêtre pour la région B, en bas à droite, transformée de Gabor obtenue après adaptation de la taille de fenêtre pour la région C

Mais avec un telle méthode, même si sur la représentation originale la sélection isole bien une composante, il est possible que, pour une autre fenêtre, l'énergie de composantes voisines apparaisse dans la région sélectionnée et perturbe la valeur du critère.

En utilisant la reconstruction partielle, on s'assure d'isoler totalement la composante sélectionnée. Tant que la composante sélectionnée est bien résolue sur la représentation et qu'elle est totalement sélectionnée, on reconstruit sensiblement le même signal après sélection indépendamment de la fenêtre utilisée pour la représentation initiale. Le processus d'adaptation est alors, dans ces conditions, lui aussi indépendant de la fenêtre initiale.

Remarque 2.19 *Pour une utilisation pratique de l'adaptation locale, on peut mettre simplement en place un outil de sélection de la région d'intérêt à la souris sur la représentation initiale. L'utilisateur peut alors, à chaque fois qu'il souhaite travailler sur un ensemble spécifique de composantes, adapter la représentation à ces composantes et ainsi toujours travailler sur une bonne représentation pour effectuer des sélections et des traitements. Comme, dans la pratique, les fenêtres utilisées sont à support fini, le signal reconstruit à partir de la sélection Ω a un support temporel plus court que celui du signal original et même souvent considérablement plus court. Il est ainsi possible d'effectuer l'adaptation sur le signal reconstruit rapidement. L'utilisation de l'outil d'adaptation est ainsi très souple et facilite notamment l'utilisation des représentations temps-fréquence pour les non-spécialistes en permettant de gérer facilement les problèmes de réglage des paramètres d'analyse. Un tel outil a été intégré au logiciel développé dans le cadre de cette thèse et décrit dans la partie 4.1.3.*

2.4 Représentations adaptatives non supervisées

2.4.1 Objectif : une représentation multifenêtre

Une extension naturelle de l'optimisation locale supervisée consiste à supprimer l'intervention de l'utilisateur et à réaliser l'optimisation locale de manière systématique dans tout le plan temps-fréquence. On construit ainsi une représentation qui utilise des fenêtres d'analyse différentes dans les différentes régions temps-fréquence. La transformée obtenue fait en conséquence partie de la classe des décompositions de Gabor multifenêtres proposée dans [48] qui consiste à décomposer le signal simultanément sur plusieurs repères de Gabor. Cependant, dans notre approche, on s'intéresse aux décompositions multifenêtres réduites [12], c'est à dire qu'au lieu d'utiliser l'ensemble des atomes des différents repères de Gabor utilisés, on utilise un nombre limité d'atomes de chaque repère en choisissant les atomes bien adaptés au signal de manière automatique.

Une manière simple de mettre en place cette démarche d'optimisation non-supervisée est de partitionner le plan temps-fréquence en rectangles $(\square_{m,n})_{(m,n) \in \mathbb{Z}^2}$ de largeur temporelle $\Delta_\tau > 0$ et de largeur fréquentielle $\Delta_\nu > 0$ définis par

$$\square_{m,n} = [(m - \frac{1}{2})\Delta_\tau, (m + \frac{1}{2})\Delta_\tau) \times [(n - \frac{1}{2})\Delta_\nu, (n + \frac{1}{2})\Delta_\nu), \quad (2.145)$$

qui constituent bien une partition du plan temps-fréquence :

$$\mathbb{R}^2 = \bigcup_{(m,n) \in \mathbb{Z}^2} \square_{m,n}. \quad (2.146)$$

En définissant ensuite une version localisée des critères de parcimonie présentés en 2.3.1, on choisit dans chaque rectangle $(\square_{m,n})_{(m,n) \in \mathbb{Z}^2}$ la meilleure fenêtre. Le signal est alors représenté dans le rectangle $(\square_{m,n})_{(m,n) \in \mathbb{Z}^2}$ par les coefficients obtenus dans cette région avec la meilleure fenêtre.

Exemple 2.10 *La figure 2.17 illustre la démarche proposée à partir de l'exemple de la figure 2.16 : les quatre transformées de Gabor apparaissant sur la figure 2.16 et reproduites en haut de la figure 2.17 sont partitionnées en rectangle temps-fréquence identiques, puis on construit la représentation du bas de la figure 2.17 en sélectionnant dans chaque rectangle la meilleure représentation.*

L'approche que nous venons d'évoquer n'est pas directement exploitable pour la transformation des sons, car elle ne permet pas la reconstruction du signal à partir de sa transformée. En effet, si on utilise pour chaque rectangle temps-fréquence la méthode de reconstruction utilisée pour une transformée de Gabor complète, des erreurs de reconstruction vont apparaître dans les zones de transitions entre les rectangles utilisant des tailles de fenêtre différentes.

Il est en conséquence nécessaire d'adapter la construction pour pouvoir l'utiliser dans le cadre de l'application au traitement des sons. Pour cela, un algorithme itératif, que nous désignerons sous le nom de "puzzle temps-fréquence", peut être mis en place pour corriger les erreurs de reconstruction du signal et obtenir une représentation satisfaisante.

L'idée générale est très simple : après sélection de la meilleure fenêtre dans chaque rectangle, on applique la formule d'inversion de Gabor en feignant d'ignorer que la reconstruction n'est pas exacte, puis en calculant la différence entre le signal initial et le signal ainsi reconstruit, on obtient un résidu sur lequel le même traitement peut être appliqué, et on réitère ainsi le même traitement sur le dernier résidu obtenu.

Il s'avère que cette itération converge (bien que nous n'en ayons pas la preuve mathématique).

Cet algorithme est présenté plus en détails (avec quelques variantes) ci-dessous.

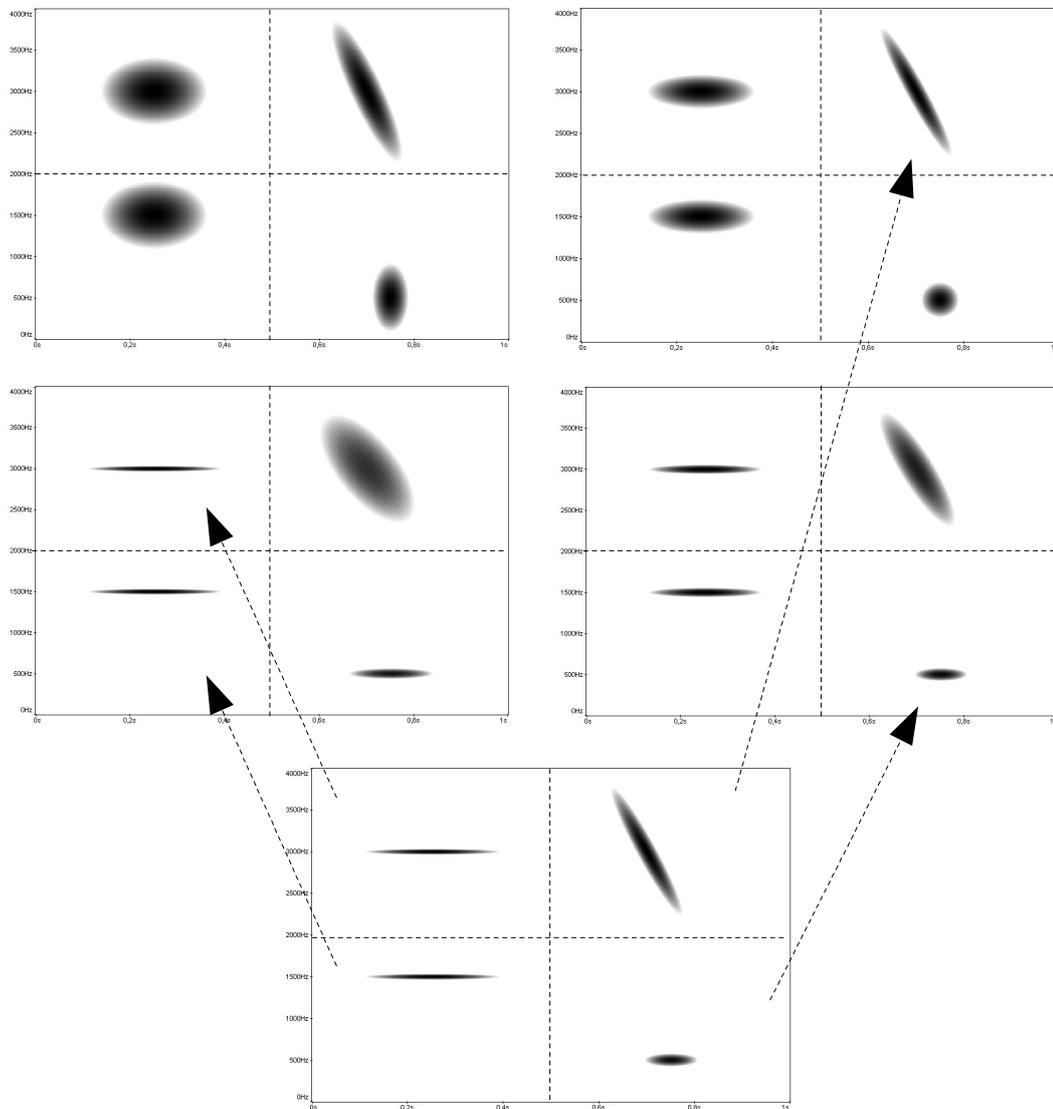


FIG. 2.17 – Principe de l'adaptation locale non supervisée : en haut, les transformées de Gabor obtenues avec quatre tailles de fenêtres différentes (il s'agit des transformées apparaissant sur la figure 2.16) sont partitionnées en rectangles temps-fréquence, puis, en bas, on construit une représentation en prenant dans chaque rectangle la meilleure des quatre fenêtres au sens du critère de parcimonie

2.4.2 Puzzle temps-fréquence

Dans cette section, on utilise une version discrétisée des critères entropiques définis précédemment pour choisir la fenêtre optimale dans chaque rectangle temps-fréquence. Plus précisément, pour $\alpha \in (0, 1)$ fixé, on calcule l'entropie de Rényi pour un vecteur $u \in \mathbb{C}^N$ par :

$$H_\alpha(u) = \frac{1}{1-\alpha} \log_2 \left(\sum_{n=0}^{N-1} \left(\frac{|u_n|^2}{\|u\|^2} \right)^\alpha \right). \quad (2.147)$$

Puzzle temps-fréquence, variante 1

On considère R fenêtres g^r de $L^2(\mathbb{R})$, à partir desquelles on définit R repères de Gabor $(g_{mn}^r)_{(m,n) \in \mathbb{Z}^2}$ tels que

$$\forall (m, n) \in \mathbb{Z}^2, \forall t \in \mathbb{R}, \quad g_{mn}^r(t) = g^r(t - m\tau_r) e^{i2\pi n\nu_r t}, \quad (2.148)$$

avec τ_r et ν_r respectivement les pas de discrétisation temporelle et fréquentielle du repère lié à g^r de la grille de discrétisation $\mathcal{L}^r = \tau_r \mathbb{Z} \times \nu_r \mathbb{Z}$ du repère r .

On utilise le pavage du plan temps-fréquence présenté précédemment en définissant la famille de pavés rectangulaires $(\square_{(s)})_{(s) \in \mathbb{Z}^2}$ par la relation :

$$\square_{(s)} = \square_{m,n} = [(m - \frac{1}{2})\Delta_\tau, (m + \frac{1}{2})\Delta_\tau] \times [(n - \frac{1}{2})\Delta_\nu, (n + \frac{1}{2})\Delta_\nu], \quad (2.149)$$

avec Δ_τ la largeur temporelle des rectangles et Δ_ν leur largeur fréquentielle.

Pour chaque repère, on identifie les atomes qui se trouvent dans un rectangle donné, c'est à dire qu'on définit pour chaque grille \mathcal{L}^r le sous-ensemble $\mathcal{L}^{r,s} = \mathcal{L}^r \cap \square_{(s)}$ des éléments de la grille appartenant au rectangle $\square_{(s)}$.

Dans chaque rectangle $\square_{(s)}$, on détermine quelle est la fenêtre la mieux adaptée pour représenter le signal x . La fenêtre optimale $g^{r(x,s)}$ est définie par :

$$r(x, s) = \arg \min_{r=0 \dots R-1} H_\alpha(v(x, r, s)), \quad (2.150)$$

avec $v(x, r, s) = \{\langle x, g_{mn}^r \rangle, (m, n) \in \mathcal{L}^{r,s}\}$ l'ensemble des coefficients correspondant aux atomes du repère r qui se trouvent dans le rectangle $\square_{(s)}$.

On note $H(x, s)$ la valeur optimale de l'entropie obtenue pour le rectangle $\square_{(s)}$ définie par :

$$H(x, s) = H(v(x, r(x, s), s)). \quad (2.151)$$

La première approximation $x^{(1)}$ du signal x est obtenue par sommation des reconstructions partielles à partir des coefficients optimaux dans chaque rectangle, selon la relation :

$$x^{(1)} = \sum_s \sum_{\lambda \in \mathcal{L}^{r(x,s),s}} \langle x, g_\lambda^{r(x,s)} \rangle \tilde{g}_\lambda^{r(x,s)}. \quad (2.152)$$

Comme indiqué précédemment, la reconstruction obtenue par cette méthode n'est pas exacte. Le résidu $\mathcal{R}^1(x)$, correspondant à l'erreur de reconstruction, est défini par :

$$\mathcal{R}^1(x) = x - x^{(1)}. \quad (2.153)$$

La procédure de décomposition est alors itérée sur le résidu, selon les relations :

$$x^{(k)} = \sum_s \sum_{\lambda \in \mathcal{L}(x,k,s)} \left\langle \mathcal{R}^{k-1}(x), g_\lambda^{r(x,k,s)} \right\rangle \tilde{g}_\lambda^{r(x,k,s)}, \quad (2.154)$$

$$\mathcal{R}^k(x) = \mathcal{R}^{k-1}(x) - x^{(k)}, \quad (2.155)$$

où $r(x, k, s) = r(\mathcal{R}^{k-1}(x), s)$ est le numéro du repère sélectionné dans le rectangle $\square_{(s)}$ à l'itération k et $\mathcal{L}(x, k, s) = \mathcal{L}^{r(x,k,s),s}$ les points de la grille de repère sélectionné appartenant au rectangle $\square_{(s)}$.

À l'itération K , on obtient ainsi une décomposition du signal x sous la forme :

$$x = \sum_{k=1}^K x^{(k)} + \mathcal{R}^k(x). \quad (2.156)$$

Le processus itératif s'arrête lorsque le résidu $\mathcal{R}^k(x)$ est jugé suffisamment petit.

Remarque 2.20 *L'étude théorique de cet algorithme est très complexe et nous ne sommes pas parvenu à prouver sa convergence. Cependant nous étudierons numériquement dans la suite le comportement de l'algorithme sur divers exemples qui font tous apparaître une convergence rapide.*

Le schéma itératif proposé ici permet dans la pratique d'obtenir une approximation atomique du signal x avec une erreur d'approximation aussi petite que souhaitée. Un des intérêts de la démarche est qu'elle permet une interprétation multicouche de la décomposition obtenue que nous allons maintenant détailler.

Décomposition multicouche

Si on se limite à une reconstruction partielle en ne faisant intervenir que les atomes d'un repère r fixé, on isole les composantes du signal qui sont "bien représentées" avec le repère r . En effectuant cette reconstruction partielle pour chacun des repères, on obtient une représentation multicouche du signal dans laquelle chaque couche correspond à un des R repères utilisés.

Remarque 2.21 *Cette décomposition est en particulier intéressante dans les cas le plus simple où seulement deux repères sont utilisés (cas $R = 2$). En effet, en choisissant deux fenêtres suffisamment différentes, une petite et une grande, on peut alors isoler des composantes aux caractéristiques temps-fréquences bien différenciées :*

- d'une part des composantes très localisées dans le temps et bien représentées avec la petite fenêtre qui correspondent aux composantes transitoires du signal,
- d'autre part des composantes quasi-stationnaires bien localisées en fréquence et bien représentées avec la grande fenêtre qui correspondent aux composantes tonales du signal.

La distinction entre ces deux types de composantes est particulièrement intéressante dans un contexte de traitement audionumérique, car les deux couches obtenues dans la décomposition ont un "sens perceptif" bien différencié, ce qui n'est généralement pas le cas lorsqu'on utilise plus de deux fenêtres.

En conséquence, le cas $R = 2$ est le cas présentant le plus grand intérêt pour les applications. Il s'agit également du cas le plus léger en coût de calcul et le plus simple à décrire.

En conséquence de la remarque précédente, dans la suite, **nous nous limiterons à l'étude du cas où seulement deux repères de Gabor sont utilisés** (cas $R = 2$). L'extension à plus de deux repères peut être aisément déduite.

Pour une itération donnée k et un repère ρ fixé, on note $S(x, k, \rho) = \{s : r(x, k, s) = \rho\}$ l'ensemble des rectangles temps-fréquence dans lesquels le repère ρ a été sélectionné lors de l'itération k . On peut alors réécrire l'expression (2.154) de façon à séparer les atomes des deux repères, sous la forme :

$$\mathcal{R}^k(x) = \sum_{\rho=0}^1 \sum_{s \in S(x, k, \rho)} \sum_{\lambda \in \mathcal{L}(x, k, s)} \langle \mathcal{R}^{k-1}(x), g_\lambda^\rho \rangle \tilde{g}_\lambda^\rho. \quad (2.157)$$

On en déduit la décomposition multicouche suivante du signal :

$$x = \ell_0(x) + \ell_1(x) + \mathcal{R}^K(x), \quad (2.158)$$

où le résidu $\mathcal{R}^K(x)$ reste défini comme précédemment, et $\ell_0(x)$ et $\ell_1(x)$ sont les deux couches correspondant aux deux fenêtres définies, avec $\rho \in \{0, 1\}$, par :

$$\ell_\rho(x) = \sum_{k=1}^K \sum_{s \in S(x, k, \rho)} \sum_{\lambda \in \mathcal{L}(x, k, s)} \langle \mathcal{R}^{k-1}(x), g_\lambda^\rho \rangle \tilde{g}_\lambda^\rho. \quad (2.159)$$

Dans cette formulation de l'algorithme, on voit apparaître la structure de l'algorithme dans le cas à 2 fenêtres, résumée sur la figure 2.18. On voit que la structure de l'algorithme reste simple et son implémentation ne nécessite que le calcul de la transformée de Gabor et la reconstruction associée avec les deux tailles de fenêtres utilisées, ainsi que les calculs du critère entropique de sélection des coefficients.

Exemple 2.11 La figure 2.19 montre la décomposition obtenue avec l'algorithme sur un son de glockenspiel. L'image du haut représente la transformée de Gabor du

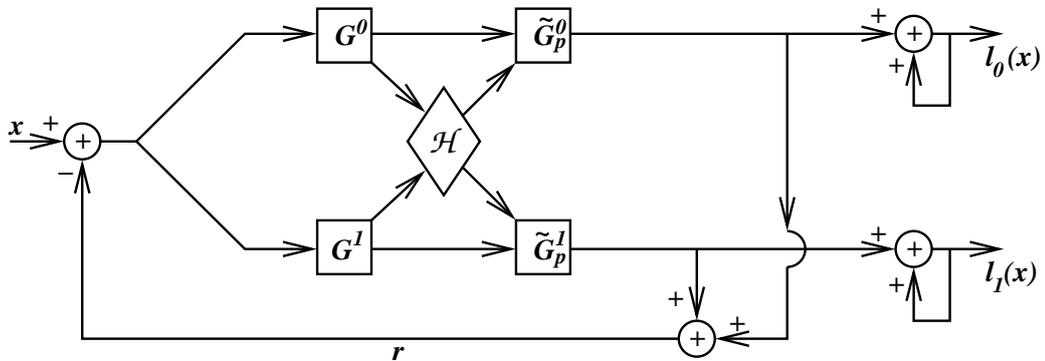


FIG. 2.18 – Structure de l’algorithme de puzzle temps-fréquence (variante 1) utilisant deux repères de Gabor : G^0 et G^1 représentent le calcul de la transformée de Gabor avec les deux repères, \tilde{G}^0 et \tilde{G}^1 représentent la resynthèse partielle à partir des coefficients temps-fréquence sélectionnés, \mathcal{H} représente le calcul d’entropie et la décision associée

son original, en bas à gauche, la représentation temps-fréquence obtenue à partir des coefficients de la couche associée à la petite fenêtre, en bas à droite, la représentation temps-fréquence obtenue à partir des coefficients de la couche associée à la grande fenêtre. On voit que comme on l’attendait, la couche correspondant à la petite fenêtre contient les zones transitoires du signal, c’est à dire les attaques des notes dans le cas du glockenspiel, et la couche correspondant à la grande fenêtre contient les partiels sinusoidaux des notes. Cependant, on voit sur la représentation associée à la couche “transitoire” correspondant à la petite fenêtre que de l’énergie correspondant aux partiels est parfois présente.

Remarque 2.22 La figure 2.20 permet de comprendre pourquoi une partie de l’énergie des partiels est présente dans la couche correspondant à la petite fenêtre. Elle montre le module au carré de la transformée de Fourier d’un sinus à 1000 Hz fenêtré avec deux fenêtres de tailles différentes. Ces courbes correspondent donc aux valeurs des modules au carré des coefficients obtenues pour des composantes sinusoidales sur la transformée de Gabor calculée avec deux tailles de fenêtres différentes. Supposons que les limites fréquentielles des pavés temps-fréquence utilisés pour le calcul des entropies soient 970 Hz, 990 Hz, 1010 Hz et 1030 Hz. Dans les rectangles couvrant la plage de fréquence de 990 Hz à 1010 Hz, on a bien comme on l’attend une distribution des coefficients plus piquée pour la grande fenêtre, ce qui entraîne la sélection de la grande fenêtre par le critère entropique. La couche correspondant à la grande fenêtre récupère ainsi quasiment l’intégralité de l’énergie correspondant au sinus. En revanche, pour les rectangles couvrant les plages de fréquences

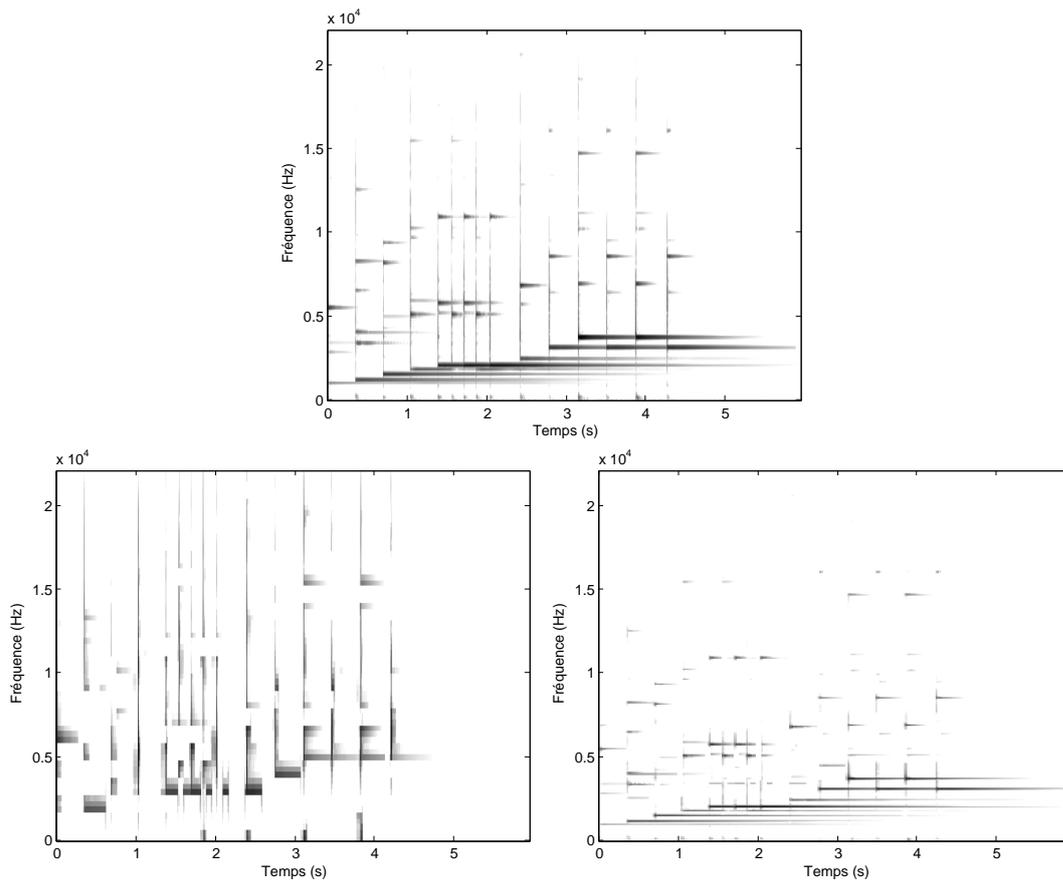


FIG. 2.19 – Décomposition obtenue avec l’algorithme de puzzle temps-fréquence (variante 1) : sur un son de glockenspiel, en haut, transformée de Gabor du signal original avec une taille de fenêtre intermédiaire, en bas à gauche, représentation de la couche de la décomposition pour la petite fenêtre, en bas à droite, représentation de la couche de la décomposition pour la grande fenêtre

de 970 Hz à 1010 Hz d'une part et de 1010 Hz à 1030 Hz d'autre part, on se trouve dans une zone où la distribution des coefficients pour la grande fenêtre est déjà très plate, alors que l'inflexion est encore importante pour la petite fenêtre, ce qui entraîne la sélection de la petite fenêtre par le critère. On récupère donc à nouveau une partie de l'énergie correspondant au sinus, mais cette fois dans la couche associée à la petite fenêtre, ce qui fausse la décomposition.

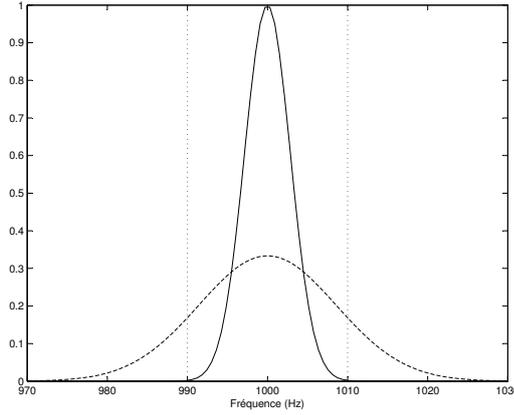


FIG. 2.20 – Module au carré de la transformée de Fourier d'un sinus à 1000 Hz fenêtré pour deux tailles de fenêtres : en trait plein, avec un fenêtrage par une grande fenêtre, en trait discontinu, avec un fenêtrage par une petite fenêtre

Pour éviter ce problème et faire en sorte que l'estimation de la composante transitoire ne soit pas perturbée par présence de la composante tonale et inversement, on propose une deuxième variante de l'algorithme dans laquelle on estime alternativement l'une des deux composantes puis l'autre.

Puzzle temps-fréquence, variante 2

La première étape consiste comme précédemment à choisir le repère optimal dans chaque rectangle $\square_{(s)}$ et à définir l'entropie correspondante :

$$\begin{cases} r(x, s) &= \arg \min_{r=0,1} H(\alpha(x, r, s)) , \\ H(x, s) &= H(\alpha(x, r(x, s), s)) , \end{cases} \quad (2.160)$$

Cependant, la première approximation est obtenue en n'utilisant que les composantes correspondant à la première fenêtre :

$$x^{(1;0)} = \sum_{s:r(x,s)=0} \sum_{\lambda \in \mathcal{L}^{0,s}} \langle x, g_\lambda^0 \rangle \tilde{g}_\lambda^0. \quad (2.161)$$

Ceci produit un résidu intermédiaire :

$$\mathcal{R}^{\frac{1}{2}}(x) = x - x^{(1;0)}, \quad (2.162)$$

et les transformées de Gabor avec les deux repères sont recalculées pour ce résidu. Le repère optimal est choisi dans chaque rectangle comme précédemment, et on effectue une reconstruction partielle n'utilisant cette fois que les composantes correspondant à la deuxième fenêtre :

$$x^{(1;1)} = \sum_{s:r(\mathcal{R}^{\frac{1}{2}},s)=1} \sum_{\lambda \in \mathcal{L}^{1,s}} \langle \mathcal{R}^{\frac{1}{2}}(x), g_{\lambda}^1 \rangle \tilde{g}_{\lambda}^1. \quad (2.163)$$

Ceci produit un nouveau résidu :

$$\mathcal{R}^1(x) = \mathcal{R}^{\frac{1}{2}}(x) - x^{(1;1)} = x - x^{(1;0)} - x^{(1;1)}, \quad (2.164)$$

auquel on va à nouveau appliquer ces deux étapes.

En itérant cette procédure, on met en place la récurrence suivante :

$$x^{(k;0)} = \sum_{s:r(\mathcal{R}^{k-1},s)=0} \sum_{\lambda \in \mathcal{L}^{0,s}} \langle \mathcal{R}^{k-1}(x), g_{\lambda}^0 \rangle \tilde{g}_{\lambda}^0 \quad (2.165)$$

$$x^{(k;1)} = \sum_{s:r(\mathcal{R}^{k-\frac{1}{2}},s)=1} \sum_{\lambda \in \mathcal{L}^{1,s}} \langle \mathcal{R}^{k-\frac{1}{2}}(x), g_{\lambda}^1 \rangle \tilde{g}_{\lambda}^1, \quad (2.166)$$

les résidus étant définis par :

$$\mathcal{R}^{k-\frac{1}{2}}(x) = \mathcal{R}^{k-1}(x) - x^{(k;0)}, \quad (2.167)$$

$$\mathcal{R}^k(x) = \mathcal{R}^{k-\frac{1}{2}}(x) - x^{(k;1)}. \quad (2.168)$$

Après K itérations, on aboutit à une décomposition du signal x sous la forme :

$$x = \sum_{k=1}^K (x^{(k;0)} + x^{(k;1)}) + \mathcal{R}^K(x), \quad (2.169)$$

qui peut comme précédemment se réécrire sous forme de décomposition multicouche :

$$x = \ell_0(x) + \ell_1(x) + \mathcal{R}^K(x), \quad (2.170)$$

avec

$$\ell_0(x) = \sum_{k=1}^K x^{(k;0)} \quad (2.171)$$

$$\ell_1(x) = \sum_{k=1}^K x^{(k;1)}, \quad (2.172)$$

l'interprétation des deux couches étant la même que plus haut.

Exemple 2.12 La figure 2.21 permet de vérifier que le problème observé sur la figure 2.19 est résolu avec cette variante de l'algorithme. En reconstruisant à la première demi-itération uniquement la couche correspondant à la grande fenêtre, la quasi-totalité de l'énergie des composantes sinusoïdales est extraite du signal. Les sinusoïdes ne peuvent alors plus perturber l'estimation de la couche correspondant à la petite fenêtre qui est faite par la suite.

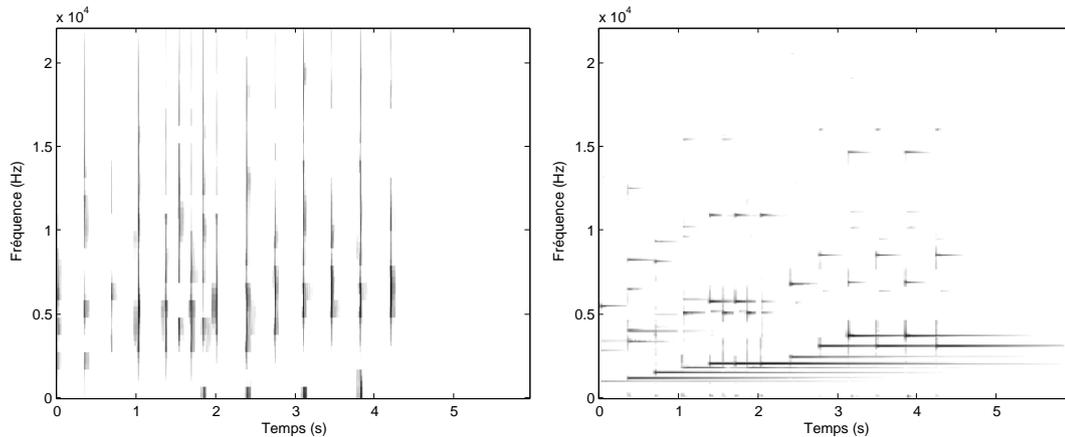


FIG. 2.21 – Décomposition obtenue avec l'algorithme de puzzle temps-fréquence (variante 2) : sur le son de glockenspiel utilisé pour la figure 2.19, à gauche, représentation de la couche de la décomposition pour la petite fenêtre, à droite, représentation de la couche de la décomposition pour la grande fenêtre

2.4.3 Étude numérique de l'algorithme

Le comportement de l'algorithme est particulièrement difficile à étudier théoriquement, du fait de sa forte dépendance au signal et de la complexité du critère de choix utilisé. Ainsi il ne nous a pas été possible de prouver sa convergence. Cependant, un certain nombre de constatations et de tendances peuvent être observées à l'aide de tests numériques sur quelques signaux bien choisis. Pour cela, on étudie la convergence de l'algorithme pour un son donné en traçant le rapport signal à bruit défini comme le rapport de l'énergie du signal analysé sur l'énergie du résidu en fonction du numéro d'itération de l'algorithme. Pour faciliter la lecture des courbes obtenues, on affiche le rapport signal à bruit en dB. Toutes ces courbes sont obtenues en utilisant une décomposition avec deux tailles de fenêtres différentes (cas $R = 2$), sauf mention contraire.

Influence du type de signal

En premier lieu, la figure 2.22 montre que la vitesse de convergence est différente selon la nature du signal analysé. La figure représente le rapport signal à bruit obtenu pour les mêmes paramètres d'analyse sur trois sons différents, un bruit blanc, un son de glockenspiel, et un son constitué d'une somme de sinus et d'impulsions.

La convergence est plus lente pour le bruit blanc, bien qu'elle reste très satisfaisante, puisqu'un rapport signal à bruit de plus de 100 dB, que l'on peut considérer convenable pour les applications audionumériques, est atteint en 13 itérations. Cette moins bonne efficacité de l'algorithme s'explique par le fait que la parcimonie de la représentation du bruit n'est bonne ni avec la petite fenêtre, ni avec la grande fenêtre et les erreurs lors des passages d'un repère à l'autre entre les différents rectangles temps-fréquence sont importantes.

La convergence est la meilleure pour le son constitué d'une somme de sinus et d'impulsions, conçu spécifiquement pour avoir une représentation parcimonieuse avec cette décomposition. En effet, les impulsions sont bien représentées avec la petite fenêtre et les sinus avec la grande fenêtre. Les erreurs dues au changement de repères restent limitées car au sein de chaque rectangle temps-fréquence, l'énergie est très bien localisée.

Le son de glockenspiel aboutit à une convergence intermédiaire. Le son est bien adapté à l'algorithme de par sa structure constituée d'attaques très marquées et de partiels quasi-sinusoidaux, ce qui explique la meilleure convergence que pour le bruit blanc.

On peut toutefois noter que la différence se fait sur les premières itérations, les trois courbes évoluant ensuite de façon quasi-parallèle. Ceci milite en faveur d'une convergence exponentielle, comparable à celle de l'algorithme de matching pursued en dimension finie.

Influence de la taille des rectangles

La vitesse de convergence est également dépendante des paramètres d'analyse utilisés pour réaliser la décomposition. Sur la figure 2.23, on peut voir l'influence de la taille des rectangles utilisés pour découper le plan temps-fréquence. La taille des pavés est exprimée de la manière suivante : un pavé de taille $m \times n$ contient m points de discrétisation temporelle pour la grande fenêtre et n points de discrétisation fréquentielle pour la petite fenêtre.

On voit que la vitesse de convergence augmente avec la taille des rectangles. Ceci est dû au fait que l'énergie présente dans le résidu provient essentiellement des erreurs de reconstructions qui apparaissent aux interfaces entre des rectangle temps-fréquence pour lesquelles des tailles de fenêtres différentes ont été choisies. Lorsque la surface des rectangles augmente, la taille relative de la zone où se créent ces erreurs

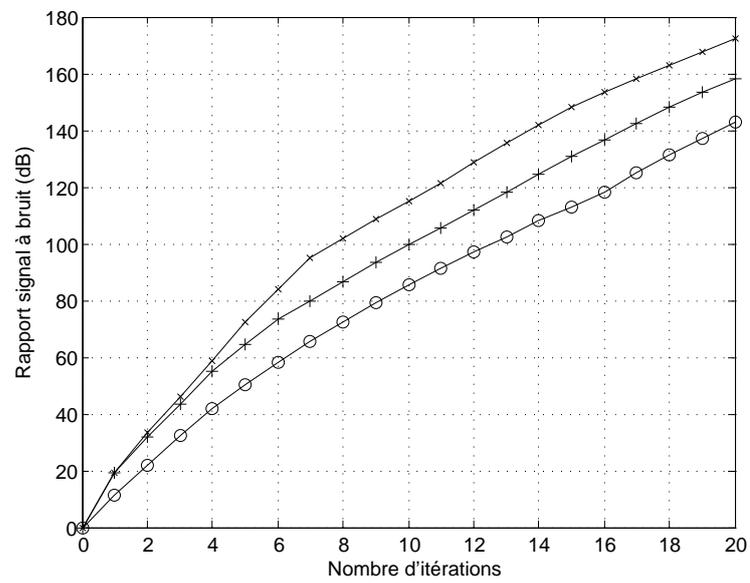


FIG. 2.22 – Influence du type de signal sur la convergence de l'algorithme de puzzle temps-fréquence (variante 1) : rapport signal à bruit en fonction du nombre d'itérations, \circ : pour un bruit blanc, $+$: pour un son de glockenspiel, \times : pour une somme de sinus et d'impulsions

diminue, ce qui implique une énergie plus faible dans le résidu. Dans le cas extrême, le rectangle couvre toute l'énergie du signal. On retrouve alors le cas de l'optimisation globale de la fenêtre, et comme on utilise des repères avec reconstruction exacte, une seule itération de l'algorithme est alors nécessaire. Ainsi, on peut être tenté d'utiliser une grande taille de rectangle, pour obtenir une convergence rapide. Cependant, ceci va à l'encontre de l'objectif initial qui était la localisation dans des régions suffisamment "concentrées" du plan temps-fréquence. Comme souvent, il y a là un compromis à trouver pour l'utilisateur. L'étude théorique de ce compromis semble ardue.

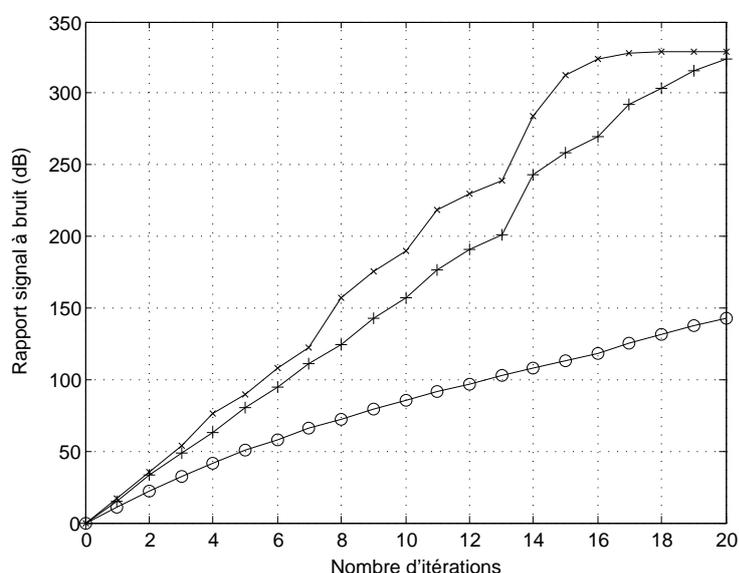


FIG. 2.23 – Influence de la taille des rectangles sur la convergence de l'algorithme de puzzle temps-fréquence (variante 1) : rapport signal à bruit en fonction du nombre d'itérations pour un bruit blanc, \circ : rectangles de taille 1×1 , $+$: rectangles de taille 3×3 , \times : rectangles de taille 5×5

Comparaison des deux variantes de l'algorithme

La figure 2.24 permet de comparer la convergence des deux variantes de l'algorithme sur le son de glockenspiel. Nous avons été amenés à proposer la variante 2 en raison du problème apparaissant dans la décomposition sur les partiels et décrit dans la remarque 2.22. On peut constater que l'effet de ce problème apparaît sur les courbes de convergence lors des premières itérations. Ainsi, avec la variante 2, en évitant le problème sur les partiels, on limite fortement l'erreur lors de la première

itération.

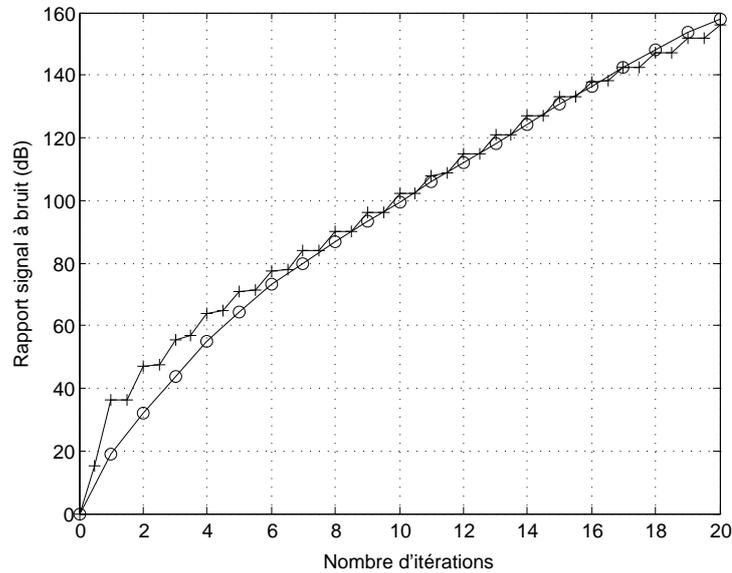


FIG. 2.24 – Comparaison de la convergence des deux variantes de l'algorithme de puzzle temps-fréquence : rapport signal à bruit en fonction du nombre d'itérations pour un son de glockenspiel, ○ : variante 1, + : variante 2 avec grande fenêtre en premier

Convergence avec trois repères

Pour vérifier la convergence de l'algorithme dans le cas où l'on utilise plus de deux tailles de fenêtre simultanément, le cas à trois fenêtres a été implémenté (cas $R = 3$). La figure 2.25 montre les résultats de convergence obtenus dans ce cas sur un bruit blanc et un son de glockenspiel. La convergence est là aussi rapide et tout à fait comparable à celle observée pour deux fenêtres sur la figure 2.22. Cependant, l'intérêt pratique de la version avec trois tailles de fenêtres est limitée dans le cadre général, car les couches obtenues ne sont pas aussi facilement interprétables que dans le cas à deux tailles de fenêtres.

Remarque 2.23 *On peut imaginer de nombreuses variantes de l'algorithme, par exemple en utilisant des décompositions sur des atomes chirpés. On risque cependant de se retrouver face au même problème que dans le cas à trois fenêtres : les différentes couches de la transformée risquent alors de ne pas être facilement interprétable dans un cadre général (mais elles peuvent être bien adaptées à des cas particuliers).*

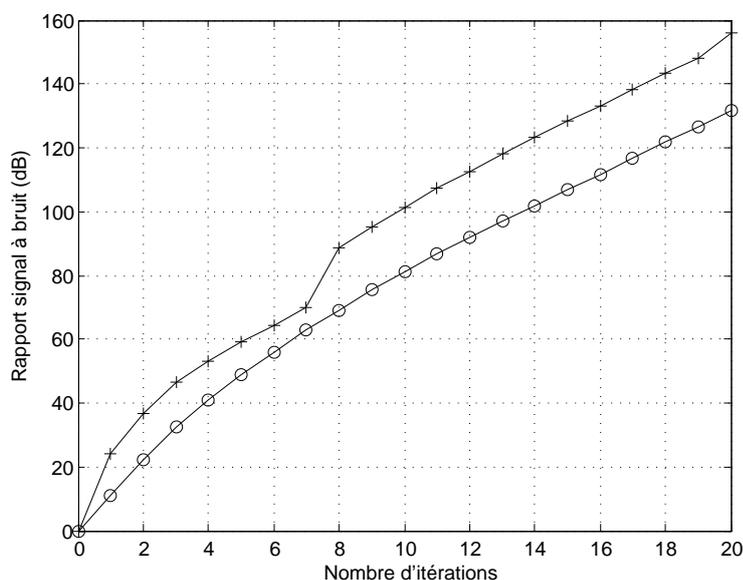


FIG. 2.25 – Convergence de l’algorithme de puzzle temps-fréquence (variante 1) utilisant trois tailles de fenêtres : \circ pour un bruit blanc, $+$ pour un son de glockenspiel

2.4.4 Puzzle temps-fréquence avec seuillage sur les valeurs d’entropie

Description de l’algorithme

L’algorithme de puzzle temps-fréquence est basé sur l’idée qu’en choisissant dans chaque région du plan temps-fréquence la représentation la plus parcimonieuse, on construit une représentation finale ayant une bonne parcimonie. Cependant, pour certaines composantes, même en choisissant la meilleure représentation il n’est pas possible d’obtenir une “bonne” représentation, c’est à dire une représentation suffisamment parcimonieuse. C’est par exemple le cas avec un bruit blanc pour lequel la parcimonie de la transformée de Gabor est mauvaise indépendamment de la fenêtre utilisée.

Ceci pose en particulier des problèmes dans l’interprétation multicouche de la transformée. En effet, dans le cas où l’on utilise deux fenêtres, si l’on peut bien s’attendre à ce que les composantes transitoires soient bien représentées avec la petite fenêtre et que les composantes tonales soient bien représentées avec la grande fenêtre, il n’est pas évident de savoir comment seront représentées les composantes bruitées par exemple. Ces dernières sont en fait représentées avec les deux tailles de fenêtres, la représentation étant meilleure avec la petite fenêtre dans certaines zones et avec la grande dans d’autres, en restant peu parcimonieuse dans tous les

cas. Certaines composantes se trouvent ainsi “partagées” entre les deux couches de la décomposition, ce qui rend l’interprétation de la décomposition moins directe.

Dans ce contexte, on peut souhaiter que les composantes pour lesquelles la représentation n’est pas suffisamment parcimonieuse ne soient pas incluses dans les couches de la décomposition et soient conservées dans le résidu à chaque itération. Compte tenu de la structure de l’algorithme, ceci peut-être mis en place de façon simple en considérant les valeurs d’entropie obtenues dans chaque rectangle temps-fréquence.

Dans l’algorithme, seul le niveau relatif des valeurs d’entropie est pris en compte pour sélectionner la meilleure fenêtre, c’est à dire celle pour laquelle la valeur d’entropie est la plus petite. Mais la valeur en elle-même de l’entropie donne aussi une mesure absolue du niveau de parcimonie de la représentation et il est ainsi possible, en comparant cette valeur à un seuil, de s’assurer que la parcimonie avec la meilleure fenêtre est satisfaisante.

On peut ainsi modifier les deux variantes de l’algorithme de manière à n’intégrer les composantes d’un rectangle temps-fréquence dans la couche correspondant à la fenêtre optimale qu’à la condition que l’entropie correspondante soit inférieure à un seuil fixé, les composantes étant conservées dans le résidu dans le cas contraire.

Plus précisément, pour la première variante de l’algorithme, cela revient à introduire un seuil τ et à modifier les calculs de la première itération en remplaçant les relations (2.152) et (2.153) par :

$$x_\tau^{(1)} = \sum_{s:H(x,s)\leq\tau} \sum_{\lambda\in\mathcal{L}^r(x,s),s} \langle x, g_\lambda^{r(x,s)} \rangle \tilde{g}_\lambda^{r(x,s)} \quad (2.173)$$

$$\mathcal{R}_\tau^1(x) = x - x^{(1)}. \quad (2.174)$$

On introduit de la même manière cette condition de seuil dans les relations d’itération de l’algorithme.

On obtient ainsi après K itérations une décomposition qui garde la forme précédente :

$$x = \sum_{k=1}^K x_\tau^{(k)} + \mathcal{R}_\tau^k(x), \quad (2.175)$$

que l’on peut également l’exprimer sous une forme multicouche en regroupant les composantes correspondant à chaque fenêtre :

$$x = \ell_{0,\tau}(x) + \ell_{1,\tau}(x) + \mathcal{R}_\tau^K(x). \quad (2.176)$$

Cependant dans ce cas il n’y a plus convergence vers 0 du résidu, puisque les composantes mal représentée avec les deux tailles de fenêtres y sont conservées. Ainsi le résidu n’est plus négligeable mais devient une couche à part entière de la décomposition. L’utilisateur peut ainsi par la suite choisir pour ces composantes un mode de représentation bien approprié.

Remarque 2.24 *Le même principe de seuillage peut être mis en place pour la variante 2 de l'algorithme de puzzle temps-fréquence. Cependant l'intérêt d'utiliser la variante 2 avec seuillage est limité, car l'introduction du seuillage dans la variante 1 permet déjà de résoudre le problème évoqué sur l'exemple de la figure 2.19. Les résultats obtenus pour les deux variantes avec seuillage sont très proches, alors que la charge de calcul est plus élevée pour la variante 2.*

Exemple 2.13 *La figure 2.26 présente la décomposition obtenue sur un exemple didactique constitué d'une somme de sinus, d'impulsions et d'un bruit blanc. Comme on l'attend, la couche correspondant à la petite fenêtre contient (approximativement) les rectangles contenant les impulsions, la couche correspondant à la grande fenêtre contient les rectangles contenant les sinus et le résidu contient le bruit (exceptée l'énergie du bruit qui a été sélectionné avec les autres composantes dans les deux couches précédentes).*

Exemple 2.14 *La figure 2.27 présente la décomposition obtenue pour un signal de parole (le mot "test"). L'algorithme permet de séparer de manière assez précise les différentes lettres du mot : les "t" apparaissent dans la couche correspondant à la petite fenêtre, le "e" dans la couche correspondant à la grande fenêtre, et le "s" dans le résidu. Il faut noter que le résultat obtenu est assez sensible au calage temporel du signal et aux valeurs de seuil utilisées.*

Remarque 2.25 *Pour faciliter le réglage du seuil par l'utilisateur, pour qui les valeurs d'entropie risquent d'être peu "parlantes", car difficilement interprétables, l'approche statistique suivante peut être mise en place : la distribution des valeurs d'entropie est estimée numériquement à partir de la transformée de Gabor d'un bruit blanc de référence pour chaque fenêtre, puis le seuil τ est réglé par l'utilisateur en fixant un niveau de signification par rapport au bruit. En d'autres termes, on rejette un rectangle temps-fréquence lorsque la probabilité pour que le signal présent dans ce rectangle soit du bruit dépasse un niveau fixé.*

Évolution du rapport signal à bruit

Comme on l'a déjà indiqué, le résidu ne converge plus vers zéro lorsque l'on introduit le seuillage sur les valeurs d'entropie. Cependant, il reste intéressant d'étudier l'évolution du rapport signal à bruit en fonction du nombre d'itérations. La figure 2.28 représente pour cela le rapport de l'énergie du signal sur l'énergie du résidu exprimée en dB en fonctions du nombre d'itérations sur un son de glockenspiel. On voit apparaître après quelques itérations une stabilisation de l'énergie du résidu, à

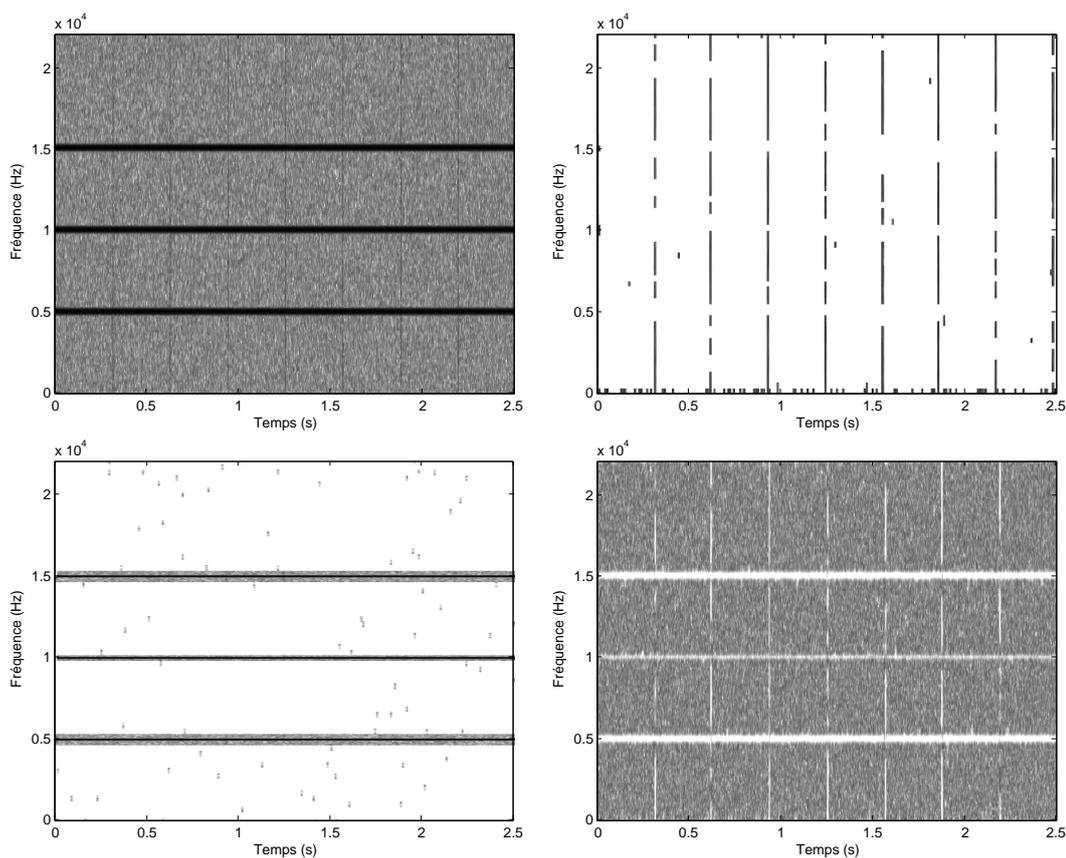


FIG. 2.26 – Décomposition obtenue avec l’algorithme de puzzle temps-fréquence (variante 1) avec seuillage des valeurs d’entropie : sur un signal constitué d’une somme de sinus, d’impulsions et d’un bruit blanc, en haut à gauche, transformée de Gabor du signal original, en haut à droite, représentation de la couche de la décomposition pour la petite fenêtre, en bas à gauche, représentation de la couche de la décomposition pour la grande fenêtre, en bas à droite, transformée de Gabor du résidu

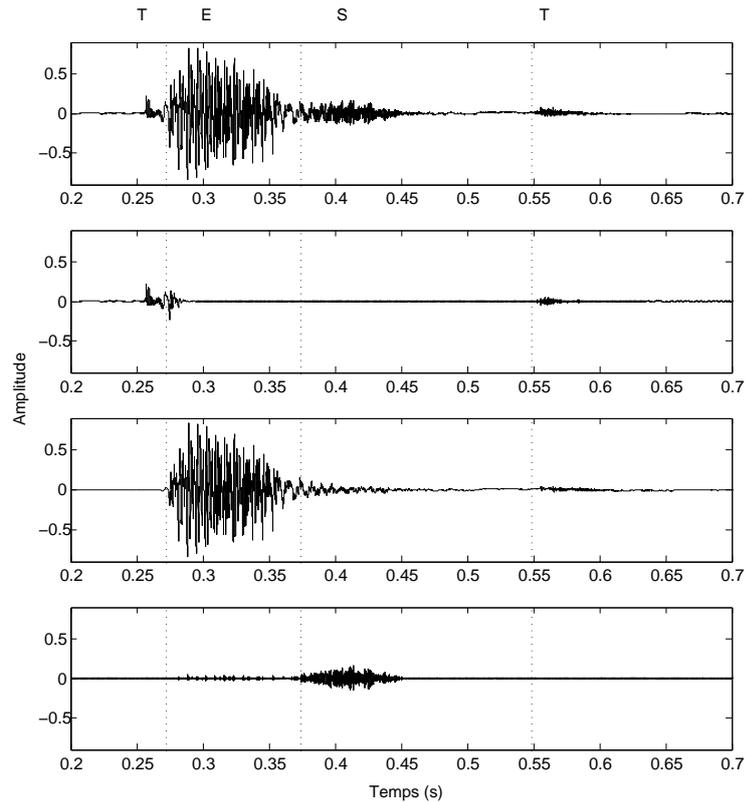


FIG. 2.27 – Application de l’algorithme de puzzle temps-fréquence (variante 1) avec seuillage des valeurs d’entropie sur un signal de parole : en haut, forme d’onde du signal original (le mot “test”), en dessous, forme d’onde de la couche de la décomposition pour la petite fenêtre, en dessous, forme d’onde de la couche de la décomposition pour la grande fenêtre, en bas, forme d’onde du résidu

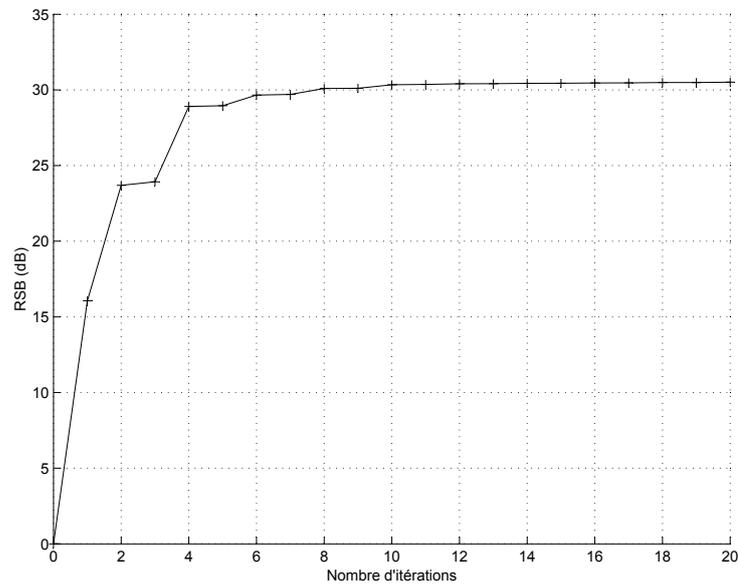


FIG. 2.28 – Évolution du rapport signal à bruit en fonction du nombre d'itérations pour l'algorithme de puzzle temps-fréquence (variante 1) avec seuillage

un niveau qui dépend de la valeur de seuil choisie par l'utilisateur. On retrouve ce comportement sur l'ensemble des sons testés.

Ce comportement pourrait laisser penser qu'après un certain nombre d'itérations, plus aucune valeur d'entropie ne dépasse le seuil et donc que plus aucun rectangle temps-fréquence n'est sélectionné dans les deux couches de la décomposition. En considérant le nombre de rectangles sélectionnés pour chaque couche en fonction du numéro d'itération, on s'aperçoit que ce phénomène ne se produit pas en réalité : le nombre de rectangles sélectionnés converge lui aussi. Il semble ainsi que l'algorithme converge vers des états quasi-stables où il effectue les mêmes sélections à chaque itérations sans qu'elles n'entraînent de modification notable de l'énergie du résidu. Les tests que nous avons menés pour étudier ce comportement n'ont pas permis de dégager une explication simple de ce phénomène.

Chapitre 3

Détection et estimation de composantes

Après avoir étudié comment assurer une bonne représentation des signaux dans le plan temps-fréquence au chapitre précédent, nous allons maintenant étudier comment améliorer l'accès à l'information perceptive présente dans ces représentations, de manière à permettre à l'utilisateur de réaliser facilement des traitements plus puissants. Pour cela nous allons proposer des méthodes de détection de composantes structurées sur ces représentations. Deux types de composantes sont abordés : les composantes localement harmoniques et les composantes transitoires.

3.1 Composantes localement harmoniques

3.1.1 Définition et caractérisation

Définition

On s'intéresse aux composantes localement harmoniques de la forme :

$$x(t) = \sum_{k=k_1}^{k_2} A_k(t) \cos \left(2\pi \int_0^t k f_0(u) du + \phi_k \right) \quad (3.1)$$

où f_0 est la fréquence fondamentale (variable dans le temps) de la composante, A_k est l'amplitude (variable dans le temps) de l'harmonique k , ϕ_k est la phase à l'origine de l'harmonique k , k_1 est le rang du premier harmonique et k_2 le rang du dernier harmonique.

Nous utilisons cette dénomination pour ces composantes car notre analyse est basée sur l'heuristique¹ suivante : lorsque les fonctions du temps f_0 et A_k pour

¹Pour donner un sens plus précis à cette heuristique, il faut se doter d'outils mathématiques

$k = \{1, \dots, K\}$ varie suffisamment lentement par rapport à la valeur de la fréquence fondamentale f_0 , on peut, autour d'un instant t , considérer localement ces fonctions comme des constantes et donc considérer localement la composante comme harmonique.

Remarque 3.1 *L'étude de ce type de composantes est particulièrement intéressante pour les applications de design sonore industriel car les composantes localement harmoniques apparaissent dans de nombreux domaines industriels. En effet, lorsqu'un système physique linéaire est excité de manière périodique, il crée ce type de composantes. C'est en particulier le cas des machines tournantes que l'on retrouve dans de nombreuses applications industrielles. Dans les machines tournantes, la fréquence fondamentale de la composante localement harmonique émise lors du fonctionnement évolue au cours du temps en fonction de l'évolution du régime de rotation du moteur.*

Dans le cadre de notre application au design sonore, notons que d'un point de vue perceptif, une composante localement harmonique est perçue comme un élément unique. Il est en conséquence souhaitable de permettre à l'utilisateur de traiter cette composante comme un seul objet. Il faut pour cela mettre à sa disposition des outils permettant de détecter ces composantes et de les traiter de manière globale.

Pour mettre en place une méthode de détection de ces composantes, on doit tout d'abord étudier comment elles apparaissent sur les représentations temps-fréquence, plus précisément sur une transformée de Gabor. De là, on pourra ensuite proposer une méthode de détection dans le plan temps-fréquence.

Caractérisation temps-fréquence

En utilisant l'approximation présentée dans l'exemple 2.4 à la section 2.1.3, lorsque les fonctions $A_k(t)$ pour $k \in \{k_1, \dots, k_2\}$ et $f_0(t)$ varient suffisamment lentement avec le temps, on a l'approximation suivante pour la transformée de Gabor de la composante localement harmonique x définie précédemment :

$$G_x^g \approx \frac{1}{2} \sum_{k=k_1}^{k_2} A_k(\tau) e^{i(2\pi \int_0^\tau k f_0(u) du + \phi_k)} \overline{\hat{g}(f_0(\tau) - \nu)}. \quad (3.2)$$

Dans ces conditions, et avec une fenêtre appropriée, l'énergie de la composantes apparaît sur le spectrogramme localisée autour de crêtes (*ridges* en anglais) qui suivent la fréquence instantanée de chacun des partiels, et se trouvent donc à chaque instant à des fréquences en rapport harmonique. La détection des composantes localement

permettant de quantifier ce que l'on entend par "lentement variable" et "localement". Ce sera là le rôle de la fenêtre que nous utiliserons dans le cadre de la transformée de Gabor.

harmoniques se fera en conséquence en recherchant sur le spectrogramme des crêtes harmoniques de ce type. La figure 3.1 montre le spectrogramme obtenu pour un exemple de composante localement harmonique composée de quatre harmoniques avec une fréquence fondamentale variant sinusoidalement autour d'une fréquence de 4000 Hz et avec une amplitude A_k gaussienne identique pour tous les partiels.

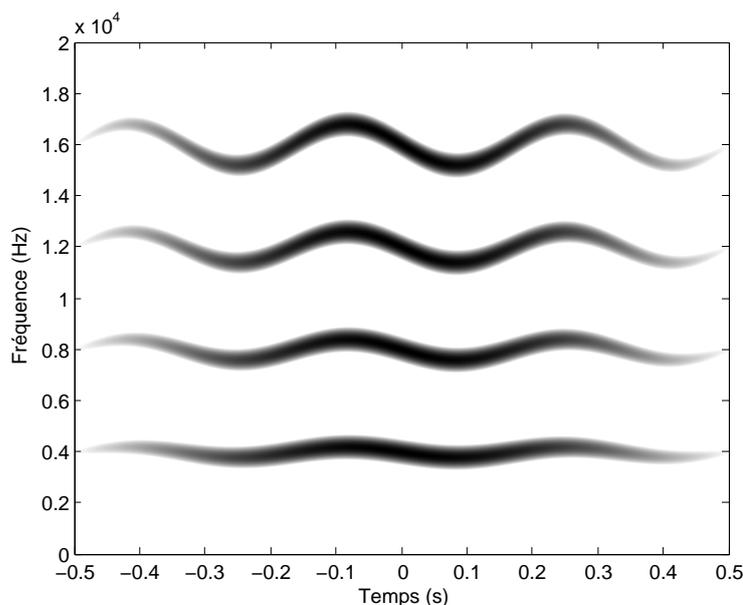


FIG. 3.1 – Exemple de spectrogramme pour une composante localement harmonique

Remarque 3.2 *Le choix de la fenêtre utilisée joue un rôle essentiel dans la validité de l'approximation (3.2). En effet, la notion de variation lente de la fréquence fondamentale f_0 et des amplitudes A_k est définie par rapport à la fenêtre : sur la durée de la fenêtre, la variation de f_0 et des A_k doit être suffisamment faible pour que ces fonctions puissent être considérées comme constantes. On est donc amené à choisir une fenêtre suffisamment courte pour que cette condition soit respectée. Cependant, pour que le spectrogramme obtenu fasse effectivement apparaître une crête pour chaque harmonique, il faut que les harmoniques soient bien résolus. Pour cela, on est amené à choisir une fenêtre suffisamment grande par rapport à la fréquence fondamentale qui détermine la proximité des harmoniques dans le plan temps-fréquence. Si la fenêtre est mal adaptée, les termes d'interférences entre les harmoniques deviennent importants et le spectrogramme ne fait plus simplement apparaître une crête pour chaque harmonique, comme on l'a vu sur l'exemple 2.5 de la section 2.1.3.*

On a donc généralement un compromis à trouver entre une taille de fenêtre suffisamment grande pour assurer la bonne résolution des harmoniques et suffisamment

courte pour que la variation des fonctions f_0 et A_k reste faible. Dans certains cas de figure, lorsque ces fonctions varient trop rapidement par rapport à la valeur de la fréquence fondamentale, il n'est pas possible de trouver de taille de fenêtre satisfaisante. On peut limiter le nombre de cas de ce type en utilisant la transformée de Gabor instationnaire présentée en 2.2.6 qui permet d'adapter la taille de la fenêtre au cours du temps.

3.1.2 Approches courantes pour la détection

Les approches décrites dans la littérature utilisent généralement la caractérisation temps-fréquence sur la transformée de Gabor que nous venons de décrire pour détecter les composantes² localement harmoniques. Les méthodes proposées consistent donc à rechercher des lignes de crêtes sur le spectrogramme du signal.

Nous allons donner ici très rapidement la structure généralement retenue pour réaliser cette recherche. Pour une présentation plus détaillée des méthodes de modélisation sinusoïdale, on pourra consulter [40]. On pourra également trouver une comparaison des résultats obtenus avec diverses méthodes dans [47].

La recherche des crêtes dans le plan temps-fréquence se fait généralement en deux étapes, en découplant la recherche selon la dimension fréquentielle et selon la dimension temporelle :

Détection des pics spectraux : la première étape consiste à rechercher pour chaque instant de calcul de la transformée les maxima locaux “significatifs” du spectrogramme,

Détermination des trajets des partiels : la deuxième étape consiste à lier temporellement les maxima locaux détectés précédemment pour retrouver les lignes de crêtes correspondant aux différents harmoniques (ou partiels dans le cas où l'on utilise un modèle plus général de sinusoïdes sans rapport spécifique de fréquence).

Dans chacune des deux étapes, quelques problèmes-clefs se posent.

Concernant la détection des pics spectraux, la principale difficulté est d'identifier, pour un instant donné, les maxima locaux de la transformée correspondant effectivement à une composante sinusoïdale. En effet, beaucoup de maxima locaux peuvent apparaître sur la transformée, en particulier lorsque le signal est bruité, mais également à cause des lobes secondaires de la fenêtre d'analyse. Différentes méthodes sont proposées pour faire face à ce problème, par exemple en jouant sur

²Il faut noter que les auteurs ne se limitent pas toujours au cas des composantes localement harmoniques, mais utilisent souvent un modèle plus général de somme de sinusoïdes d'amplitudes et de fréquences variant lentement dans le temps sans rapport spécifique entre les fréquences des sinusoïdes. Les composantes localement harmoniques constituent un cas particulier de ce modèle.

la taille du voisinage utilisé pour définir la “localité” du maximum, ou en corrélant le spectre avec la réponse en fréquence de la fenêtre pour identifier les pics ayant une forme appropriée.

Concernant la détermination des trajets de partiels, la difficulté est d’éviter les erreurs dans le suivi d’une composante sinusoïdale, en sélectionnant un maximum spectral correspondant à une autre composante, ou un maximum ne correspondant pas à une composante sinusoïdale. Pour éviter ces erreurs, on introduit dans les processus de décision des connaissances a priori sur les composantes, en particulier le fait qu’elle varient lentement dans le temps. L’introduction de ces a priori se fait de manière plus ou moins formelle selon les auteurs, allant d’une simple limitation de la zone de recherche à l’utilisation de modèle probabilistes plus poussés.

Il est intéressant de noter que la recherche de crêtes se fait rarement par une optimisation temps-fréquence directe sur la transformée : généralement, les dimensions fréquentielle et temporelle sont traitées séquentiellement, comme on vient de l’indiquer, et même dans des cas où la recherche des trajets de partiels se fait de manière globale, cette recherche ne se fait pas directement sur la transformée, mais sur les pics détectés lors de la première étape [11]. On peut cependant citer l’approche décrite dans [10] qui propose une modélisation sinusoïdale probabiliste du signal, les paramètres du modèles étant estimés de manière globale sur la transformée.

3.1.3 Critère pour la détection de composantes localement harmoniques

Comme nous nous intéressons à la détection de composantes localement harmoniques dans des cas qui peuvent être très bruités, compte tenu de notre application au design sonore industriel, nous souhaitons mettre en place une approche qui soit robuste au bruit. Pour cela, l’approche à deux étapes (détection de pics, recherche de trajets de partiels) généralement utilisée ne semble pas bien adaptée. En particulier, en présence de fort bruit, la distinction des maxima locaux correspondant à des sinusoïdes des maxima dus au bruit est particulièrement difficile. De plus, les maxima correspondant à une sinusoïde peuvent localement être déplacés voir disparaître en raison du bruit. Ainsi l’effet du bruit peut être localement très important, mettant en défaut une approche de détection trop locale.

Pour être plus robuste, nous allons utiliser une approche de détection plus globale en introduisant les a priori que nous avons sur les composantes cherchées (composantes localement harmoniques sans inharmonicité variant lentement dans le temps, et contenant des rangs d’harmoniques supposés connus) et en réalisant une optimisation globale dans une région temps-fréquence.

Cependant, nous souhaitons mettre en place une méthode de détection qui reste bien adapté à notre cadre d’étude. En particulier, nous souhaitons que son coût de

calcul reste compatible avec le traitement de son de plusieurs minutes, et également, qu'elle ne requiert pas le réglage d'un trop grand nombre de paramètres, pour faciliter son utilisation. Pour ces raisons, nous n'avons pas retenu d'approche bayésienne du type de celle proposée dans [10] .

Notre méthode de détection se base sur l'optimisation d'un critère de détection et utilise la stratégie proposée dans [3], en l'adaptant à notre cadre d'étude, c'est à dire au cas où l'on a plus d'un harmonique.

Compte tenu de leur caractérisation temps-fréquence, pour détecter les composantes localement harmoniques, nous cherchons dans la représentation des peignes harmoniques de crêtes. Deux arguments sont utilisés dans cette démarche pour construire un critère de détection :

- Pour détecter les crêtes, on recherche la fréquence fondamentale pour laquelle l'énergie sur la transformée de Gabor est maximale sous le peigne harmonique correspondant. Pour cela on définit pour une fréquence fondamentale f fonction du temps le critère suivant :

$$E(f) = - \int M(\tau, f(\tau)) d\tau \quad (3.3)$$

où

$$M(\tau, \nu) = \sum_{k=k_1}^{k_2} |G_x^g(\tau, \nu)|^2 \quad (3.4)$$

représente l'énergie apparaissant sur la représentation à l'instant τ sous un peigne harmonique de fréquence fondamentale ν . La fonction f minimisant le critère E correspond à celle pour laquelle l'énergie est maximale et coïncide donc avec la fréquence fondamentale f_0 lorsque l'on analyse une composante localement harmonique. Nous utilisons ici le fait que la composante est supposée strictement harmonique avec des rangs d'harmoniques connus.

Dans les applications pratiques, le signal peut être plus complexe et contenir d'autres composantes dont l'énergie va fausser le critère E et biaiser l'estimation de la fréquence fondamentale. En particulier, lorsque le signal est fortement bruité, des maxima locaux d'énergie dus au bruit peuvent devenir localement prépondérants par rapport à l'énergie de la composante harmonique, ce qui va se traduire par un saut dans la valeur de la fréquence fondamentale optimale obtenue en utilisant le critère E . Pour tenir compte de ce phénomène, on introduit un deuxième terme dans le critère, dont le rôle est d'assurer une certaine régularité à la fréquence fondamentale et éviter les sauts brusques.

- Dans une composante localement harmonique, la fréquence fondamentale varie par hypothèse lentement au court du temps. On peut en conséquence introduire dans le critère un terme de pénalisation sur la régularité de la fonction f afin de favoriser les variations lentes de la fréquence fondamentale.

Pour cela, on introduit un terme de régularisation basé sur la dérivée f' de la fonction f donnant la fréquence fondamentale, pour obtenir le critère :

$$\mathcal{C}(f) = - \int M(\tau, f(\tau)) d\tau + c \int |f'(\tau)|^2 d\tau, \quad (3.5)$$

où c est une constante positive permettant de contrôler l'importance relative des deux termes du critère, une plus grande valeur de c favorisant une plus grande régularité à la solution. La détection est donc finalement réalisée en cherchant la fonction f minimisant le critère \mathcal{C} .

Remarque 3.3 (Réglage du paramètre c) *Le réglage a priori du paramètre c ne paraît pas possible. En effet, si on fait l'hypothèse que le signal y sur lequel on effectue la détection est constitué d'une composante localement harmonique x et d'un bruit blanc gaussien additif b , l'expression du critère devient :*

$$\mathcal{C}(f) = - \int \sum_{k=k_1}^{k_2} \left(|G_x^g(\tau, k f(\tau))|^2 + \mathbb{E} \left(|G_b^g(\tau, k f(\tau))|^2 \right) \right) d\tau + c \int |f'(\tau)|^2 d\tau. \quad (3.6)$$

On souhaite que le terme de régularité $c \int |f'(\tau)|^2 d\tau$ puisse compenser l'effet du bruit, il doit donc être du même ordre le terme $\int \sum_{k=k_1}^{k_2} \left(\mathbb{E} |G_b^g(\tau, k f(\tau))|^2 \right) d\tau$. La valeur de c devrait donc idéalement dépendre de la variance du bruit b et de la borne supérieure de $|f'_0|$, qui sont inconnus.

La valeur de c doit en conséquence être réglée par l'utilisateur en fonction du signal, de manière à assurer un bon compromis permettant de limiter le biais d'estimation dû au terme de régularisation tout en compensant l'influence du bruit et assurer une bonne détection (c permet en fait de régler le compromis biais/variance de l'estimateur).

Pour faciliter le réglage du paramètre c , en constatant qu'il paraît naturel que le résultat de la détection soit indépendant du facteur d'amplitude du signal, l'utilisateur règle en fait une constante c_1 tel que $c = \frac{c_1}{p_x}$, où p_x est la puissance moyenne du signal (qui est bien définie car on réalise dans la pratique la détection sur une durée finie).

3.1.4 Discrétisation du critère

Pour une utilisation pratique, l'optimisation est faite numériquement. Pour cela on utilise une version discrétisée \mathcal{C}_d du critère \mathcal{C} avec un pas de discrétisation temporel Δ_t et un pas fréquentiel Δ_f . On remplace les intégrales par des sommes et on approxime la dérivée de f en t par $\frac{f(t+\Delta_t)-f(t)}{\Delta_t}$. Le critère à minimiser est donc de

la forme :

$$\mathcal{C}_d(f) = -c_2 \sum_p M(p\Delta_t, f(p\Delta_t)) d\tau + c_3 \sum_p |f((p+1)\Delta_t) - f(p\Delta_t)|^2, \quad (3.7)$$

avec c_2 et c_3 des constantes positives.

De plus, la détection se fait sur une durée limitée, l'indice temporel varie donc en fait entre 0 et un entier $P - 1$. Enfin, on limite également entre deux bornes f_{min} et f_{max} la zone de recherche de fréquence fondamentale en imposant à f la condition : $\forall t \in \mathbb{R}, 0 < f_{min} \leq f \leq f_{max}$. La limitation inférieure est motivée par la limitation de résolution fréquentielle de la transformée (qui dépend de la fenêtre d'analyse), la limitation supérieure est liée à l'hypothèse de bande limitée des signaux discrets.

Plus de détails sur la discrétisation du critère sont donnés dans [25].

Cependant, même si on limite ainsi la zone de recherche, cette zone peut être de très grande dimension, puisqu'on souhaite traiter des sons de longue durée avec des fréquences d'échantillonnage élevées. Il n'est en conséquence pas possible de réaliser l'optimisation numérique directement, car elle serait beaucoup trop coûteuse en temps de calcul.

Nous allons proposer deux approches alternatives permettant de réaliser cette optimisation de façon compatible avec une application réaliste :

- La première est une approche par optimisation stochastique basée sur l'algorithme de recuit simulé. Nous verrons quelles sont les difficultés et les limitations rencontrées avec cette méthode.
- Nous verrons ensuite que le problème d'optimisation peut être résolu de manière efficace et déterministe en utilisant des méthodes de programmation dynamique. Cette approche permet une solution rapide tout à fait compatible avec les applications sans présenter les difficultés de l'optimisation stochastique.

3.1.5 Résolution par optimisation stochastique

Une première approche possible est d'utiliser des algorithmes stochastiques d'optimisation, comme cela est proposé dans [3]. L'idée de base de ces méthodes est de modifier les méthodes de gradient en laissant la possibilité de déplacements dans l'espace des paramètres dans des directions autres que la direction de plus fort gradient. Ceci permet, sous certaines conditions, d'échapper aux minima locaux. De tels déplacements sont permis avec une certaine probabilité qui décroît au fur et à mesure que l'on avance dans les itérations ce qui assure la convergence de la méthode. On s'intéresse en particulier à l'algorithme de recuit simulé³ qui, appliqué à notre

³Pour une présentation générale de l'algorithme de recuit simulé et du cadre de son étude, on peut se référer à [45].

problème d'optimisation, peut être formulé sous la forme de l'algorithme 1 et dont nous allons maintenant décrire le fonctionnement.

Il consiste à choisir une valeur de fréquence fondamentale initiale f^0 , puis à tirer au hasard un instant p et à modifier la position de la fréquence fondamentale à cet instant, en l'augmentant ou en la diminuant d'un pas fréquentiel selon un choix fait au hasard (choix de ϵ). Si la nouvelle position de fréquence fondamentale h permet d'obtenir une meilleure valeur du critère, on conserve la nouvelle position comme référence. Si la nouvelle valeur du critère est moins bonne, on peut tout de même conserver la nouvelle position de fréquence fondamentale comme référence, avec une probabilité donnée, ce qui permet d'éviter de rester bloquer par un minimum local du critère. Le processus de déplacement est ensuite itéré jusqu'à ce que la position de la fréquence fondamentale soit inchangée depuis un nombre N d'itérations fixé par l'utilisateur.

La convergence de l'algorithme est assurée par le fait que la probabilité de conserver une position de fréquence fondamentale pour laquelle le critère est moins bon tend vers zéro au fur et à mesure du déroulement de l'algorithme. Cette probabilité est fixée à $\exp\left(\frac{C(f^{n-1})-C(h)}{T_n}\right)$ où T_n est un paramètre, appelé température, qui varie avec le numéro d'itération n . Cette valeur de température est donnée par la relation $T_n = \frac{T_0}{\log_2(n+2)}$, où T_0 est une température initiale fixée par l'utilisateur. Cette relation de mise à jour permet théoriquement d'assurer une convergence vers le minimum global du critère pour une valeur initiale de température T_0 suffisamment élevée.

Influence de la température initiale

L'utilisation pratique de l'algorithme pose certaines difficultés. La principale est liée au choix de la valeur initiale de température T_0 . En effet, dans l'algorithme 1, lorsque la température T_n tend vers 0, la probabilité d'aller vers une position pour laquelle le critère est moins bon tend également vers 0, ce qui signifie que si la température devient trop faible, l'algorithme risque de ne pas parvenir à dépasser un minimum local qu'il devrait passer pour atteindre la position optimale correspondant au minimum global du critère. Il faut en conséquence que la température initiale soit suffisamment grande pour qu'en début d'algorithme la probabilité de déplacement "en sens inverse" du critère soit suffisamment élevée pour sortir des bassins correspondant à des minima locaux et permettre au système de se figer en convergeant vers la bonne solution. Le choix de la valeur de T_0 doit donc dépendre du niveau et du nombre des minima locaux à passer depuis la position initiale. Comme dans le cas général, ces minima locaux sont inconnus le choix a priori de T_0 est très délicat.

Exemple 3.1 *La figure 3.2 montre les résultats obtenus sur un son de synthèse avec l'algorithme de recuit simulé pour la même sélection initiale de fréquence fondamen-*

[Initialisation du numéro d'itération n]

$n \leftarrow 1$

Répéter

| [Choix d'un instant p]

| $p \leftarrow \text{HASARD}(\{0, \dots, P-1\})$

| [Choix d'un déplacement ϵ]

| $\epsilon \leftarrow \text{HASARD}(\{-1, 1\})$

| [Définition de la position de fréquence fondamentale candidate h obtenue à partir de la position précédente en déplaçant le point à l'instant p de ϵ]

| $h \leftarrow f^{n-1}$

| $h_p \leftarrow f_p^{n-1} + \epsilon$

| **Si** ($C(h) < C(f^{n-1})$) **Alors**

| | [Le critère est meilleur pour la nouvelle position, on la conserve]

| | $f^n \leftarrow h$

| **Sinon**

| | [Le critère est moins bon pour la nouvelle position, on peut cependant la conserver avec la probabilité $\exp\left(\frac{C(f^{n-1})-C(h)}{T_n}\right)$]

| | $T_n \leftarrow \frac{T_0}{\log_2(n+2)}$

| | **Si** ($\text{HASARD}([0, 1]) < \exp\left(\frac{C(f^{n-1})-C(h)}{T_n}\right)$) **Alors**

| | | $f^n \leftarrow h$

| | **Sinon**

| | | $f^n \leftarrow f^{n-1}$

| | **Fin Si**

| **Fin Si**

| [Mise à jour du numéro d'itération]

| $n \leftarrow n + 1$

jusqu'à ce que ($f^n = f^{n-1} = \dots = f^{n-(N-1)}$)

ou $\text{HASARD}(E)$ renvoi un élément de l'ensemble E tiré au hasard uniformément parmi les éléments de E

Algorithme 1: Algorithme de recuit simulé appliqué au problème d'optimisation du critère C

tale, mais avec deux valeurs différentes de T_0 . Le son est constitué d'une somme d'un bruit blanc et d'une composante localement harmonique dont la fréquence fondamentale varie sinusoidalement et contenant trois harmoniques. On voit qu'avec une valeur élevée de T_0 , le résultat est satisfaisant, alors qu'avec une valeur plus faible, l'algorithme est bloqué par un minimum local dû au bruit, ce qui se traduit par une mauvaise détection entre les instants 0,2 s et 0,4 s.

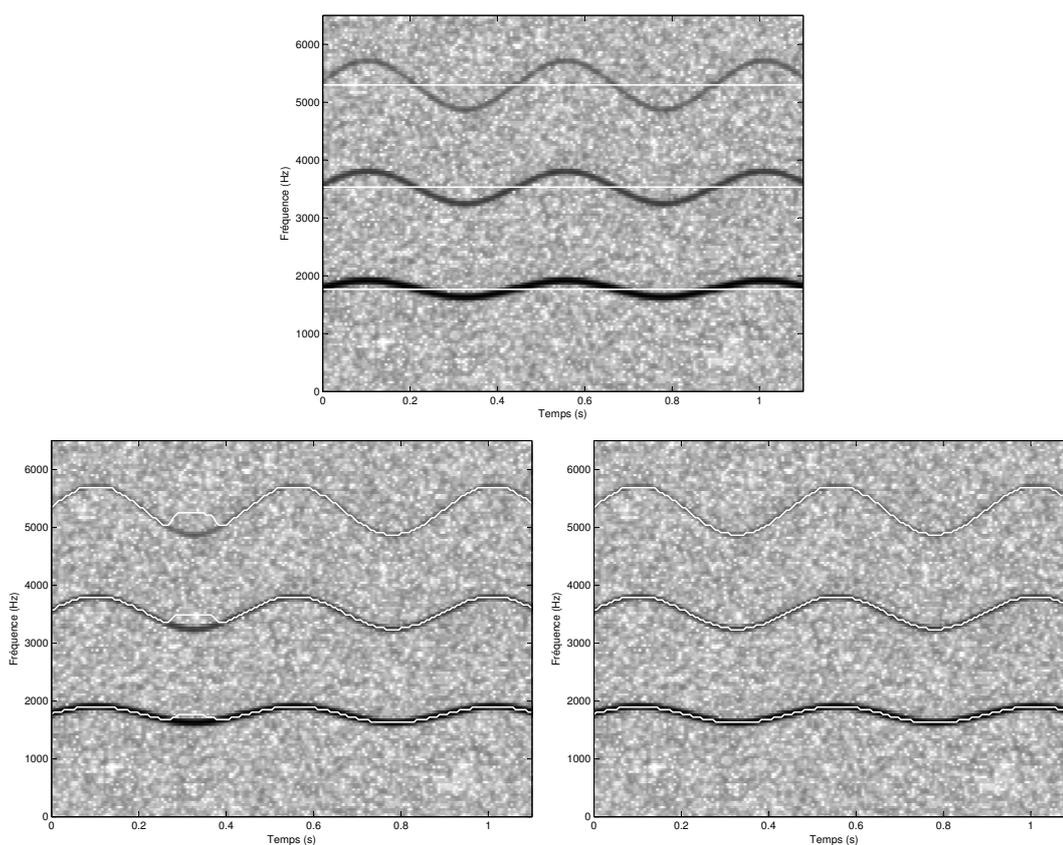


FIG. 3.2 – Exemple de détection de composante localement harmonique avec l'algorithme de recuit simulé : en haut, les lignes blanches représentent la position des harmoniques à l'initialisation de l'algorithme, en bas les résultats de la détection obtenu pour deux valeurs de température initiale, à gauche une valeur faible, à droite une valeur plus élevée

Remarque 3.4 *Par rapport au cas où un seul harmonique est présent, comme dans [3], l'optimisation du critère est plus complexe lorsque plusieurs harmoniques sont présents. En effet, le critère présente alors des minima locaux importants du fait de*

la structure du problème étudié. En plus des minima locaux dus au bruit, et dont les niveaux dépendent du niveau relatif du bruit, le terme énergétique du critère calculé sur un peigne harmonique présente des minima locaux pour les fréquence multiple et sous-multiple de la fréquence fondamentale. En considérant par exemple un cas où l'on recherche une composante harmonique contenant 6 harmoniques avec une fréquence fondamentale f_{fond} , l'énergie mesurée sous un peigne de 6 harmoniques avec la fréquence fondamentale $\frac{f_{\text{fond}}}{2}$ est importante puisque l'on récupère l'énergie des trois premiers harmoniques. Si de plus le niveau des harmoniques est décroissant avec leur rang, comme c'est généralement le cas dans des signaux réels, la différence de valeur de la partie énergétique du critère entre une fréquence fondamentale de f_{fond} et une fréquence fondamentale de $\frac{f_{\text{fond}}}{2}$, qui correspond à l'énergie des harmoniques 4 à 6, peut être faible. On peut donc avoir un minimum local très marqué et de niveau proche du minimum global pour la fréquence fondamentale $\frac{f_{\text{fond}}}{2}$. Si la position initiale f^0 est plus proche de $\frac{f_{\text{fond}}}{2}$ que de f_{fond} , il faut donc que la température soit très élevée pour permettre la convergence de l'algorithme vers la bonne solution.

Cependant, comme l'augmentation de la température initiale implique une convergence plus lente, et donc un temps de calcul plus élevé, on souhaite choisir une température initiale aussi faible que possible. Concrètement, dans certains cas de figure, pour assurer la convergence vers le minimum global, on est amené à utiliser des valeur de T_0 pour lesquelles le temps de calcul devient trop élevé et incompatible avec une utilisation de la méthode dans un contexte de traitement pour le design sonore industriel.

Le risque de convergence vers un minimum local du critère pour une valeur inadaptée de T_0 implique une ambiguïté sur la validité de la solution trouvée : il n'est pas possible de s'assurer que la solution obtenue soit bien le minimum global du critère. Ceci pose en particulier des problèmes pour l'étude de l'influence du paramètre c permettant de régler l'importance de la régularisation dans le critère. Il devient en effet très difficile de distinguer l'effet des deux paramètres c et T_0 . Par exemple, dans le cas où une augmentation de c permet, pour une même valeur de T_0 , d'obtenir un résultat satisfaisant, on ne sait pas si le bon résultat est dû au fait que l'augmentation de c a effectivement déplacé le minimum global du critère à la position correspondant à la composante harmonique, en contrebalançant une perturbation due au bruit, ou si l'augmentation du c a permis de supprimer un minimum local vers lequel l'algorithme convergeait à tort en raison d'une valeur de T_0 trop petite, le minimum global du critère correspondant déjà à la bonne solution.

Influence de la condition d'arrêt

Un deuxième élément est déterminant en ce qui concerne la durée de convergence de l'algorithme. Il s'agit du nombre d'itérations sans déplacement N intervenant

dans la condition d'arrêt de l'algorithme. Ce nombre doit être suffisamment grand pour s'assurer que le système a effectivement atteint une position d'équilibre. Cependant, comme pour la température initiale, une valeur plus élevée de N implique une plus grande durée de convergence.

En effet, même lorsque la fréquence fondamentale se trouve à la position optimale, elle peut encore se déplacer puisqu'une position correspondant à une moins bonne valeur du critère peut être retenue avec la probabilité non-nulle $\exp\left(\frac{C(f^{n-1})-C(h)}{T_n}\right)$. Or, à proximité de la position optimale, la valeur du critère pour la position h qui ne diffère de f^{n-1} que d'un pas fréquentiel du fondamental en un instant est très proche de celle obtenue pour f^{n-1} . Plus précisément, le déplacement faible du fondamental autour de la position optimale correspond à un déplacement sur le sommet des lobes principaux apparaissant sur la transformée autour des crêtes. Le niveau d'énergie évolue lentement le long de ces lobes et la différence $C(f^{n-1}) - C(h)$ peut donc être très faible autour de la position optimale. La probabilité associée peut rester importante, même avec une faible valeur de T_n . Si N est trop grand, on risque ainsi d'avoir une valeur de fréquence fondamentale qui "vibre" lentement mais longtemps autour de la position optimale en attendant un grand nombre d'itérations inutiles avant d'avoir N itérations sans déplacement et de stopper l'algorithme. Pour limiter cet effet, on peut par exemple modifier la condition d'arrêt en vérifiant si les N dernières valeurs de fréquence fondamentale sont suffisamment proches plutôt que strictement égales.

En conclusion, l'algorithme de recuit simulé permet effectivement de résoudre le problème d'optimisation étudié, mais dans le cas général, on est amené à utiliser des valeurs élevées de T_0 et de N pour assurer la convergence vers le minimum global, ce qui implique des temps de calcul incompatibles avec une application pratique au traitement pour le design sonore industriel.

Recuit à température faible ou nulle

Cependant, si la position initiale f^0 de la fréquence fondamentale est suffisamment proche de la solution, il devient possible d'utiliser une valeur de T_0 faible et d'avoir une convergence rapide de l'algorithme vers la bonne solution. C'est finalement dans ce cas de figure, c'est à dire avec une faible température initiale, que l'algorithme de recuit simulé peut être utilisé de façon satisfaisante. Ce qui apparaissait comme un défaut pour résoudre le problème d'optimisation global peut même alors devenir une qualité. En effet, dans certains cas de figure, on utilise avec profit la capacité de l'algorithme à converger non pas vers le minimum global, mais vers un minimum local du critère. Par exemple, lorsque le signal est constitué de deux composantes localement harmoniques de fréquences fondamentales différentes avec une des deux composantes de plus forte énergie que l'autre, l'optimisation globale du

critère ne permet de détecter que la composante la plus énergétique. En revanche, avec l'algorithme de recuit simulé, si on initialise l'algorithme avec une fréquence fondamentale initiale suffisamment proche de la fréquence fondamentale de la composante de plus faible énergie, et si la température initiale est suffisamment faible, on peut converger vers le minimum local correspondant à la composantes de plus faible énergie. On a le même cas de figure lorsque l'on souhaite sélectionner uniquement les harmoniques paires d'une composantes localement harmoniques : l'optimisation globale ne permet de détecter que la fréquence fondamentale de la composante localement harmonique, alors qu'en initialisant l'algorithme à proximité du double de la fréquence fondamentale de la composante, on peut sélectionner avec l'algorithme uniquement les harmoniques paires.

Dans l'optique d'une utilisation du recuit simulé avec une faible température initiale, le cas limite où la température devient nulle peut avoir un intérêt particulier. La probabilité de conserver un déplacement pour lequel le critère est moins bon devient alors nulle, et seuls les déplacements vers une meilleure valeur du critère sont alors possible. L'algorithme a alors un comportement de type descente de gradient.

Pour étudier le cas limite du recuit simulé avec une température nulle, il est intéressant de mettre en place l'algorithme 2 dont le comportement est très proche, mais qui permet de réduire le temps de calcul. En effet, à chaque itération, le choix d'un instant et d'un déplacement ne se font plus au hasard, mais dans un ordre systématique. L'algorithme devient ainsi déterministe, ce qui permet de lever la difficulté du choix de N pour la convergence de l'algorithme, qui est maintenant détectée facilement lorsque la fréquence fondamentale obtenue pour un instant p est identique à celle obtenue la fois précédente où l'instant p a été considéré.

Cet algorithme permet une convergence rapide, par contre, il est très sensible aux minima locaux, ce qui peut être gênant pour des sons très bruités, pour lesquels il est nécessaire de sélectionner précisément la fréquence fondamentale initiale pour obtenir un résultat satisfaisant. Ce phénomène est illustré sur la figure 3.3 où l'on peut comparer les résultats obtenus avec l'algorithme 2 sur le même signal que dans la figure 3.2 pour deux sélections de fréquences fondamentales initiales différentes.

En conclusion, on observe que l'optimisation du critère est possible en utilisant l'algorithme de recuit simulé ou l'algorithme 2. Cependant, pour obtenir un résultat satisfaisant dans des conditions réalistes d'utilisation sur un signal bruité, on doit débiter l'optimisation à partir d'une position initiale proche (voir très proche) de la solution, ce qui est trop contraignant pour une utilisation pratique. On préférera en conséquence utiliser la deuxième approche, qui, grâce à une résolution par programmation dynamique, permet le calcul déterministe efficace du minimum global du critère.

```

[Initialisation du numéro d'itération n]
n ← 1
Répéter
  Pour p de 0 à P - 1 faire
    [Définition de deux positions de fréquence fondamentale candidates
    h1 et h2 obtenue à partir de la position précédente en déplaçant le
    point à l'instant p de +1 et -1]
    h1 ← fn-1
    hp1 ← fpn-1 + 1
    h-1 ← fn-1
    hp-1 ← fpn-1 - 1
    [Mise à jour de la position de fréquence fondamentale avec
    la meilleure position parmi les deux candidates et la position
    précédente]
    fn = arg ming ∈ {h1, h-1, fn-1} (C(g))
    [Mise à jour du numéro d'itération]
    n ← n + 1
  Fin Pour
jusqu'à ce que (fn = fn-P)

```

Algorithme 2: Algorithme déterministe similaire à un recuit simulé à température nulle appliqué au problème d'optimisation du critère C

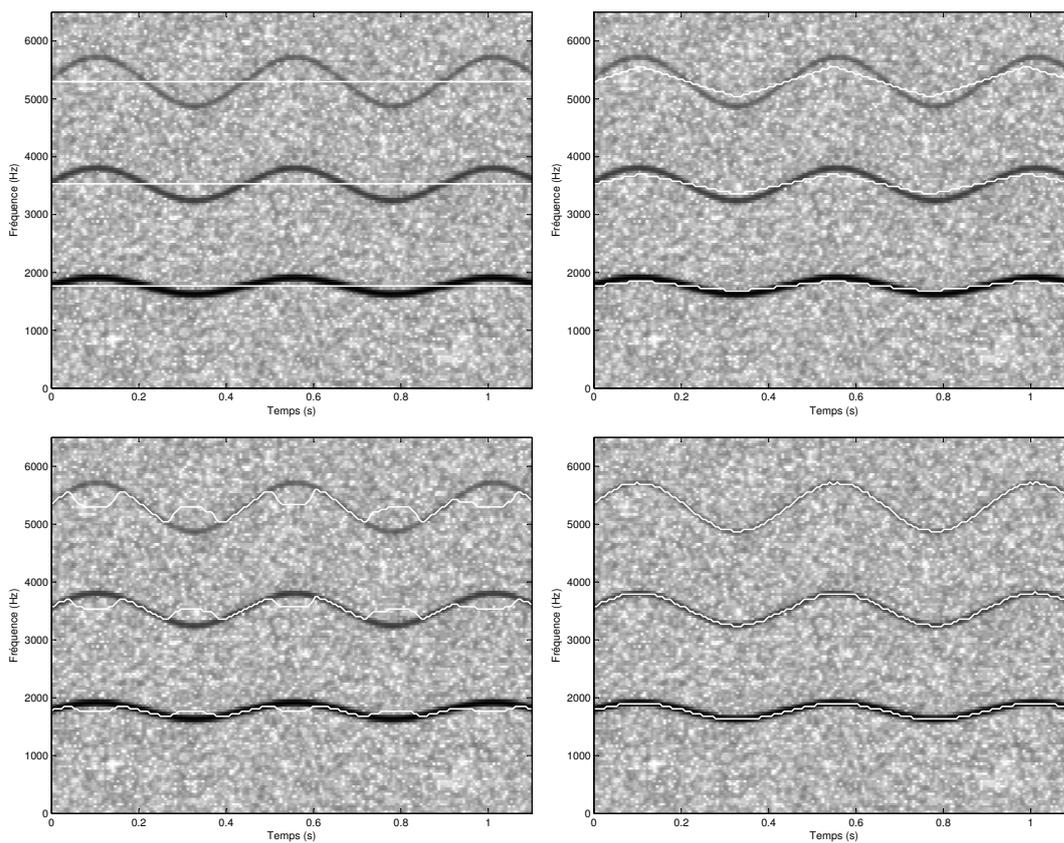


FIG. 3.3 – Exemple de détection de composante localement harmonique avec l'algorithme 2 : en haut, les lignes blanches représentent la position des harmoniques pour deux initialisations différentes de l'algorithme, en bas les résultats de détection correspondants

3.1.6 Résolution par programmation dynamique

Dans le cadre du réseau européen HASSIP (Harmonic Analysis and Statistics for Signal Processing) coordonné par Bruno Torrèsani, j'ai eu l'opportunité de visiter l'IBB (Institut für Biomathematik und Biometrie) du GSF (Forshungszentrum für Umwelt und Gesundheit, Neuherberg) également membre du réseau, durant une semaine. Cette visite était motivée par les difficultés décrites dans la section précédente, liées à l'utilisation de l'algorithme de recuit simulé. Le groupe de recherche en modélisation mathématique de l'IBB possède en effet une grande expérience dans l'utilisation de ce type d'algorithme, ainsi qu'une très forte compétence dans la théorie associée, particulièrement grâce à la présence de Gerhard Winkler, spécialiste du domaine [45], et de Felix Friedrich, spécialiste de l'implémentation de ce type de méthodes et développeur de la vaste bibliothèque dédiée AntsinField disponible à l'adresse <http://www.antsinfields.de>.

Lors de ma visite, Felix Friedrich, par analogie avec ce qui avait été fait auparavant dans l'équipe sur un autre problème d'optimisation habituellement résolu par optimisation stochastique⁴, a repéré que l'optimisation de la version discrétisée du critère pouvait être réalisée efficacement de manière déterministe.

Ceci est permis par la structure "causale" du critère dans sa formulation discrète grâce à laquelle il est possible de mettre en place une méthode de résolution par programmation dynamique, c'est à dire en résolvant une série de sous-problèmes d'optimisation du critère qui utilisent les résultats des sous-problèmes déjà résolus pour réduire la charge de calcul.

Pour retrouver le formalisme de programmation dynamique utilisé en théorie des graphes, on peut réécrire le problème d'optimisation du critère comme un problème de recherche de plus court chemin dans un graphe construit à partir des valeurs de M . Le problème d'optimisation peut alors être résolu en utilisant un algorithme de recherche de plus court chemin classique. Pour des raisons de confidentialité, j'ai décrit en détail la formulation correspondante et l'algorithme de résolution associé dans [25].

La méthode proposée permet une résolution déterministe du problème d'optimisation avec un temps de calcul très réduit, ce qui permet son utilisation sur des signaux longs.

Influence du paramètre de contrôle de régularité

Avec cet algorithme, l'influence du paramètre de contrôle de régularité c peut être évaluée sans ambiguïté, puisqu'on est assuré d'obtenir le minimum global du critère. On observe bien le comportement attendu : l'augmentation de la valeur de c favorise la sélection d'une fréquence fondamentale plus régulière, aboutissant dans

⁴Il s'agit du modèle de Potts, voir [46] pour plus de détails.

le cas extrême d'une très grande valeur de c au choix d'une fréquence fondamentale constante. Lorsque le signal contient d'autres composantes que la composante localement harmonique à détecter, et en particulier lorsque le signal est très bruité, on constate bien, comme on l'avait évoqué dans la remarque 3.3, que le réglage du paramètre c doit être bien adapté au signal pour obtenir un résultat satisfaisant. Il faut cependant noter que le choix de c ne devient critique que pour les forts niveaux de bruit. En effet, en dessous d'un certain niveau, les maxima locaux d'énergie dus au bruit ne dépassent pas le niveau des crêtes correspondant à la composante localement harmonique. En utilisant une faible valeur de c , la détection ne sera donc pas perturbée. L'influence de c est illustrée sur l'exemple suivant.

Exemple 3.2 *On peut voir sur la figure 3.4 les résultats de détection obtenus pour un signal bruité avec diverses valeurs du paramètre c . Le signal est constitué d'une composante localement harmonique dont la fréquence fondamentale varie sinusoidalement autour de 2000 Hz, à laquelle on ajoute un bruit blanc, avec un rapport signal à bruit de -13 dB. On voit qu'avec une trop faible valeur de c , la fréquence fondamentale présente des sauts brusques vers des maxima locaux d'énergie du bruit. En augmentant la valeur de c , ces sauts sont supprimés, et le résultat est satisfaisant. Pour une valeur trop élevée de c , le résultat obtenu est quasiment constant, l'influence du terme énergétique du critère devenant trop faible.*

Robustesse au bruit

La robustesse au bruit de la méthode de détection est bonne. Pour l'illustrer, nous avons repris le signal utilisé dans l'exemple 3.2 en diminuant le rapport signal à bruit, c'est à dire en augmentant le niveau du bruit, pour chercher jusqu'à quel niveau de bruit la détection restait possible. Sur cet exemple, la limite se situe autour d'un rapport signal à bruit de -20 dB. On peut voir sur la figure 3.5 que la composante localement harmonique n'est pas visible sur le spectrogramme dans ce cas (elle ne semble pas perceptible à l'écoute du signal non plus). L'algorithme de détection permet cependant de retrouver approximativement la forme de la composante localement harmonique (cette composante est la même que dans l'exemple 3.2, on souhaiterait donc idéalement retrouver le même résultat que sur la représentation du bas à gauche de la figure 3.4). Il faut noter que la qualité du résultat obtenu avec ce rapport signal à bruit est particulièrement sensible au réglage du paramètre c .

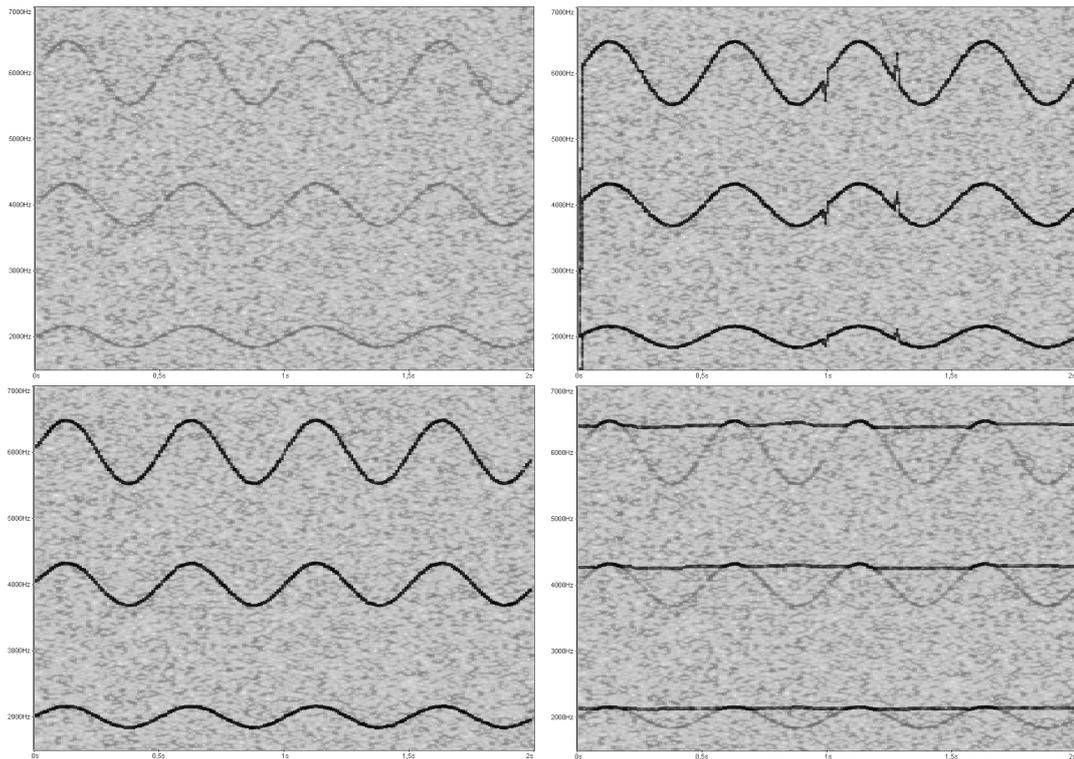


FIG. 3.4 – Influence de paramètre c sur la détection : en haut à gauche, spectrogramme du signal analysé, en haut à droite, détection pour une faible valeur de c , en bas à gauche, détection pour une valeur intermédiaire de c , en bas à droite, détection pour une grande valeur de c

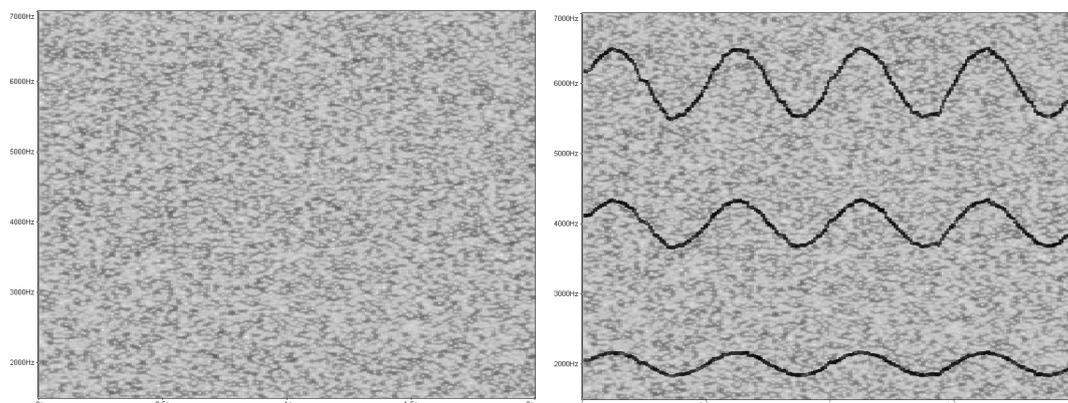


FIG. 3.5 – Exemple de détection en présence d'un fort bruit : à gauche, spectrogramme du signal analysé (rapport signal à bruit de -20 dB), à droite, résultat de la détection

3.1.7 Reconstruction de composantes localement harmoniques

Approches courantes pour la reconstruction

La détection de composantes localement harmoniques se fait généralement dans le but d'obtenir une représentation paramétrique du signal. Dans ce cadre, après la détection, la reconstruction des composantes localement harmoniques (ou plus généralement de sinusoides variant dans le temps) se fait généralement en estimant de manière explicite les paramètres du modèle associé aux composantes : on estime f , A_k , ϕ_k à partir de la transformée, puis on reconstruit directement les composantes à partir de la formule (3.1). Pour connaître les méthodes d'estimation des paramètres du modèle sinusoidal proposées dans la littérature, on pourra consulter [37].

Pour obtenir une bonne précision, l'estimation des paramètres du modèle n'utilise généralement pas uniquement le module de la transformée de Gabor, mais, également sa phase. Comme la transformée de Gabor est calculée avec un pas temporel plus grand que celui du signal, il est nécessaire d'interpoler les valeurs des paramètres estimés entre deux instants de la transformée.

Lorsque le nombre de sinusoides est important, la charge de calcul pour la reconstruction, qui se partage entre l'interpolation des paramètres estimés et la formule de reconstruction (3.1) elle-même, peut devenir importante. Des algorithmes rapides ont été proposés pour permettre une reconstruction plus rapide à partir des paramètres du modèle. Par exemple, l'algorithme FFT^{-1} de l'Ircam [41], utilise pour cela la formule d'inversion de la transformée de Gabor appliquée à une transformée approchée déterminée directement à partir des paramètres du modèle.

Reconstruction dans notre cadre de travail

Dans notre cadre général de traitement des sons par modification de leur représentation temps-fréquence, une manière simple de reconstruire les composantes localement harmoniques est d'utiliser les multiplicateurs de Gabor évoqués en 2.1.3. Pour cela, on fait l'hypothèse que l'énergie d'une composante sinusoïdale se localise sur sa transformée de Gabor dans une bande de largeur finie Δ_f autour de la fréquence instantanée de la sinusoïde et qu'elle est négligeable en dehors de cette bande⁵. On prendra typiquement comme largeur de bande la largeur du lobe principal de la fenêtre d'analyse. On reconstruit alors la composante localement harmonique par inversion de la transformée de Gabor du signal en ne conservant que les coefficients de la transformée se trouvant dans une bande de largeur Δ_f autour des crêtes détectées par l'algorithme.

Les difficultés apparaissant avec cette méthode sont les suivantes :

- La reconstruction des composantes n'est pas exacte, puisqu'on néglige une partie de l'énergie des composantes dans le plan temps-fréquence. Cependant, cette approximation peut être très bonne en utilisant des fenêtres appropriées, c'est à dire des fenêtres avec des lobes secondaires de très faible niveau.
- La variation de la fréquence instantanée d'une sinusoïde sur la durée d'une fenêtre d'analyse entraîne un élargissement du lobe principale en fréquence. Si la fréquence évolue suffisamment lentement, cet effet est négligeable, en revanche lorsque le variation devient rapide, il est souhaitable d'élargir la bande fréquentielle dans laquelle on conserve les coefficients pour la reconstruction, afin de conserver la même proportion de l'énergie de la composante indépendamment de la pente de sa fréquence instantanée. Ceci est particulièrement important dans le cas des composantes localement harmoniques, puisque la variation de fréquence instantanée n'est pas la même pour les différents rangs d'harmoniques (elle est proportionnelle au rang). Si on utilise la même largeur de bande pour reconstruire les différents harmoniques alors que la variation de fréquence instantanée est importante, on récupère une proportion plus faible de l'énergie des harmoniques de rangs plus élevés, ce qui modifie le niveau relatif des harmoniques et n'est pas satisfaisant dans le cadre de traitements audionumériques. Il faut donc adapter la largeur de bande en fonction de la pente de chaque harmonique pour éviter ce problème.
- En présence d'une autre composante, et en particulier de bruit, on ne reconstruit avec cette méthode pas uniquement la composante localement harmonique, mais on récupère également l'énergie de l'autre composante qui se

⁵Cette approximation est de même nature que celle faite dans l'algorithme de reconstruction FFT^{-1} [41]. On peut d'ailleurs remarquer que la méthode de reconstruction à l'aide des multiplicateurs de Gabor est très proche de l'algorithme FFT^{-1} , à la différence que les paramètres du modèle sinusoïdal ne sont pas estimés explicitement.

trouve dans la zone sélectionnée. On peut négliger ce phénomène lorsque le niveau relatif de l'autre composante est suffisamment faible. Par contre lorsque les deux composantes sont de niveaux comparables dans la zone sélectionnée, comme sur la figure 3.5, ou que le niveau de l'autre composante est plus élevé, un problème se pose. Aucune solution basée sur les simples multiplicateurs de Gabor ne semble alors possible. On peut cependant remarquer que dans ce type de cas, la composante que l'on souhaite reconstruire n'est pas audible dans la zone de chevauchement des composantes à cause des effets auditifs de masquage. On peut ainsi envisager dans certains cas de ne pas reconstruire ces portions de la composante localement harmonique.

Malgré ces limitations, la reconstruction à l'aide des multiplicateurs de Gabor présente un grand intérêt dans notre cadre de travail puisqu'elle s'intègre directement dans la boucle de traitement des sons par modification de leur représentation temps-fréquence en ne constituant alors qu'une méthode spécifique de sélection de régions temps-fréquence, parmi un ensemble de méthodes de sélection proposé à l'utilisateur. Cette méthode utilise pour la reconstruction l'inversion de la transformée de Gabor et ne demande pas d'estimation intermédiaire des paramètres du modèle. Elle permet donc une reconstruction avec un temps de calcul limité, compatible avec des signaux de grande durée.

De plus, l'utilisateur ne souhaite pas systématiquement extraire la composante localement harmonique. Il peut souvent vouloir simplement modifier le niveau de cette composante, ou le niveau relatif des harmoniques de cette composante, par exemple. L'utilisation des multiplicateurs de Gabor permet d'effectuer simplement ce type de traitements.

Une illustration sur un signal de montée en régime automobile des méthodes de détection et de reconstruction de composantes localement harmonique que nous venons de proposer est présenté en section 4.2, sur l'exemple 4.1.

3.2 Composantes transitoires

3.2.1 Caractérisation

Bien que couramment utilisé en traitement du signal audionumérique, la définition du terme "transitoire" reste imprécise et en particulier, il n'existe pas de modélisation mathématique simple des composantes transitoires comparable à celle donnée dans l'équation (3.1) pour les composantes localement harmoniques. Les composantes transitoires sont généralement associées aux attaques des notes en informatique musicale.

Pour décrire les composantes transitoires, on utilise généralement la caractérisation suivante : les composantes transitoires correspondent à des régions

de très courte durée dans lesquelles le signal évolue brusquement et rapidement, généralement de manière peu prévisible. Ces composantes apparaissent donc sur une représentation temps-fréquence comme des structures verticales occupant une large bande fréquentielle et bien localisées temporellement. Typiquement, ces composantes correspondent à la zone temporelle proche de l'impact dans un son de choc (avant une éventuelle zone de résonance).

La détection de ces composantes se fait généralement en recherchant directement les zones du signal correspondant à cette caractérisation. Dans des cas simples, la détection peut se faire directement à partir du signal en recherchant les zones d'augmentation brusque de l'énergie à court terme. Pour cela, on peut par exemple rechercher les pics de la dérivée de l'énergie à court terme du signal. Cependant lorsque les sons sont plus complexes, en particulier lorsque plusieurs composantes se superposent dans le temps, les transitoires peuvent ne pas apparaître directement sur la forme d'onde du signal. On est alors amené à utiliser la caractérisation temps-fréquence des transitoires, en réalisant la détection à partir d'une représentation temps-fréquence du signal.

D'autres approches sont possibles, on pourra se référer à [2] pour plus d'approches et plus de détails sur les approches évoquées ici.

Dans la suite, nous allons explorer rapidement les pistes qui peuvent être suivies pour utiliser les éléments présentés dans le cadre des puzzles temps-fréquence (voir 2.4) pour la détection de transitoires. Cette section ne prétend pas présenter une méthode complète de détection des transitoires. En particulier, aucune étude systématique des méthodes n'est réalisée. Pour évaluer ces méthodes et choisir les paramètres les mieux adaptés à la détection, il serait nécessaire de mettre en place une base de tests, ce qui est particulièrement long et difficile pour les composantes transitoires, en particulier en raison de l'absence de bonne définition des composantes (voir [30] pour les problèmes de méthodologie).

3.2.2 Détection dans le cadre du puzzle temps-fréquence

On a vu dans la partie 2.3.3, à travers l'interprétation multicouche de l'algorithme de puzzle temps-fréquence, que l'algorithme utilisé avec deux tailles de fenêtres a tendance à isoler les composantes transitoires dans la couche correspondant à la petite fenêtre d'analyse. Il paraît ainsi intéressant d'étudier comment les éléments mis en oeuvre dans cet algorithme, et en particulier les critères entropiques, peuvent être utilisés dans le cadre spécifique de la détection des transitoires.

Compte tenu de leur caractérisation, les composantes transitoires apparaissent de manière bien localisée sur une transformée de Gabor utilisant une fenêtre de courte durée. Sur une telle représentation, elles peuvent être caractérisées comme des composantes "verticales" (qui occupent une plage de fréquence étendue), le cas extrême de telles composantes étant l'impulsion, représentée comme une bande

verticale étroite couvrant toutes les fréquences pour une petite fenêtre d'analyse. En exploitant l'approche présentée dans la partie 2.3.1, on peut identifier les régions de la transformée de Gabor correspondant à ces caractéristiques, en recherchant des régions où l'algorithme de puzzle temps-fréquence a assez systématiquement préféré les fenêtres courtes.

Nous illustrons ce type d'approche sur le signal de glockenspiel déjà utilisé précédemment, qui n'est pas un exemple difficile, mais permet de bien montrer le comportement des différents critères que nous proposons.

Pour détecter les transitoires, on peut déjà remarquer sur la courbe du milieu de la figure 3.6 qu'un simple comptage au cours du temps du nombre de rectangles pour lesquels la petite fenêtre a été choisie dans l'algorithme de puzzle temps-fréquence variante 1 sans seuillage lors de la première itération fait apparaître des pics aux instants correspondant aux transitoires. La bonne localisation temporelle de ces composantes implique le choix de la petite taille de fenêtre par le critère de parcimonie, et la présence de la composante dans une plage de fréquences étendue implique un nombre important de régions sélectionnées à l'instant correspondant.

On voit cependant que le nombre de rectangles sélectionnés peut également être important dans des zones où il n'y a pas de transitoire marqué. Ceci est dû au fait déjà évoqué à la fin de la partie 2.4, que la petite taille de fenêtre peut être sélectionnée dans des régions pour laquelle aucune des deux tailles de fenêtre n'est bien adaptée et où la valeur d'entropie reste grande pour les deux tailles de fenêtre. C'est en raison de ce phénomène qu'une variante de l'algorithme de puzzle temps-fréquence avec seuillage sur les valeurs d'entropie a été introduite. Ainsi, pour effectuer une bonne détection des transitoires, il est nécessaire d'introduire ce seuillage. On vérifie effectivement sur la courbe du bas de la figure 3.6 que le problème est ainsi résolu, le comptage du nombre de rectangles sélectionnés ne présente des pics que pour les transitoires d'attaque du signal. On peut ainsi envisager une détection automatique des transitoires à partir de la fonction du temps obtenue, par exemple par seuillage des valeurs afin de détecter les pics dépassant un niveau suffisant. Sur l'exemple étudié, on voit que le choix du seuil ne sera pas "critique". Il peut évidemment en aller autrement pour des signaux plus complexes.

L'intérêt de cette méthode dans le cadre de l'utilisation des décompositions à partir de l'algorithme de puzzle temps-fréquence est qu'elle permet de détecter les transitoires sans avoir à introduire de calculs supplémentaires, la décomposition et la détection étant réalisés simultanément.

3.2.3 Détection dans un cadre général

Dans un cadre où l'on ne cherche qu'à effectuer la détection de composantes transitoires sans décomposition du signal, une méthode simplifiée donnant des résultats assez similaires peut être utilisée, puisqu'il n'est alors plus nécessaire de calculer la

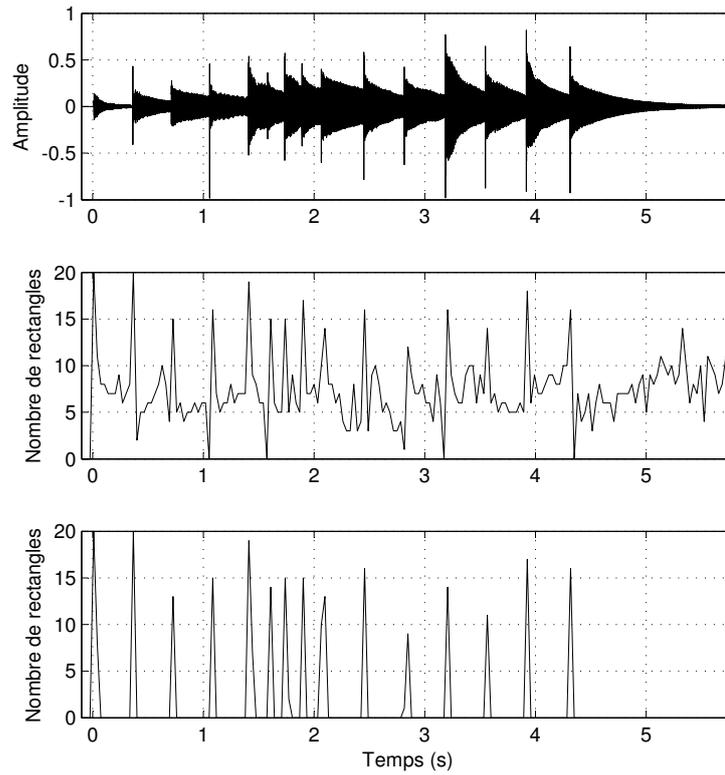


FIG. 3.6 – Évolution au cours du temps du nombre de rectangles temps-fréquence sélectionnés pour la petite fenêtre par l’algorithme de puzzle temps-fréquence sur un son de glockenspiel : en haut, forme d’onde du signal de glockenspiel, au milieu, nombre de rectangles sélectionnés pour la petite fenêtre lors de la première itération de l’algorithme de puzzle temps-fréquence (variante 1) sans seuillage, en bas nombre de rectangles sélectionnés pour la petite fenêtre lors de la première itération de l’algorithme de puzzle temps-fréquence (variante 1) avec seuillage

transformée de Gabor avec la grande fenêtre et d'effectuer une phase de choix entre la petite et la grande fenêtre. Il suffit alors de calculer la transformée de Gabor du signal avec une petite fenêtre, puis de calculer la valeur du critère entropique dans chaque rectangle d'un pavage rectangulaire du plan temps-fréquence, et enfin de compter pour chaque instant le nombre de rectangles pour lesquels la valeur du critère est inférieure à un seuil donné. Comme on peut le voir sur la figure 3.7, on obtient ainsi sensiblement la même courbe que celle obtenue précédemment sur la figure 3.6 en comptant le nombre rectangles sélectionnés dans l'algorithme de puzzle temps-fréquence avec seuillage.

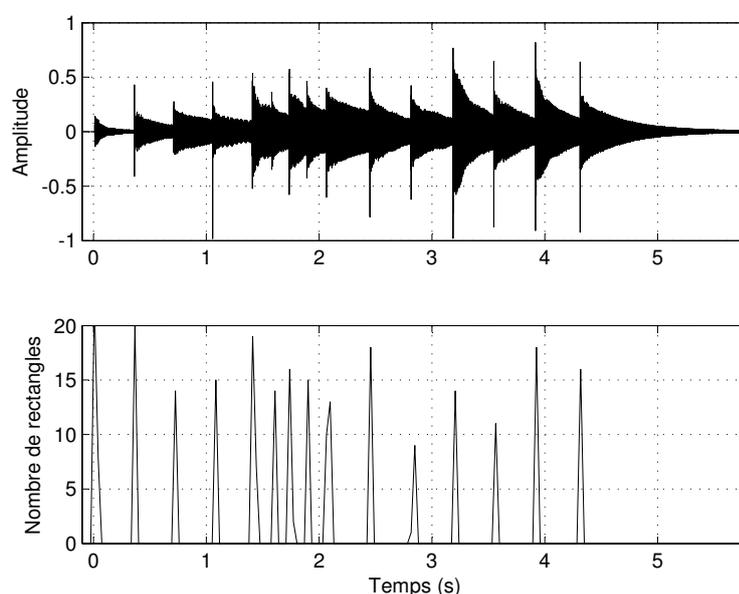


FIG. 3.7 – Évolution au cours du temps du nombre de rectangles temps-fréquence dont l'entropie est inférieure à un seuil fixé : en haut, forme d'onde du signal de glockenspiel, en bas nombre de rectangles dont la valeur d'entropie est en dessous d'un seuil fixé

On peut noter que cette méthode implique l'introduction d'un seuil sur les valeurs d'entropie, ce qui se traduit par l'ajout d'un paramètre à régler pour l'utilisateur et risque par conséquent de nuire à la simplicité d'utilisation de la méthode. On peut éviter l'introduction du seuil dans cette étape en abordant le problème de manière un peu différente. En conservant les deux mêmes premières étapes, c'est à dire le calcul de la transformée de Gabor avec une petite fenêtre, puis le calcul des valeurs du critère entropique, on peut ensuite directement sommer⁶ pour chaque instant

⁶On prend ici la somme des valeurs d'entropie, ce qui est équivalent à prendre la moyenne

les valeurs d'entropie obtenues dans les rectangles qui se trouvent à cet instant. Lorsqu'un transitoire est présent, on a une faible valeur d'entropie pour plusieurs rectangles à l'instant du transitoire, ainsi, la fonction obtenue par sommation des valeurs d'entropie a une valeur plus faible dans la zone du transitoire. On peut ainsi détecter les transitoires à partir de cette fonction en recherchant les "puits" de la fonction (diminutions brusques de la valeur suivies d'une remontée). On vérifie sur la figure 3.8 que ces puits sont bien marqués dans le cas de l'exemple du glockenspiel et donc bien exploitable pour effectuer la détection.

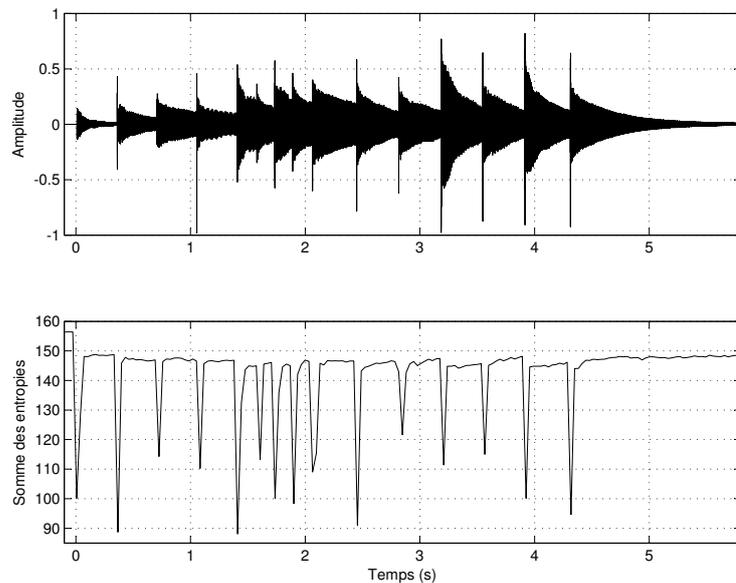


FIG. 3.8 – Évolution au cours du temps de la somme des valeurs d'entropie obtenues pour un pavage rectangulaire du plan temps-fréquence sur un son de glockenspiel : en haut, forme d'onde du signal de glockenspiel, en bas, somme des valeurs d'entropie

On peut envisager deux cadres d'utilisation des fonctions de détection que nous venons de proposer :

Cadre supervisé : Dans un cadre supervisé, l'utilisateur détermine une zone du signal où il souhaite détecter un unique transitoire. On cherche alors le pic (ou le puits, selon l'approche utilisée) le plus marqué de la fonction de détection pour détecter le transitoire. Pour cela, une méthode simple est de repérer

arithmétique des valeurs pour la détection. On peut se demander si d'autres moyennes (par exemple pondérées) ne peuvent pas être mieux appropriées. Pour répondre, il faudrait mettre en place une méthode d'évaluation des algorithmes de détection et comparer les résultats obtenus avec plusieurs moyennes.

l'extremum (maximum ou minimum selon l'approche utilisée) de la fonction dans la zone sélectionnée.

Cadre non-supervisé : Dans un cadre non-supervisé, on cherche à détecter les transitoires de manière automatique sur l'ensemble du signal. Pour cela, il va donc être nécessaire de repérer les pics (ou les puits, selon le cas) de la fonction de détection. On peut par exemple utiliser un simple seuillage de la fonction de détection pour repérer les transitoires. D'autres approches plus fines peuvent être envisagées. On peut consulter [2] pour une présentation des méthodes de détection de pics sur la fonction de détection.

3.2.4 Reconstruction

Un des intérêts de notre approche est qu'elle permet naturellement le traitement et la reconstruction des composantes transitoires détectées. En effet, en seuillant les valeurs d'entropie obtenues sur la transformée de Gabor avec la petite fenêtre à l'instant correspondant au transitoire, on peut identifier les composantes dont l'énergie est bien concentrée à cet instant. On peut ensuite reconstruire ces composantes par inversion de la transformée de Gabor en ne conservant que les rectangles sélectionnés après seuillage.

Une illustration de la méthode de détection avec reconstruction par cette méthode est donnée sur l'exemple suivant.

Exemple 3.3 *La figure 3.9 présente le résultat de la détection obtenu sur un signal de musique indienne (tabla et sitar). La méthode utilisée est le comptage au cours du temps du nombre de rectangles dont l'entropie est au dessous d'un seuil fixé. On voit que la méthode permet bien de détecter et de reconstruire l'attaque des sons percussifs en préservant les partiels dans certaines zones riches en composantes tonales.*

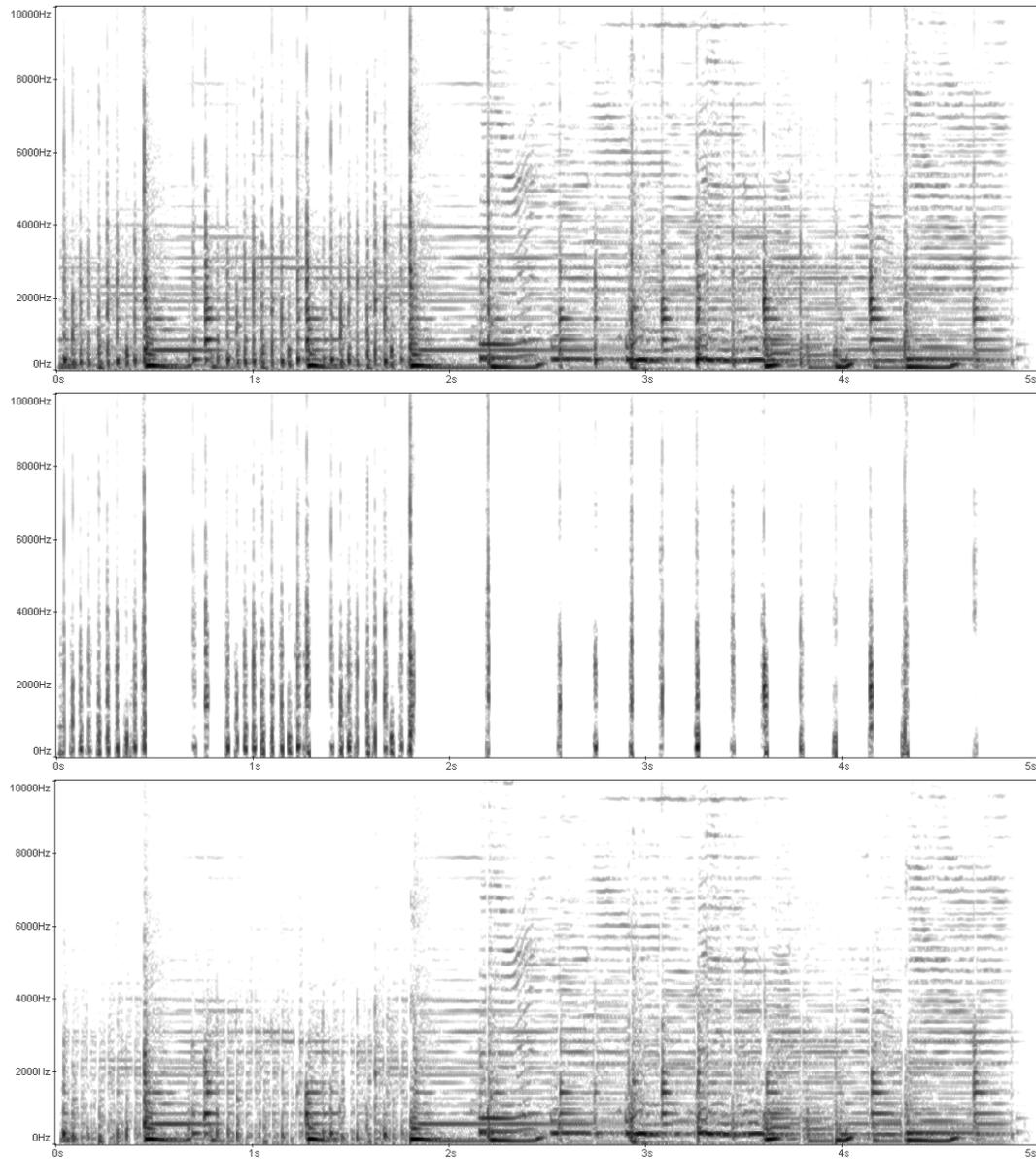


FIG. 3.9 – Exemple de détection de composantes transitoires : en haut, spectrogramme du signal original, au milieu, spectrogramme des composantes transitoires extraites, en bas, spectrogramme du résidu obtenu par soustraction des composantes transitoires extraites au signal original

Chapitre 4

Application

L'ensemble des transformées et des algorithmes décrits dans les chapitres 2 et 3 a fait l'objet d'implémentations de test en utilisant le logiciel de calcul numérique Matlab afin de valider les concepts proposés, d'étudier le comportement des algorithmes en fonction de leurs divers paramètres et d'évaluer leurs possibilités de traitement sur des extraits de signaux réels.

Ces implémentations rapides, même si elles ont effectivement permis de tester et d'évaluer les algorithmes, ne fournissent pas des niveaux d'ergonomie, de finalisation et d'efficacité de calcul suffisants pour pouvoir être réellement utilisés comme des outils pour le design industriel comme décrits au chapitre 1.

Si l'on considère la boucle de traitement des sons par modification de leur représentation temps-fréquence, présentée en figure 1.2 au chapitre 1, on voit que les chapitres 2 et 3 ont permis de proposer les outils pour construire la boucle de transformation temps-fréquence, modification, puis reconstruction. Pour obtenir des outils exploitables pour le design sonore industriel, il faut maintenant s'intéresser à l'interfaçage de cette boucle avec l'utilisateur. Ceci se fait à travers l'affichage graphique des représentations, le contrôle des modifications et la restitution sonore des sons modifiés.

Les aspects d'interfaçage des traitements ne sont généralement pas étudiés en détail dans une thèse de traitement des signaux numériques. Cependant, compte tenu du contexte de la thèse et de son objectif, l'étude a pu être poussée à son terme puisque GENESIS commercialise aujourd'hui un logiciel intégrant la boucle de traitement temps-fréquence décrite en introduction. Ce logiciel est nommé LEA (Logiciel d'Expertise Acoustique).

Nous allons d'abord préciser les étapes qui ont permis d'aboutir à la réalisation du logiciel, puis, après une courte présentation générale des fonctionnalités du logiciel, qui ne se limitent pas au seul traitement temps-fréquence des signaux, nous décrirons plus en détail comment la boucle de traitement temps-fréquence a effectivement été mise en place, en insistant sur les choix retenus pour l'interfaçage avec l'utilisateur.

Enfin, nous décrirons rapidement quelques traitements qui ont été réalisés avec LEA dans des cadres appliqués de design sonore.

4.1 Logiciel d'Expertise Acoustique (LEA)

4.1.1 Genèse du projet LEA

Un prototype embryonnaire d'outil graphique de traitement temps-fréquence des sons avait été développé au cours de mon stage de DEA réalisé à GENESIS [24]. Les points étudiés au cours du début de thèse ont rapidement été intégrés à ce prototype auquel un certain nombre d'améliorations ergonomiques ont également été apportées afin de pouvoir tester dans de bonnes conditions la pertinence des approches proposées. Ce prototype a été montré à des industriels pour leur présenter le concept proposé et recueillir plus précisément leurs besoins en matière d'outils pour le design sonore industriel. Les traitements disponibles ont retenu l'attention d'un certain nombre d'industriels, ce qui a débouché sur des demandes de traitements sur des exemples sonores afin d'évaluer la capacité des outils et également sur des demandes d'outils logiciels commerciaux permettant de réaliser les traitements.

En vue du développement futur d'un produit commercial, et pour dépasser les limites du prototype liées au développement sous le logiciel Matlab, notamment en matière de taille des données traitées, de vitesse de calcul et de possibilités d'interfaçage, il a alors été décidé de mettre en place une structure logicielle d'outil d'édition temps-fréquence spécifique dans laquelle des implémentations optimisées des algorithmes proposés dans la thèse pourraient être intégrés. La création de cette structure logicielle, codée en C/C++, a débuté lors du stage de fin d'école d'ingénieur de Florent Poirot [38], que j'ai encadré d'avril à septembre 2003 au sein de GENESIS.

Cependant, compte tenu de l'engagement d'un partenaire industriel, le stage a été intégré dans un projet commercial en vue de concevoir un logiciel d'expertise acoustique comprenant notamment l'outil d'édition temps-fréquence. Ainsi, le stage a été consacré à la conception du logiciel complet, réalisée de manière commune par Florent Poirot et moi-même, et au début du développement du logiciel. Dans le développement, Florent Poirot s'est principalement consacré à l'interfaçage homme-machine et je me suis principalement consacré aux algorithmes de traitement du signal. Le développement s'est ensuite poursuivi avec l'embauche par GENESIS de Florent Poirot et l'intervention d'autres ingénieurs de GENESIS. Le développement et la mise au point de la première version du logiciel s'est déroulé sur 21 mois, d'avril 2003 à décembre 2004.

Au moment de la rédaction de ce document, le logiciel est utilisé par des équipes de recherche de grands groupes industriels (Valeo, Renault, Volkswagen, SCE Klaxon) et de laboratoires de recherche (France Telecom R&D, équipe d'acous-

tique perceptive du Laboratoire de Mécanique et d'Acoustique de Marseille).

4.1.2 Description générale du logiciel LEA

LEA est un logiciel dédié à l'analyse de mesures acoustiques, l'étude de la qualité sonore et le design sonore industriel, permettant l'analyse et la transformation de signaux numériques.

Il est constitué de plusieurs modules consacrés à différents types d'analyses et de traitements que nous allons rapidement décrire.

Module temporel

Le module temporel, dont l'aspect est présenté en figure 4.1, regroupe les fonctions de base que l'on trouve généralement dans les éditeurs de sons. Il permet l'édition des signaux à partir de leur forme d'onde, la comparaison de plusieurs signaux en superposant leurs représentations et en repérant des zones temporelles au moyen de curseurs et propose des traitements courants tels que la copie/collage/coupe de segments, le mixage de signaux, l'application de gains et le rééchantillonnage.

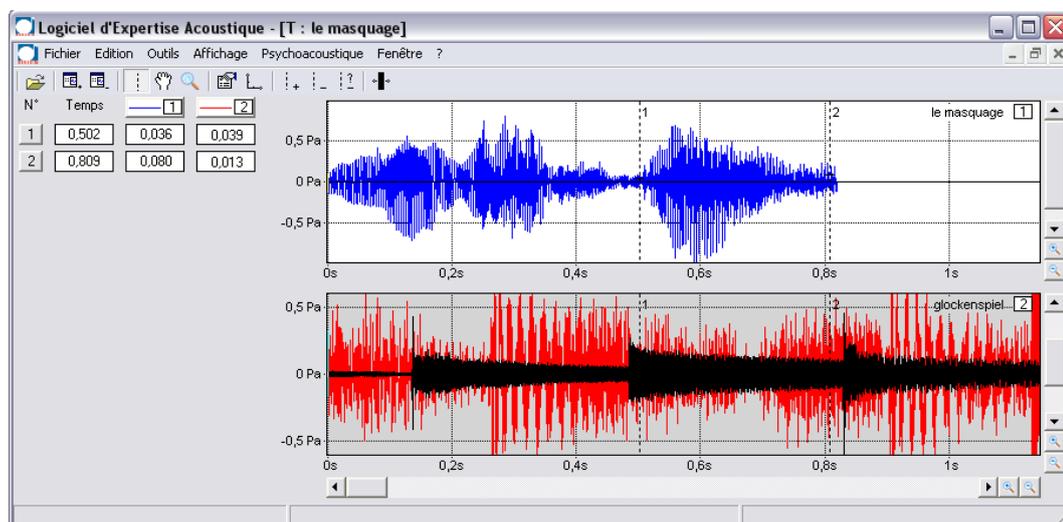


FIG. 4.1 – Exemple d'affichage obtenu avec le module temporel de LEA

Un égaliseur en bande de Bark permettant le filtrage en temps-réel des sons stéréophoniques est également associé au module temporel. Bien qu'associé au module temporel, cet égaliseur réalise en fait un traitement temps-fréquence pour effectuer ce filtrage, puisqu'il utilise une implémentation temps-réel de la transformée

de Gabor et de son inversion afin d'appliquer dans chacune des bandes Bark une atténuation définie en temps-réel par l'utilisateur à l'aide d'un ensemble de curseurs.

Module spectral

Le module spectral, dont l'aspect est présenté en figure 4.2, permet d'effectuer une analyse spectrale des signaux par la méthode du périodogramme de Welch avec la possibilité de comparer les spectres entre eux ou avec des gabarits. Diverses opérations d'analyse de niveaux sur les spectres sont également mises en oeuvre.

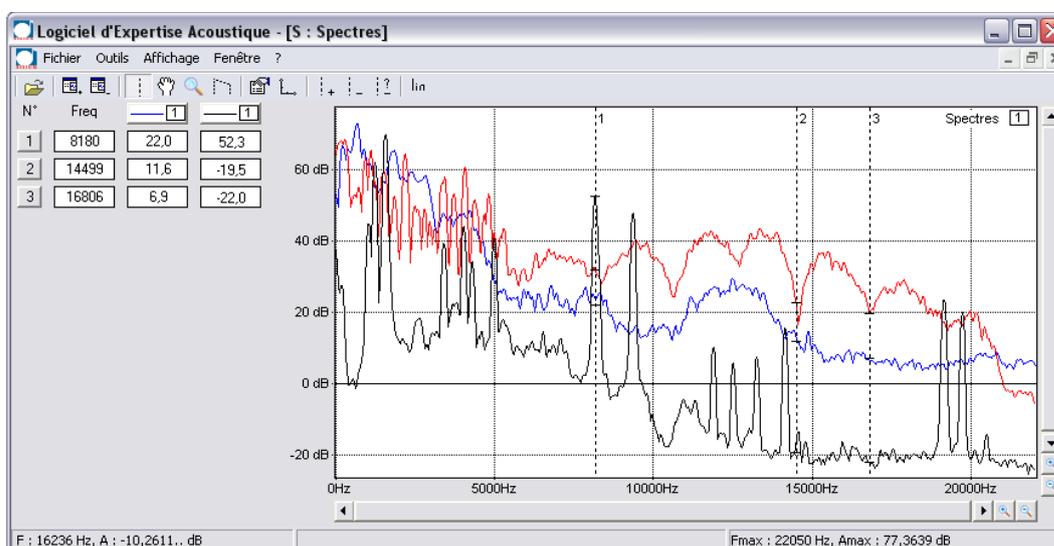


FIG. 4.2 – Exemple d'affichage obtenu avec le module spectral de LEA

Module temps-fréquence

Le module temps-fréquence est une implémentation complète de la boucle de transformation des sons par modification de leur représentation temps-fréquence décrite au chapitre 1, et constitue en conséquence une réalisation de l'objectif de cette thèse.

Le module permet l'affichage de représentations temps-fréquence des signaux à partir de leur transformée de Gabor et le traitement temps-fréquence des signaux par sélection graphique de régions du plan temps-fréquence et application de gains sur ces régions.

Une présentation plus détaillée de ce module est faite en section 4.1.3.

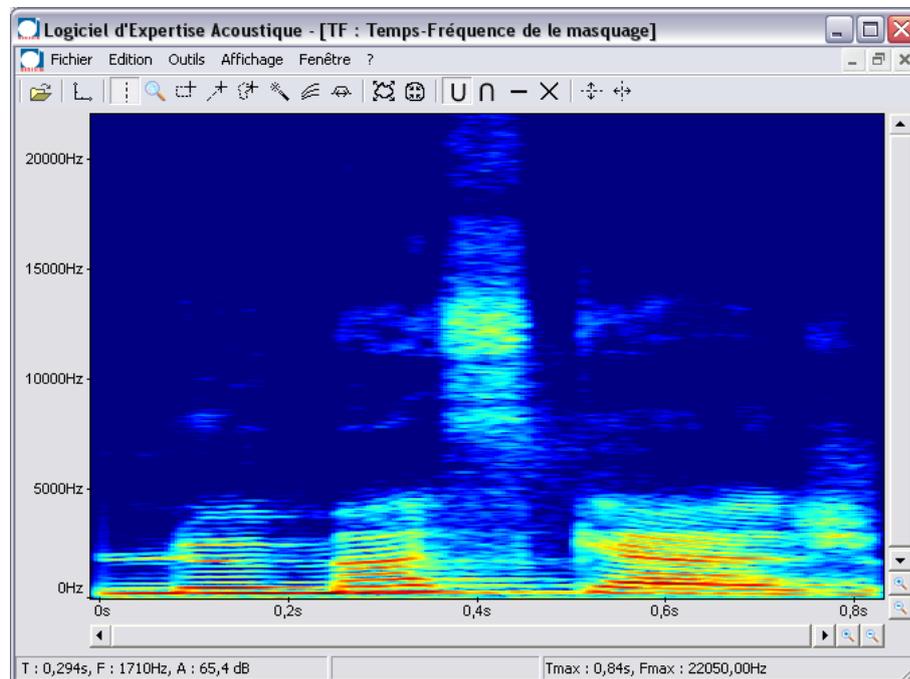


FIG. 4.3 – Exemple d’affichage obtenu avec le module temps-fréquence de LEA

Module psychoacoustique

Le module psychoacoustique permet l’estimation, à partir de modèles algorithmiques, des principaux indices psychoacoustiques des sons (sonie, acuité, force de fluctuation, rugosité). Il permet d’obtenir des valeurs globales des indices sur un segment de signal, comme le montre l’exemple présenté en figure 4.4, mais également d’obtenir des courbes montrant l’évolution en fonction du temps de certains des indices. Ce module s’appuie sur le projet de recherche mené au sein de GENESIS sur la sonie des sons non-stationnaires, dont fait partie le travail de thèse d’Isabelle Boulet sur la sonie des sons impulsionnels (soutenance prévue en 2005).

Module reporting

LEA permet également d’exporter les représentations et les résultats d’analyse directement dans le logiciel Word pour faciliter la réalisation de rapports d’analyse. Selon les besoins des utilisateurs, il est également possible de chaîner automatiquement des analyses afin d’optimiser le dépouillement systématique de mesures acoustiques.

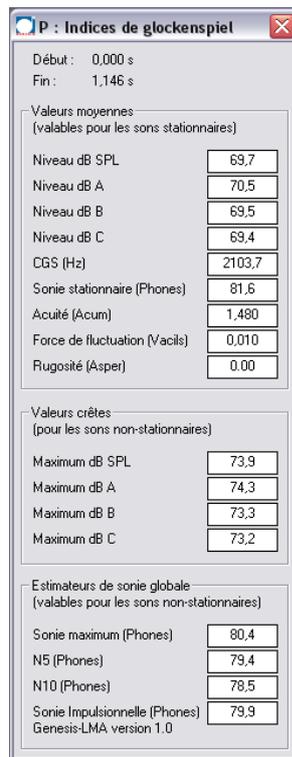


FIG. 4.4 – Exemple d’affichage des valeurs d’indices globaux obtenu avec le module psychoacoustique de LEA

4.1.3 Fonctionnalités du module temps-fréquence

Cette section décrit rapidement les fonctionnalités disponibles dans le module temps-fréquence de LEA. Plus de détails sur la conception et l'implémentation du module sont donnés dans [25].

Calcul des transformées temps-fréquence

Le module temps-fréquence met en place la boucle de traitement temps-fréquence des signaux décrite en 1.3 en utilisant la transformée de Gabor pour construire les représentations temps-fréquence. Pour cela, nous avons développé une implémentation particulièrement soignée de la transformée de Gabor discrète en dimension finie décrite dans [25] en nous appuyant sur une bibliothèque de fonctions optimisées de traitement du signal. Nous disposons ainsi d'une implémentation très efficace permettant le traitement de sons de quelques minutes dans un temps "raisonnable", comme le souhaitaient les utilisateurs.

Les paramètres d'analyse réglables par l'utilisateur pour la transformée sont la taille de fenêtre, le type de fenêtre, la taille de transformée de Fourier et le taux de recouvrement entre deux fenêtres successives. Pour faciliter le réglage de la taille de fenêtre, un outil d'adaptation supervisée de la taille de fenêtre est disponible. Il est basé sur l'approche présentée en 2.3.3 : l'utilisateur sélectionne à la souris la zone temps-fréquence contenant les composantes pour lesquelles il veut adapter la représentation, puis la meilleure taille de fenêtre est choisie en minimisant un critère entropique.

Affichage des représentations

L'affichage des représentations est de type sonographique et se fait à partir du spectrogramme avec une échelle de niveau en dB. L'affichage des représentations a également été particulièrement soigné, afin d'assurer une bonne ergonomie de visualisation. Pour cela, nous avons mis en place un rééchantillonnage spécifique du spectrogramme pour l'affichage permettant de bien localiser les zones de forte énergie à tous les niveaux de zoom. Ce rééchantillonnage permet notamment d'éviter des pertes (disparitions) d'informations visuelles à certains niveaux de zoom que l'on constate généralement dans les logiciels affichant des représentations temps-fréquence.

Pour optimiser la visualisation de la représentation, l'utilisateur a également la possibilité d'adapter l'échelle de couleur utilisée pour le codage des niveaux sur le spectrogramme en choisissant les couleurs qui la constitue et la plage de niveaux codés sur cette échelle.

Traitements

Les traitements temps-fréquence utilisent les multiplicateurs de Gabor évoqués en 2.1.3. Plus précisément, l'utilisateur sélectionne une région temps-fréquence à l'aide d'un ensemble d'outils spécifiques qu'il peut combiner suivant trois modes de superposition des sélections (union, intersection, différence). Le traitement est alors réalisé en appliquant un gain fixé par l'utilisateur sur les coefficients sélectionnés, puis en reconstruisant le signal à partir des coefficients ainsi modifiés. Le logiciel affiche alors le spectrogramme du signal reconstruit. L'utilisateur peut écouter ce signal pour juger si le traitement lui convient.

Deux types d'outils de sélection sont disponibles :

- D'une part, l'utilisateur peut tracer directement la sélection (un rectangle, un segment de droite, ou une forme quelconque "à main levée") avec des outils simples de sélection non-automatiques.
- D'autre part, l'utilisateur peut utiliser des algorithmes de détection supervisés à travers des outils de sélection semi-automatiques. Pour cela, il dispose de deux outils spécifiques.

Le premier est appelé "baguette magique" en référence à l'outil équivalent disponible dans les logiciels de retouche d'images. Il permet de sélectionner une région connexe au point cliqué contenant les points dont le niveau est proche de celui du point cliqué, avec une tolérance fixée par l'utilisateur.

Le deuxième est un outil de détection de composantes localement harmoniques mettant en œuvre la méthode de détection par programmation dynamique décrite en 3.1.6. L'utilisateur précise le rang des harmoniques recherchés et délimite la zone de recherche des composantes en indiquant graphiquement la position approximative d'un des harmoniques et une largeur de recherche autour cette position.

4.2 Exemples d'utilisations industrielles et académiques

Au cours de la thèse, le logiciel LEA, et le prototype développé sous Matlab qui l'a précédé, ont été utilisés pour réaliser des traitements effectifs dans des cadres de recherche industrielle et académique. Ces traitements ont été réalisés pour les besoins propres de GENESIS, mais également pour des partenaires industriels, dans le cadre de collaborations et de contrats commerciaux. Nous allons décrire les principaux traitements réalisés afin d'illustrer l'intérêt applicatif des éléments décrits dans cette thèse.

4.2.1 Traitements de sons automobiles

GENESIS a réalisé pour le compte d'un constructeur automobile deux expertises de traitement des sons automobiles pour le design sonore. J'ai réalisé la première de ces deux expertises, la deuxième a été réalisée par un autre ingénieur de la société.

L'objectif de ces expertises était l'évaluation des possibilités de traitement des outils proposés et développés dans le cadre de la thèse sur les sons automobiles. Pour cela, quelques traitements-types pour l'étude et le design de l'environnement sonore dans l'habitacle ont été identifiés, et une banque de sons a été confiée à GENESIS pour traitement.

On peut distinguer deux grandes classes dans les traitements réalisés :

- d'une part le traitement de composantes localement harmoniques, à travers l'extraction et la suppression de la composante harmonique créée par le moteur dans des prises de sons de montées en régime, et à travers l'extraction et la suppression de bruits de turbo,
- d'autre part, le traitement de composantes transitoires, à travers le décliquage de prises de sons (par exemple la suppression de claquement de porte-clefs dans des montées en régime), l'extraction et la suppression de bruit de chocs dus aux raccords de routes, l'extraction du "bruit de claquement" de ralentis de moteurs diesel.

Les outils proposés dans la thèse ont permis d'obtenir des résultats de traitements jugés satisfaisants (le constructeur a depuis acquis des licences du logiciel LEA).

Pour aller plus loin dans l'utilisation de ce type de traitements pour le design sonore, GENESIS a par la suite étudié les possibilités de resynthèse en temps-réel des composantes extraites. L'étude s'est portée en particulier sur les sons de montée en régime. Pour cela, j'ai réalisé une version spécifique de l'outil de détection de composantes localement harmoniques permettant d'estimer et de stocker les valeurs des paramètres de la composante détectée (fréquence instantanée et amplitude des harmoniques en fonction du temps). Ces paramètres extraits sont ensuite utilisés pour alimenter un module de synthèse en temps-réel. La composante bruitée du moteur, obtenue en supprimant la composante localement harmonique sur le signal original, est elle aussi modélisée. Pour cela, on estime à partir du spectrogramme le niveau de bruit dans des bandes déterminées en fonction du temps. Ces données sont elles aussi utilisées pour alimenter un module de synthèse en temps-réel. L'utilisateur dispose ainsi en fin d'analyse d'un module de synthèse temps-réel du son complet. La qualité du son synthétisé avec cette méthode est bonne, le son obtenu est très proche du son original et très réaliste.

Avec ce module de synthèse temps-réel, l'utilisateur peut rejouer le son en contrôlant maintenant en temps-réel la valeur de régime. Il peut également modifier très simplement les caractéristiques du signal indépendamment pour chaque valeur de régime. Ce module peut alors par exemple être utilisé dans un simulateur

de conduite, permettant ainsi d'étudier facilement la perception du son dans des conditions de conduite simulées réalistes.

Exemple 4.1 *La figure 4.5 présente un exemple de traitement réalisé sur une montée en régime automobile. La prise de son est réalisée dans l'habitacle du véhicule. On peut voir sur la représentation du haut de la figure le spectrogramme du signal original. On présente en dessous les résultats du traitement réalisé avec LEA : d'abord l'extraction de la composantes localement harmonique, puis sa suppression. Enfin, sur la représentation du bas, on voit le signal synthétisé avec le module temps-réel dans le cas où l'on utilise la même loi de commande du régime que le signal original.*

4.2.2 Design sonore des sons de démarreurs

J'ai encadré le stage de DEA de Denis Jolivet [27] réalisé au sein de GENESIS sur l'étude des sons de démarreurs automobiles. Cette étude s'est appuyée sur une collaboration avec un équipementier automobile entamée autour du développement de LEA, l'équipementier ayant participé à la définition des spécifications du logiciel. L'objectif de ce stage était l'étude des possibilités de synthèse de bruit de démarreur à partir de ses caractéristiques mécaniques. Pour cela, une étude mécanique du démarreur a été menée. Cette étude a été couplée à une analyse d'enregistrements de bruit de démarreurs à l'aide d'outils de traitement du signal. Cette analyse a principalement été menée à l'aide du logiciel LEA.

Le stage a abouti à la création d'un modèle semi-physique de synthèse du son de démarreur s'appuyant à la fois sur les caractéristiques mécaniques du démarreur (type de démarreur, nombre de dents des engrenages, ...) et sur des éléments extraits d'enregistrements de démarreurs (loi de variation des vitesses de rotation, bruits de chocs, mesure de résonances).

LEA a permis de réaliser les analyses et les traitements nécessaires à cette étude. En particulier, l'outil de détection de composantes localement harmoniques a été utilisé pour extraire les lois de variations de vitesse du rotor et du volant moteur, qui sont utilisées pour commander le module de synthèse. Le module temps-fréquence de LEA a également été utilisé pour extraire les bruits de chocs apparaissant dans les différentes phases de démarrage qui sont également directement utilisés par le module de synthèse.

4.2.3 Étude de la perception des sons de plaques

L'utilisation de LEA ne se limite pas aux utilisations en contexte industriel. Le logiciel constitue en effet un outil privilégié pour l'étude du lien entre les caractéristiques physiques des signaux et leur perception. Il peut donc être utilisé avec

4.2. EXEMPLES D'UTILISATIONS INDUSTRIELLES ET ACADÉMIQUES 141

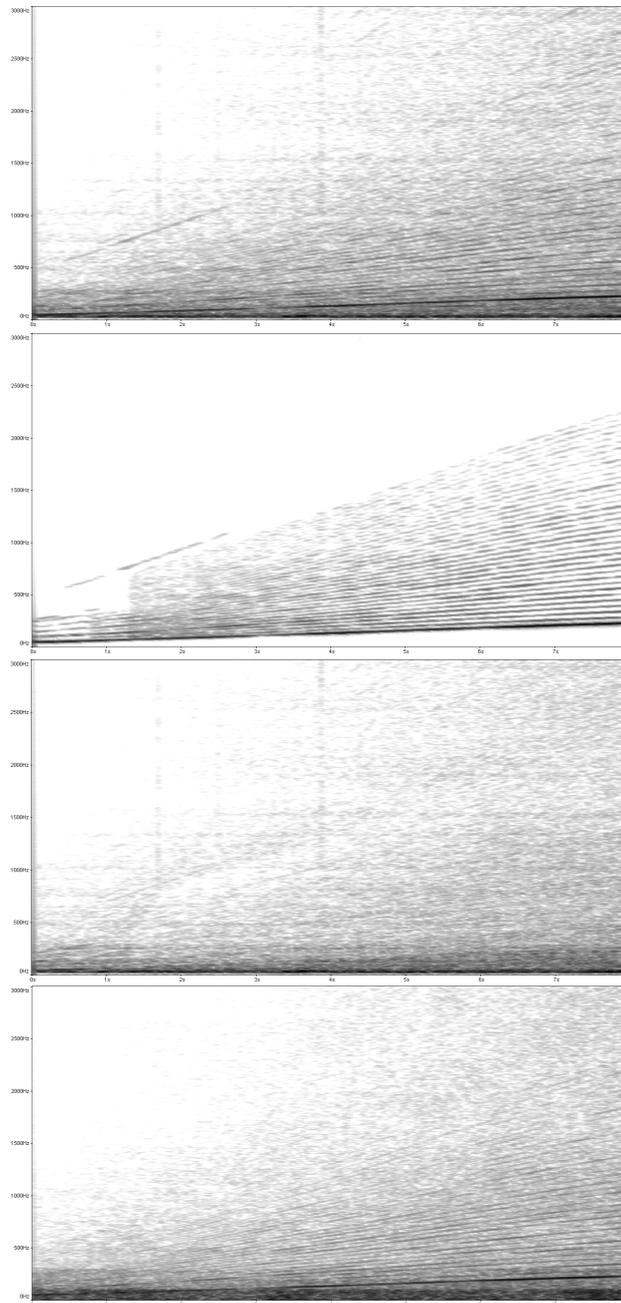


FIG. 4.5 – Exemple de traitement de montée en régime automobile : en haut, spectrogramme du signal original, en dessous, spectrogramme de la composante localement harmonique extraite, en dessous, spectrogramme du signal obtenu par suppression de la composante localement harmonique, en bas, spectrogramme du signal synthétisé par le module temps-réel en suivant la loi de variation de régime du signal original

profit dans les travaux de recherche en psychoacoustique, dont l'objet et justement l'étude de ce lien.

Ainsi, par exemple, LEA est actuellement utilisé par Florence Demirdjian dans le cadre de sa thèse en psychomécanique (couplage d'études vibroacoustiques et psychoacoustiques) réalisée au LMA (Laboratoire de Mécanique et d'Acoustique, Marseille). Sa recherche porte sur l'étude de la perception auditive des plaques en acier rectangulaires encastrées soumises à une force impulsive.

LEA permet dans ce cadre de réaliser le filtrage des modes de résonance de la plaque sur des enregistrements ou sur des sons de synthèse pour repérer l'influence perceptive des différents modes. En particulier, LEA est utilisé pour identifier très facilement les modes responsables de la perception de "battements" dans les sons, par écoute des signaux obtenus après suppression individuelle des différents modes. LEA facilite également l'identification des modes responsables de dissemblances perceptives entre des sons enregistrés et des sons obtenus par modélisation vibratoire, en utilisant la même approche. Dans ce cadre, plus que dans la complexité des traitements effectués, l'intérêt de LEA réside principalement dans la facilité et la rapidité de traitement qui permettent une identification très simple des composantes d'intérêt.

Chapitre 5

Conclusion

Dans ce chapitre de conclusion, nous allons d'abord décrire de manière synthétique les résultats qui ont été présentés dans cette thèse, puis nous donnerons quelques perspectives envisageables.

5.1 Résultats

5.1.1 Résultats théoriques

Décompositions atomiques

Dans le chapitre 2, nous avons proposé plusieurs approches pour la construction de transformées temps-fréquence atomiques permettant de dépasser certaines limitations des approches classiques rigides (transformée de Gabor ou transformée en ondelettes), tout en assurant l'inversibilité des transformées et en conservant une charge de calcul limitée.

Dans un premier temps, en 2.2, nous avons étudié comment, en conservant la covariance de la transformée selon une des dimensions, soit le temps, soit la fréquence, il était possible de relâcher très fortement la contrainte sur l'évolution de la résolution des atomes selon l'autre dimension. Ceci nous permet de construire des représentations covariantes par translation fréquentielle dans lesquelles la résolution temporelle des atomes évolue assez librement avec le temps et des représentations covariantes par translation temporelle dans lesquelles la largeur de bande des atomes évolue assez librement avec la fréquence (interprétables comme des bancs de filtres pouvant suivre une loi donnant la largeur de bande en fonction de la fréquence presque quelconque). La liberté d'évolution de la résolution des atomes est exprimée à l'aide d'une condition d'admissibilité sur une famille d'atome utilisée pour générer les atomes de la décomposition. Une méthode de construction d'atomes vérifiant cette condition d'admissibilité est proposée. Avec cette approche, nous

avons construit une transformée de Gabor instationnaire, c'est à dire dans laquelle la taille de fenêtre évolue dans le temps, et nous avons vérifié son intérêt pour l'analyse et le traitement de signaux naturels. Nous avons également construit une transformée utilisant des paquets d'atomes temps-fréquence construits à partir d'atomes de la transformée de Gabor et de la transformée en ondelettes. Le calcul explicite de la fréquence centrale et de la largeur de bande de ces paquets en fonction de ceux des atomes pères utilisés pour la construction est effectué. Ceci permet alors d'utiliser la transformée proposée pour réaliser une analyse temps-fréquence suivant une loi de fréquence choisie librement, par exemple en suivant une échelle de fréquence perceptive.

Dans un deuxième temps, en 2.4, nous avons proposé une approche pour adapter la résolution des atomes simultanément selon les deux dimensions. Pour cela, nous avons mis en place l'algorithme itératif adaptatif de puzzle temps-fréquence. Cet algorithme, construit à partir de transformées Gabor calculées avec plusieurs fenêtres, permet de décomposer le signal en utilisant des atomes aux caractéristiques de localisation temps-fréquence différentes dans différentes régions rectangulaires du plan temps-fréquence. Nous avons étudié numériquement le comportement de cet algorithme et notamment vérifié que l'algorithme permet la reconstruction du signal avec une erreur de reconstruction aussi petite que souhaitée (sans malheureusement pouvoir le prouver analytiquement).

Adaptativité

Nous avons étudié dans la section 2.3 une approche permettant d'introduire une part d'adaptativité dans les transformées atomiques rigides. Cette approche est basée sur l'optimisation de la parcimonie des représentations mesurée à l'aide d'un critère entropique. Nous avons repéré dans quel cadre cette approche pouvait être utilisée de manière satisfaisante, en montrant les limites d'une approche globale et proposant un cadre d'adaptation locale supervisée dans le plan temps-fréquence qui s'est montré pertinent pour une application aux signaux naturels.

En étendant ce cadre d'adaptation locale au cas non-supervisé, nous avons proposé l'algorithme de puzzle temps-fréquence évoqué plus haut. L'adaptativité de cet algorithme est particulièrement intéressante lorsqu'il utilise deux fenêtres de tailles très différentes. En effet, l'interprétation multicouche de l'algorithme permet alors de séparer dans le signal les composantes bien représentées avec une petite taille de fenêtre, généralement interprétables comme les composantes transitoires du signal, des composantes bien représentées avec une grande taille de fenêtre, généralement interprétables comme les composantes tonales du signal. Pour pousser plus loin cette interprétation, nous avons proposé des variantes de l'algorithme assurant que les composantes séparées soient bien de la nature souhaitée, transitoire ou tonale. En particulier, en introduisant un seuillage sur les valeurs d'entropies, l'algorithme per-

met de séparer les composantes transitoires des composantes tonales en conservant dans le résidu les composantes d'une autre nature, en particulier les composantes bruitées.

Détection de composantes harmoniques

Nous avons proposé en 3.1 une méthode de détection de composantes localement harmoniques robuste au bruit. Cette méthode s'appuie sur un critère énergétique régularisé calculé à partir du spectrogramme du signal. Deux approches ont été étudiées pour l'optimisation de ce critère :

- une approche stochastique utilisant l'algorithme de recuit simulé,
- une approche déterministe utilisant une méthode de programmation dynamique.

Nous avons décrits les difficultés d'utilisation du recuit simulé qui limitent sa mise en œuvre dans certains cas de figure. Nous préférons en conséquence utiliser une méthode d'optimisation par programmation dynamique qui consiste à rechercher le plus court chemin dans un graphe.

Nous avons étudié l'influence du paramètre de régularité du critère sur les résultats de détection et vérifié la bonne robustesse au bruit de l'algorithme.

Enfin, nous avons signalé une méthode simple de reconstruction des composantes utilisant les multiplicateurs de Gabor et bien adaptée à notre cadre de travail.

Détection de composantes transitoires

Nous avons exploré rapidement en 3.2 les possibilités d'utilisation des éléments mis en œuvre dans l'algorithme de puzzle temps-fréquence pour la détection de composantes transitoires. Un intérêt de cette approche est qu'elle permet la reconstruction des composantes détectées. Nous avons vérifié qu'on obtenait des résultats pertinents sur des exemples simples.

5.1.2 Résultats pratiques

Réalisation de l'objectif : LEA

L'objectif initial du projet de recherche, c'est à dire la création d'un outil de traitement temps-fréquence des sons pour le design sonore industriel, est atteint. En effet, le logiciel LEA actuellement commercialisé par GENESIS intègre, à travers son module temps-fréquence, une boucle complète de traitement des signaux par modification graphique de leur transformée de Gabor. Il utilise pour cela une partie des algorithmes proposés dans cette thèse.

Même s'il n'était pas initialement prévu qu'un produit logiciel complet soit déjà disponible et commercialisé à la fin de la thèse, nous avons pu aboutir à une

première version complète du logiciel avant ce terme. J'étais chef de projet pour le développement de LEA et j'ai participé à la conception et au développement du logiciel complet, en collaboration avec d'autres ingénieurs de la société.

Utilisation des outils proposés

Plusieurs clients ont acquis le logiciel LEA et l'utilisent régulièrement pour l'étude de la qualité sonore et le design sonore. Nous avons également eu l'occasion d'utiliser les outils proposés dans cette thèse pour réaliser des traitements effectifs pour le design sonore, à travers des contrats et des collaborations avec des industriels.

5.2 Perspectives

5.2.1 Perspectives théoriques

Couplage des approches proposées

Certains points étudiés dans cette thèse pourraient facilement être utilisés de manière conjointe pour améliorer la qualité des résultats obtenus. On peut citer les couplages suivants :

- La taille de fenêtre dans la transformée de Gabor instationnaire pourrait être choisie de manière adaptative en utilisant les critères entropiques de parcimonie. Des difficultés se poseraient cependant pour assurer la variation lente de la taille de fenêtre afin d'éviter de trop fortes déformations des fenêtres utilisées. On pourrait également avoir une ambiguïté de définition de la meilleure fenêtre, comme dans le cas de l'adaptation globale, lorsque des composantes avec des caractéristiques très différentes sont présentes à un même instant. On pourrait également coupler la méthode de détection de transitoire proposée (ou une autre) avec la transformée de Gabor instationnaire de manière à réduire automatiquement la taille de fenêtre dans les zones transitoires et obtenir de manière automatique des représentations du même type que celle de la figure 2.13 de la section 2.2.6. Cette démarche poserait moins de difficultés, mais resterait limitée à des signaux alternant des attaques marquées avec des résonances quasi-sinusoïdales avec faible variation de la fréquence instantanée.
- La méthode de détection de composantes localement harmoniques proposée en 3.1 pourrait facilement être utilisée sur une transformée de Gabor instationnaire du signal, plutôt que sur une transformée de Gabor classique, ou encore sur la composante tonale extraite à l'aide de l'algorithme de puzzle temps-fréquence. Dans les deux cas, la méthode de détection devrait être directement utilisable.

Extensions des approches proposées

L'extension des approches proposées dans cette thèse est envisageable. On peut citer en particulier :

- L'extension à d'autres représentations temps-fréquence de l'approche d'adaptation des paramètres d'analyse à partir des critères entropiques. Ceci peut se faire facilement avec la même approche en se donnant une famille paramétrée de représentations temps-fréquence sur lesquelles on peut calculer les valeurs de critères.
- La mise en place de pavages adaptatifs dans l'algorithme de puzzle temps-fréquence, notamment pour réduire la sensibilité de l'algorithme au calage temporel. Cet aspect semble cependant assez difficile à mettre en place, le problème ne semblant pas trouver de solution simple.
- L'introduction de structures dans l'algorithme de puzzle temps-fréquence par l'intermédiaire de modélisations dans le domaine des coefficients, dans le but d'améliorer la qualité de la décomposition transitoire/tonale. Pour cela, on peut s'appuyer sur les approches utilisant les chaînes de Markov décrites dans [34].

Utilisation et adaptation des décompositions proposées pour des problèmes classiques

Les décompositions proposées dans cette thèse sont des transformées non-paramétriques utilisables dans un cadre général. On peut en conséquence les appliquer dans d'autres domaines du traitement du signal. En restant dans le domaine de traitement des signaux numériques, ces décompositions peuvent être utilisées comme pré-traitement dans des problèmes classiques. Par exemple :

- L'estimation d'un modèle sinusoïdal du signal peut se faire sans modification de la méthode d'estimation sur la composante tonale obtenue avec l'algorithme de puzzle temps-fréquence. L'estimation des paramètres du modèle pourraient se faire ainsi plus facilement, le signal contenant moins de composantes perturbatrices. L'estimation du modèle pourrait également se faire sur une transformée de Gabor instationnaire, ce qui permettrait d'optimiser les paramètres d'analyse en fonction du temps.
- La décomposition obtenue avec l'algorithme de puzzle temps-fréquence pourrait faciliter certaines tâches classiques comme le débruitage, la segmentation, la séparation de source ou encore la transcription musicale.

5.2.2 Perspectives pratiques

Intégration des méthodes proposées dans LEA

Actuellement, LEA n'intègre pas l'ensemble des décompositions et algorithmes proposés dans cette thèse. L'implémentation des algorithmes non intégrés est envisagée (ou envisageable), en particulier :

- l'ajout de la transformée de Gabor instationnaire,
- l'utilisation des transformées covariantes par translation temporelle pour la mise en place de la transformée en ondelettes ou d'une analyse sur une échelle perceptive,
- l'ajout de l'algorithme de puzzle temps-fréquence, avec possibilité de représentation graphique temps-fréquence multicalque, par analogie aux logiciels de retouche d'images,
- l'ajout d'outils de détection de transitoires,
- l'ajout d'un module de resynthèse temps-réel des composantes localement harmoniques.

D'autres extensions à partir de résultats issus d'autres programmes de recherche sont également envisagées. En particulier, des outils pour le traitement de la parole vont être intégrés à court terme.

Utilisateurs de LEA

Enfin, les plus grandes perspectives pratiques sont ouvertes par les utilisateurs du logiciel LEA. J'espère qu'il leur permettra d'étudier plus facilement, plus efficacement, et plus précisément la perception de leurs sons.

Annexe A

Détails de calculs

A.1 Calcul des moments des paquets d'atomes temps-fréquence

Cette partie utilise les définitions et les notations introduites dans la partie 2.2.7.

A.1.1 Paquets construits à partir de la transformée de Gabor

Pour une fonction $g \in L^2(\mathbb{R})$, la fonction $\rho_g \in L^1(\mathbb{R})$ définie par

$$\rho_g(\nu) = \frac{|\hat{g}(\nu)|^2}{\int |\hat{g}(\nu')|^2 d\nu'} \quad (\text{A.1})$$

représente la densité d'énergie de g dans le domaine fréquentiel. Le calcul des moments de cette densité va permettre de définir les notions de fréquence centrale et de largeur de bande de la fonction g . A partir du moment d'ordre k de ρ_g , défini par

$$\langle \nu^k \rangle_g = \int \nu^k \rho_g(\nu) d\nu, \quad (\text{A.2})$$

la fréquence centrale de g est définie par $f_g = \langle \nu \rangle_g$, et la largeur de bande est définie comme l'écart type Δ_g de ρ_g vérifiant $\Delta_g^2 = \langle \nu^2 \rangle_g - \langle \nu \rangle_g^2$.

Pour le paquet Γ_n , on a :

$$\int |\hat{\Gamma}_n(\nu)|^2 d\nu = \int \int_{\nu_{n-1}}^{\nu_n} |\hat{g}(\xi - \nu)|^2 d\xi d\nu \quad (\text{A.3})$$

$$= \int_{\nu_{n-1}}^{\nu_n} \int |\hat{g}(\xi - \nu)|^2 d\nu d\xi \quad (\text{A.4})$$

$$= \int_{\nu_{n-1}}^{\nu_n} \int |\hat{g}(u)|^2 du d\xi \quad (\text{A.5})$$

$$= \int_{\nu_{n-1}}^{\nu_n} d\xi \quad (\text{A.6})$$

$$= \nu_n - \nu_{n-1}. \quad (\text{A.7})$$

On suppose que le moment d'ordre k de g , $\langle \nu^k \rangle_g$, est défini.

Pour le calcul de $\langle \nu^k \rangle_{\Gamma_n}$, on a donc :

$$\langle \nu^k \rangle_{\Gamma_n} = \frac{1}{\nu_n - \nu_{n-1}} \int \nu^k \int_{\nu_{n-1}}^{\nu_n} |\hat{g}(\xi - \nu)|^2 d\xi d\nu \quad (\text{A.8})$$

$$= \frac{1}{\nu_n - \nu_{n-1}} \int \nu^k \int_{\nu_{n-1}-\nu}^{\nu_n-\nu} |\hat{g}(u)|^2 du d\nu \quad (\text{A.9})$$

après intégration par partie par rapport à la variable ν , on a :

$$\langle \nu^k \rangle_{\Gamma_n} = \frac{1}{\nu_n - \nu_{n-1}} \left[\frac{\nu^{k+1}}{k+1} \int_{\nu_{n-1}-\nu}^{\nu_n-\nu} |\hat{g}(u)|^2 du \right]_{-\infty}^{-\infty} \quad (\text{A.10})$$

$$- \frac{1}{\nu_n - \nu_{n-1}} \int \frac{\nu^{k+1}}{k+1} (|\hat{g}(\nu_n - \nu)|^2 + |\hat{g}(\nu_{n-1} - \nu)|^2) d\nu \quad (\text{A.11})$$

Or, on peut montrer, que si le moment d'ordre k de g est défini, alors le terme $\left[\frac{\nu^{k+1}}{k+1} \int_{\nu_{n-1}-\nu}^{\nu_n-\nu} |\hat{g}(u)|^2 du \right]_{-\infty}^{-\infty}$ est nul. Il reste donc :

$$\langle \nu^k \rangle_{\Gamma_n} = - \frac{1}{\nu_n - \nu_{n-1}} \int \frac{\nu^{k+1}}{k+1} (|\hat{g}(\nu_n - \nu)|^2 + |\hat{g}(\nu_{n-1} - \nu)|^2) d\nu \quad (\text{A.12})$$

$$= \frac{1}{\nu_n - \nu_{n-1}} \int \frac{(\nu_n - u)^{k+1} - (\nu_{n-1} - u)^{k+1}}{k+1} |\hat{g}(u)|^2 du \quad (\text{A.13})$$

$$= \frac{1}{k+1} \int \frac{(\nu_n - u)^{k+1} - (\nu_{n-1} - u)^{k+1}}{(\nu_n - u) - (\nu_{n-1} - u)} |\hat{g}(u)|^2 du \quad (\text{A.14})$$

$$= \frac{1}{k+1} \int \left(\sum_{l=0}^k (\nu_n - u)^l (\nu_{n-1} - u)^{k-l} \right) |\hat{g}(u)|^2 du \quad (\text{A.15})$$

$$= \frac{1}{k+1} \sum_{l=0}^k \langle (\nu_n - \nu)^l (\nu_{n-1} - \nu)^{k-l} \rangle_g \quad (\text{A.16})$$

Pour les deux premiers moments du paquet Γ_n , on a donc :

$$\langle \nu \rangle_{\Gamma_n} = \frac{1}{2} (\langle \nu_{n-1} - \nu \rangle_g + \langle \nu_n - \nu \rangle_g) \quad (\text{A.17})$$

$$= \frac{\nu_{n-1} + \nu_n}{2} - \langle \nu \rangle_g \quad (\text{A.18})$$

et

$$\langle \nu^2 \rangle_{\Gamma_n} = \frac{1}{3} (\langle (\nu_{n-1} - \nu)^2 \rangle_g + \langle (\nu_n - \nu)(\nu_{n-1} - \nu) \rangle_g + \langle (\nu_n - \nu)^2 \rangle_g) \quad (\text{A.19})$$

$$= \frac{1}{3} (\nu_{n-1}^2 + \nu_{n-1}\nu_n + \nu_n^2) - (\nu_{n-1} + \nu_n)\langle \nu \rangle_g + \langle \nu^2 \rangle_g. \quad (\text{A.20})$$

Pour une fenêtre g réelle, ρ_g et symétrique et on a $\langle \nu \rangle_g = 0$, on obtient alors sans surprise pour le paquet Γ_n la fréquence centrale :

$$f_{\Gamma_n} = \frac{\nu_{n-1} + \nu_n}{2} \quad (\text{A.21})$$

et la largeur de bande Δ_{Γ_n} vérifie :

$$\Delta_{\Gamma_n}^2 = \langle \nu^2 \rangle_{\Gamma_n} - \langle \nu \rangle_{\Gamma_n}^2 \quad (\text{A.22})$$

$$= \frac{1}{3} (\nu_{n-1}^2 + \nu_{n-1}\nu_n + \nu_n^2) + \langle \nu^2 \rangle_g + \left(\frac{\nu_{n-1} + \nu_n}{2} \right)^2 \quad (\text{A.23})$$

$$= \Delta_g^2 + \frac{(\nu_n - \nu_{n-1})^2}{12}. \quad (\text{A.24})$$

A.1.2 Paquets construits à partir de la transformée en ondelettes

On a :

$$\int |\hat{\Psi}_n(\nu)|^2 d\nu = \int \int_{a_{n-1}}^{a_n} |\hat{h}(a\nu)|^2 \frac{da}{a} d\nu \quad (\text{A.25})$$

$$= \int_{a_{n-1}}^{a_n} \left(\int |\hat{h}(u)|^2 du \right) \frac{da}{a^2} \quad (\text{A.26})$$

$$= \|\hat{h}\|^2 \left(\frac{1}{a_{n-1}} - \frac{1}{a_n} \right) \quad (\text{A.27})$$

$$= \|\hat{h}\|^2 (\mu_{n-1} - \mu_n), \quad (\text{A.28})$$

en posant $\mu_n = \frac{1}{a_n}$.

D'où pour le calcul du moment d'ordre k

$$\langle \nu^k \rangle_{\Psi_n} = \frac{1}{\|h\|^2(\mu_{n-1} - \mu_n)} \int \nu^k \int_{a_{n-1}}^{a_n} |\hat{h}(a\nu)|^2 \frac{da}{a} d\nu \quad (\text{A.29})$$

$$= \frac{1}{\|h\|^2(\mu_{n-1} - \mu_n)} \int_{a_{n-1}}^{a_n} \int u^k |\hat{h}(u)|^2 du \frac{da}{a^{k+2}} \quad (\text{A.30})$$

$$= \frac{1}{\|h\|^2(\mu_{n-1} - \mu_n)} \|h\|^2 \langle \nu^k \rangle_h \int_{a_{n-1}}^{a_n} \frac{da}{a^{k+2}} \quad (\text{A.31})$$

$$= \frac{1}{\mu_{n-1} - \mu_n} \langle \nu^k \rangle_h \frac{1}{k+1} \left(\frac{1}{a_{n-1}^{k+1}} - \frac{1}{a_n^{k+1}} \right) \quad (\text{A.32})$$

$$= \frac{1}{k+1} \frac{\mu_{n-1}^{k+1} - \mu_n^{k+1}}{\mu_{n-1} - \mu_n} \langle \nu^k \rangle_h \quad (\text{A.33})$$

$$= \frac{1}{k+1} \left(\sum_{l=0}^k \mu_n^l \mu_{n-1}^{k-l} \right) \langle \nu^k \rangle_h \quad (\text{A.34})$$

La fréquence centrale du paquet vaut donc

$$f_{\Psi_n} = \frac{\mu_n + \mu_{n-1}}{2} f_h, \quad (\text{A.35})$$

et largeur de bande vérifie

$$\Delta_{\Psi_n}^2 = \langle \nu^2 \rangle_{\Psi_n} - \langle \nu \rangle_{\Psi_n}^2 \quad (\text{A.36})$$

$$= \frac{\mu_n^2 + \mu_n \mu_{n-1} + \mu_{n-1}^2}{3} \langle \nu^2 \rangle_h - \left(\frac{\mu_n + \mu_{n-1}}{2} \right)^2 f_h^2 \quad (\text{A.37})$$

$$= \frac{\mu_n^2 + \mu_n \mu_{n-1} + \mu_{n-1}^2}{3} (\Delta_h^2 + f_h^2) - \left(\frac{\mu_n + \mu_{n-1}}{2} \right)^2 f_h^2 \quad (\text{A.38})$$

$$= \frac{\mu_n^2 + \mu_n \mu_{n-1} + \mu_{n-1}^2}{3} \Delta_h^2 + \frac{(\mu_n - \mu_{n-1})^2}{12} f_h^2 \quad (\text{A.39})$$

A.2 Calcul des entropies pour un et deux atomes gaussiens

A.2.1 Pour un atome gaussien

On s'intéresse à un signal x constitué d'un atome temps-fréquence gaussien d'échelle σ_0 et de fréquence ν_0 de la forme suivante :

$$x(t) = e^{i2\pi\nu_0 t} e^{-\frac{t^2}{2\sigma_0^2}}, \quad (\text{A.40})$$

A.2. CALCUL DES ENTROPIES POUR UN ET DEUX ATOMES GAUSSIENS 153

et on réalise la transformée de Gabor en utilisant une fenêtre gaussienne g d'échelle σ normalisée :

$$g(t) = \frac{1}{\sqrt{\sigma}\sqrt{\pi}} e^{-\frac{t^2}{2\sigma^2}}. \quad (\text{A.41})$$

On a :

$$G_x^g(\tau, \nu) = \frac{1}{\sqrt{\sigma}\sqrt{\pi}} \int e^{i2\pi\nu t} e^{-\frac{t^2}{2\sigma_0^2}} e^{-\frac{(t-\tau)^2}{2\sigma^2}} e^{-i2\pi\nu(t-\tau)} dt \quad (\text{A.42})$$

$$= \frac{1}{\sqrt{\sigma}\sqrt{\pi}} \int e^{-\frac{1}{2}\left(\frac{1}{\sigma_0^2} + \frac{1}{\sigma^2}\right)t^2} e^{i(2\pi(\nu_0-\nu) - i\frac{\tau}{\sigma^2})t} e^{-\frac{\tau^2}{2\sigma^2} + i2\pi\nu\tau} dt, \quad (\text{A.43})$$

en utilisant la relation connue

$$\forall a \in \mathbb{R}_+^*, \forall b \in \mathbb{C}, \quad \int e^{-\frac{t^2}{2a^2}} e^{ibt} dt = a\sqrt{2\pi} e^{-\frac{a^2 b^2}{2}}, \quad (\text{A.44})$$

et en posant $\lambda = \frac{\sigma\sigma_0}{\sqrt{\sigma^2 + \sigma_0^2}}$, il vient :

$$G_x^g(\tau, \nu) = \frac{\lambda\sqrt{2\pi}}{\sqrt{\sigma}\sqrt{\pi}} e^{-\frac{\tau^2}{2\sigma^2} + i2\pi\nu\tau} e^{-\frac{\lambda^2}{2}(2\pi(\nu_0-\nu) - i\frac{\tau}{\sigma^2})^2} \quad (\text{A.45})$$

$$= \frac{\lambda\sqrt{2\pi}}{\sqrt{\sigma}\sqrt{\pi}} e^{-\frac{\tau^2}{2(\sigma^2 + \sigma_0^2)}} e^{-2\pi^2\lambda^2(\nu_0-\nu)^2} e^{i2\pi\tau\left(\nu + \frac{\lambda^2}{\sigma^2}(\nu_0-\nu)\right)}, \quad (\text{A.46})$$

de plus,

$$\|x\|^2 = \int |x(t)|^2 dt = \int e^{-\frac{t^2}{\sigma_0^2}} dt = \sigma_0\sqrt{\pi}. \quad (\text{A.47})$$

donc

$$\rho_x^g(\tau, \nu) = \frac{|G_x^g(\tau, \nu)|^2}{\|x\|^2} \quad (\text{A.48})$$

$$= \frac{1}{\sigma_0\sqrt{\pi}} \frac{2\pi\lambda^2}{\sigma\sqrt{\pi}} e^{-\frac{\tau^2}{\sigma^2 + \sigma_0^2}} e^{-4\pi^2\lambda^2(\nu_0-\nu)^2} \quad (\text{A.49})$$

$$= \frac{2\lambda^2}{\sigma\sigma_0} e^{-\frac{\tau^2}{\sigma^2 + \sigma_0^2}} e^{-4\pi^2\lambda^2(\nu_0-\nu)^2}. \quad (\text{A.50})$$

L'entropie de Rényi pour un atome gaussien vaut donc :

$$R_\alpha(g) = \frac{1}{1-\alpha} \log_2 \left(\iint (\rho_x^g(\tau, \nu))^\alpha d\tau d\nu \right) \quad (\text{A.51})$$

$$= \frac{1}{1-\alpha} \log_2 \left(\left(\frac{2\lambda^2}{\sigma\sigma_0} \right)^\alpha \int e^{-\frac{\alpha\tau^2}{\sigma^2+\sigma_0^2}} d\tau \int e^{-\alpha 4\pi^2\lambda^2(\nu_0-\nu)^2} d\nu \right) \quad (\text{A.52})$$

$$= \frac{1}{1-\alpha} \log_2 \left(\left(\frac{2\lambda^2}{\sigma\sigma_0} \right)^\alpha \sqrt{\frac{\pi(\sigma^2+\sigma_0^2)}{\alpha}} \frac{\sqrt{\pi}}{2\pi\lambda\sqrt{\alpha}} \right) \quad (\text{A.53})$$

$$= \frac{1}{1-\alpha} \log_2 \left(\frac{2^{\alpha-1}}{\alpha} \left(\frac{\sigma\sigma_0}{\sigma^2+\sigma_0^2} \right)^\alpha \sqrt{\sigma^2+\sigma_0^2} \frac{\sqrt{\sigma^2+\sigma_0^2}}{\sigma\sigma_0} \right) \quad (\text{A.54})$$

$$= \frac{1}{1-\alpha} \log_2 \left(\frac{2^{\alpha-1}}{\alpha} \left(\frac{\sigma}{\sigma_0} + \frac{\sigma_0}{\sigma} \right)^{1-\alpha} \right) \quad (\text{A.55})$$

$$= -1 - \frac{\log_2(\alpha)}{1-\alpha} + \log_2 \left(\frac{\sigma}{\sigma_0} + \frac{\sigma_0}{\sigma} \right). \quad (\text{A.56})$$

Et l'entropie de Shannon correspond à la limite pour α tendant vers 1 :

$$S(g) = \frac{1}{\ln(2)} - 1 + \log_2 \left(\frac{\sigma}{\sigma_0} + \frac{\sigma_0}{\sigma} \right) \quad (\text{A.57})$$

A.2.2 Pour deux atomes gaussiens éloignés

On s'intéresse maintenant au cas où le signal x est composé de la somme de deux atomes gaussiens x_1 et x_2 d'échelles σ_1 et σ_2 et de fréquences ν_1 et ν_2 , **que nous supposons très éloignés dans le plan temps-fréquence** pour simplifier les calculs :

$$x(t) = x_1(t) + x_2(t) = e^{i2\pi\nu_1 t} e^{-\frac{t^2}{2\sigma_1^2}} + e^{i2\pi\nu_2 t} e^{-\frac{t^2}{2\sigma_2^2}}. \quad (\text{A.58})$$

On conserve la fenêtre gaussienne définie précédemment.

Comme les deux atomes sont très éloignés dans le plan temps-fréquence, les termes d'interférences entre les deux atomes x_1 et x_2 sont très petits dans le spectrogramme de x , on peut alors faire l'approximation :

$$|G_x^g|^2 = |G_{x_1}^g + G_{x_2}^g|^2 \approx |G_{x_1}^g|^2 + |G_{x_2}^g|^2, \quad (\text{A.59})$$

qui implique l'approximation :

$$\rho_x^g(\tau, \nu) \approx \frac{\sigma_1}{\sigma_1 + \sigma_2} \rho_{x_1}^g(\tau, \nu) + \frac{\sigma_2}{\sigma_1 + \sigma_2} \rho_{x_2}^g(\tau, \nu). \quad (\text{A.60})$$

A.2. CALCUL DES ENTROPIES POUR UN ET DEUX ATOMES GAUSSIENS 155

L'approximation de la valeur du critère dans ces conditions peut alors être déduite,

$$\begin{aligned} R_\alpha(g) &= \frac{1}{1-\alpha} \log_2 \left(\iint (\rho_x^g(\tau, \nu))^\alpha d\tau d\nu \right) \\ &\approx \frac{1}{1-\alpha} \log_2 \left(\iint \left(\frac{\sigma_1}{\sigma_1 + \sigma_2} \rho_{x_1}^g(\tau, \nu) + \frac{\sigma_2}{\sigma_1 + \sigma_2} \rho_{x_2}^g(\tau, \nu) \right)^\alpha d\tau d\nu \right) \end{aligned}$$

compte tenu de l'éloignement des deux atomes, on peut encore une fois ignorer les termes croisés et approximativement effectuer ce calcul comme si les supports temps-fréquence des deux atomes étaient disjoints et on a :

$$\begin{aligned} R_\alpha(g) &\approx \frac{1}{1-\alpha} \log_2 \left(\iint \left(\frac{\sigma_1}{\sigma_1 + \sigma_2} \rho_{x_1}^g(\tau, \nu) \right)^\alpha + \left(\frac{\sigma_2}{\sigma_1 + \sigma_2} \rho_{x_2}^g(\tau, \nu) \right)^\alpha d\tau d\nu \right) \\ &\approx -1 - \frac{\log_2(\alpha)}{1-\alpha} - \frac{\alpha}{1-\alpha} \log_2(\sigma_1 + \sigma_2) \\ &\quad + \frac{1}{1-\alpha} \log_2 \left(\sigma_1^\alpha \left(\frac{\sigma}{\sigma_1} + \frac{\sigma_1}{\sigma} \right)^{1-\alpha} + \sigma_2^\alpha \left(\frac{\sigma}{\sigma_2} + \frac{\sigma_2}{\sigma} \right)^{1-\alpha} \right). \quad (\text{A.61}) \end{aligned}$$

Bibliographie

- [1] Richard G. Baraniuk, Patrick Flandrin, Augustus J. E. M. Janssen, and Olivier J. J. Michel. Measuring time-frequency information content using the Rényi entropies. *IEEE Trans. Inform. Theory*, 47(4) :1391–1409, 2001.
- [2] J.-P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, 2005. to appear.
- [3] R. Carmona, W.L. Hwang, and B. Torrèsani. *Practical Time-Frequency Analysis : continuous wavelet and Gabor transforms, with an implementation in S*, volume 9 of *Wavelet Analysis and its Applications*. Academic Press, San Diego, 1998.
- [4] S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1) :33–61, 1998.
- [5] L. Cohen. *Time-Frequency Analysis*. Prentice Hall, New York, 1994.
- [6] Christophe d’Alessandro and Denis Beutemps. Transformation en ondelettes sur une échelle fréquentielle auditive. In *Actes du XIIIème colloque GRETSI*, pages 745–748, septembre 1991.
- [7] I. Daubechies. *Ten lectures on wavelets*. SIAM, Philadelphia, PA, 1992.
- [8] Ingrid Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Inform. Theory*, 36(5) :961–1005, 1990.
- [9] G. Davis, S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Constructive Approximation*, 13 :57–98, 1997.
- [10] M. Davy and S.J.G. Godsill. Bayesian harmonic models for musical signal analysis (with discussion). In J.M. Bernardo, J.O. Berger, A.P. Dawid, and A.F.M. Smith, editors, *Bayesian Statistics VII*. Oxford University Press, 2003.
- [11] P. Depalle, G. Garcia, and X. Rodet. Tracking of partials for additive sound synthesis using hidden markov model. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 225–228, 1993.

- [12] M. Dörfler. *Gabor Analysis for a Class of Signals called Music*. PhD thesis, NuHAG, University of Vienna, 2002.
- [13] Monika Dörfler. Time–frequency analysis for music signals – a mathematical approach. *Journal of New Music Research*, 30(1), March 2001.
- [14] M. Duval-Destin, M.A. Muschietti, and B. Torrèsani. Continuous wavelet decompositions, multiresolution and contrast analysis. *SIAM J. of Math. Anal.*, 24(3) :739–755, May 1993.
- [15] Hans G. Feichtinger and K. H. Gröchenig. Banach spaces related to integrable group representations and their atomic decompositions. I. *J. Funct. Anal.*, 86(2) :307–340, 1989.
- [16] Hans G. Feichtinger and K. H. Gröchenig. Banach spaces related to integrable group representations and their atomic decompositions. II. *Monatsh. Math.*, 108(2-3) :129–148, 1989.
- [17] Hans G. Feichtinger and Krzysztof Nowak. A first survey of Gabor multipliers. In *Advances in Gabor analysis*, Appl. Numer. Harmon. Anal., pages 99–128. Birkhäuser Boston, Boston, MA, 2003.
- [18] Hans G. Feichtinger and Thomas Strohmer, editors. *Gabor analysis and algorithms*. Applied and Numerical Harmonic Analysis. Birkhäuser Boston Inc., Boston, MA, 1998. Theory and applications.
- [19] Hans G. Feichtinger and Thomas Strohmer, editors. *Gabor analysis and algorithms*. Applied and Numerical Harmonic Analysis. Birkhäuser Boston Inc., Boston, MA, 2003.
- [20] P. Flandrin. *Temps-Fréquence*. Traité des Nouvelles Technologies, série Traitement du Signal. Hermès, Paris, 1993.
- [21] D. Gabor. Theory of communication. *J. IEE (London)*, 93(III) :429–457, November 1946.
- [22] R. Gribonval. *Approximations non-linéaires pour l’analyse des signaux sonores*. PhD thesis, Université de Paris IX Dauphine, 1999.
- [23] Karlheinz Gröchenig. *Foundations of time-frequency analysis*. Applied and Numerical Harmonic Analysis. Birkhäuser Boston Inc., Boston, MA, 2001.
- [24] Florent Jaillet. Représentation et retouche de sons dans le plan temps-fréquence. Master’s thesis, DEA ATIAM, Université de la Méditerranée Aix/Marseille II, 2001.
- [25] Florent Jaillet. Projet de recherche représentation et traitement temps-fréquence des signaux audionumériques pour des applications de design sonore - annexe technique. Technical report, GENESIS, 2005.

- [26] Florent Jaillet and Bruno Torr sani. Sculpture temps-fr quence des sons par modification de leur transform e en ondelettes. In *Actes du 6 me Congr s Franais d'Acoustique*, 2002.
- [27] Denis Jolivet. Design sonore industriel - application aux d marreurs automobiles. Master's thesis, DEA d'Acoustique, Universit  de la M diterran e Aix/Marseille II, 2004.
- [28] N Kaiblinger. Approximation of the fourier transform and the dual gabor window. *J. Fourier Anal. Appl.*, 11(1) :25–42, 2005.
- [29] W. Kozek. *Matched Weyl–Heisenberg Expansions of Nonstationary Environments*. PhD thesis, NuHAG, University of Vienna, 1996.
- [30] P. Leveau, L. Daudet, and G. Richard. Methodology and tools for the evaluation of automatic onset detection algorithms in music. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR 04)*, 2004.
- [31] S. Mallat. *A wavelet tour of signal processing*. Academic Press, 1998.
- [32] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41 :3397–3415, 1993.
- [33] R.J. McAulay and Th.F. Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Trans. on Acoust., Speech and Signal Proc.*, 34 :744–754, 1986.
- [34] St phane Molla. *Signaux audiophoniques : mod lisation hybride et sch ma de codage*. PhD thesis, Universit  de Provence Aix-Marseille I, 2003.
- [35] J.A. Moorer. The use of the phase vocoder in computer music applications. *Journal of Audio Engineering Society*, 26(1/2) :42–45, 1978.
- [36] M.A. Muschietti and B. Torr sani. Pyramidal algorithms for littlewood-paley decompositions. *SIAM J. Math. Anal.*, 26 :925–943, 1995.
- [37] Geoffroy Peeters. *Mod les et modification du signal sonore adapt s   ces caract ristiques locales*. PhD thesis, Universit  Paris 6, 2001.
- [38] Florent Poirot. Structure logicielle pour un  diteur temps-fr quence des sons. Master's thesis, Projet de fin d' tude,  cole Nationale Sup rieure des T l communications de Bretagne, 2003.
- [39] M.R. Portnoff. Implementation of the digital phase vocoder using the fast fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24(3) :243–248, 1976.
- [40] X. Rodet. Musical sound signal analysis/synthesis : Sinusoidal+residual and elementary waveform models. In *Proceedings of the IEEE Time-Frequency and Time-Scale Workshop 97*, Coventry, UK, 1997.

- [41] X. Rodet and P. Depalle. Spectral envelopes and inverse FFT synthesis. In *Proceedings of 1992 AES Convention*, 1992.
- [42] X. Serra. *A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition*. PhD thesis, Stanford University, 1989.
- [43] T. Virtanen. Audio signal modeling with sinusoids plus noise. Master's thesis, Tampere University of Technology, Department of Information Technology, 2000.
- [44] M. V. Wickerhauser. *Adapted Wavelet Analysis from Theory to Software*. AK Peters, Boston, MA, USA, 1994.
- [45] G. Winkler. *Image analysis, random fields and dynamic Monte Carlo methods : a mathematical introduction*. Springer-Verlag, 1995.
- [46] G. Winkler and V. Liebscher. Smoothers for discontinuous signals. *J. Nonpar. Statist.*, 14(1-2) :203–222, 2002.
- [47] M. Wright, J. Beauchamp, K. Fitz, X. Rodet, A. Röbel, and G. Wakefield. Analysis/synthesis comparison. *Organised Sound*, 5(3) :173–189, 2000.
- [48] M. Zibulski and Y.Y. Zeevi. Analysis of multi-window Gabor-type schemes by frame methods. *Applied and Computational Harmonic Analysis*, 4(2) :188–221, 1997.
- [49] Eberhard Zwicker and Richard Feldtkeller. *Psychoacoustique - L'oreille, récepteur d'information*. Masson, 1981.

Représentation et traitement temps-fréquence des signaux audio numériques pour des applications de design sonore

Florent JAILLET

Résumé

Cette thèse décrit un certain nombre de développements théoriques, algorithmiques et logiciels en vue d'applications au design sonore industriel. Le design sonore met en oeuvre une série de traitements effectués sur des signaux audio numériques, allant de la représentation à la synthèse, en passant par des étapes d'analyse et de modification. Ce mémoire présente des contributions aux différentes étapes de traitement.

La première partie est consacrée au problème de représentation, en se limitant au cadre des représentations temps-fréquence "atomiques" (transformation de Gabor, ondelettes, ...). On montre comment ces approches peuvent être modifiées pour y introduire davantage de souplesse, ce qui permet de les adapter à un signal, une classe de signaux, ou une problématique donnée. On montre également comment ces techniques peuvent être mises en oeuvre dans le cadre d'algorithmes adaptatifs, supervisés ou non supervisés.

La deuxième partie est consacrée à la détection et l'estimation de composantes structurées dans les signaux, à partir de représentations adaptées telles que celles développées dans la première partie. On considère en particulier le cas de signaux localement harmoniques, et celui de signaux transitoires.

La troisième partie décrit les grandes lignes d'un développement logiciel effectué dans le cadre de ce travail, et mettant en oeuvre des approches décrites dans les deux premières parties du mémoire. Elle présente également des exemples de traitements réalisés dans un cadre industriel.

Mots clefs : représentation temps-fréquence, transformée de Gabor, transformée en ondelettes, traitement du son, décomposition adaptative, détection d'harmoniques, détection de transitoires

Abstract

Title : Time-frequency representation and processing of audio signal for sound design applications

This thesis describes a number of theoretical issues, algorithms and software developments in view of industrial sound design applications. Audio processing for sound design involves a set of processing steps, including signal analysis, representation, modification, and synthesis. This document describes contributions to these different steps.

The first part deals with time-frequency representations, focusing on the case of atomic decompositions (Gabor transform, wavelets, ...). We show how these approaches can be modified to bring more freedom in the decomposition, enabling to adapt the decomposition to given signal or class of signals, or a specific problem. We also show how these elements can be used in the context of supervised and unsupervised adaptive algorithms.

The second part deals with detection and estimation of structured components, starting from an adapted representation such as the ones proposed in the first part. We study the cases of locally harmonic components and transient components.

The third part describes the main aspects of software tools developed as part of this work. This software implements several of the new approaches described in the first two parts. It also shortly describes a few examples of processing for sound design done in an industrial context using the tools described in this thesis.

Keywords : time-frequency representation, Gabor analysis, wavelet analysis, sound processing, adaptive decomposition, harmonic detection, transient detection

Spécialité

Acoustique, Traitement du signal et Informatique Appliqués à la Musique

Laboratoire d'accueil

Laboratoire d'Analyse, Topologie, Probabilités
UMR 6632
Centre de Mathématiques et Informatique
Université de Provence
39 rue F. Joliot-Curie
13453 Marseille cedex 13