Tatouage haute-capacité pour mélanges sonores

Jonathan Pinel

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Électronique, Électrotechnique, Automatique et Traitement du Signal**

Arrêté ministériel : 7 août 2006

Présentée par

Jonathan PINEL

Thèse dirigée par **Laurent GIRIN**
et co-encadrée par **Cléo BARAS**

préparée au sein **Laboratoire GIPSA-Lab**
et de **l'École Doctorale EEATS (ED 220)**

Tatouage haute-capacité pour mélanges sonores

Soutenance prévue le **Mardi 23 Juillet 2013**,
devant le jury composé de :

Mme. Régine LE BOUQUIN-JEANNÈS
Professeur à l'Université de Rennes 1, Rapporteur

Mme. Madeleine BONNET
Professeur à l'Université Paris Descartes, Rapporteur (non membre du jury de soutenance)

M. Gaël MAHÉ
Maître de Conférences à l'Université Paris Descartes, Examinateur

M. Laurent DAUDET
Professeur à l'Université Paris Diderot, Examinateur

M. Gwenaël DOËRR
Ingénieur de Recherche à Technicolor France, Examinateur

M. Laurent GIRIN
Professeur à Grenoble-INP, Directeur de thèse

Mme. Cléo BARAS
Maître de Conférences à l'Université Joseph Fourier, Co-Encadrant de thèse

Remerciements

Je tiens à dire un grand merci à mes deux encadrants de thèse, pour avoir toujours suivi mes travaux avec attention et m'avoir soutenu même dans les moments difficiles. Je remercie aussi le Gipsa-lab pour son accueil, particulièrement le DIS et mes co-bureaux, pour leur bonne humeur et leur gentillesse. Remerciements particuliers aussi à toute l'équipe DReaM, qui a rendu toutes les réunions de projets extrêmement enrichissantes et sympathiques.

Et pour garder le meilleur pour la fin, un grand merci à mes amis et ma famille (que je ne citerai pas pour éviter les longues listes), qui m'ont soutenu tout au long de cette aventure.

Table des matières

Table des figures

Liste des tableaux

Introduction

C'est en Italie à la fin du 13ème siècle qu'apparaissent les premiers filigranes (*watermarks* en anglais). Ils sont à cette période obtenus en pressant des fils métalliques sur les trames lors de la création du papier, ce qui le rend plus mince à ces endroits et permet de voir la forme des fils par transparence. Bien que leur but à l'origine reste incertain, les filigranes ont rapidement été utilisés pour transmettre des informations telles que la provenance et la qualité du papier. Les techniques de marquage par filigranes ont été grandement améliorées à partir du milieu du 20^{e} siècle notamment pour protéger les billets de banque des contrefaçons. Une caractéristique commune de ces filigranes est qu'ils sont imperceptibles sauf lorsqu'ils sont observés à la lumière. C'est cette notion de marquage imperceptible en temps normal mais que l'on peut décoder en suivant un certain procédé qui a conduit à l'emprunt du nom anglais des filigranes, *watermarks*, pour désigner le tatouage numérique (*digital watermarking*). Cette discipline est donc l'application du principe du tatouage aux signaux numériques : insérer une information (ou marque) dans un signal numérique (appelé l'hôte) en le modifiant d'une manière qui soit imperceptible lors d'une utilisation normale (inaudibilité pour un hôte audio, invisibilité pour un hôte image ou vidéo…), tout en rendant cette marque récupérable au moyen d'un processus particulier d'extraction.

Le tatouage est donc né d'une volonté sécuritaire, afin d'assurer l'authenticité ou la propriété de certaines données multimédia. De nombreuses recherches ont été faites, dans lesquelles on assiste souvent au jeu du chat et de la souris entre les auteurs de nouvelles techniques de tatouage (plus robustes, plus sûres) et des contrevenants cherchant à trouver des méthodes systématiques pour les modifier ou les supprimer. Dans cet état d'esprit, deux autres disciplines se lient plus ou moins fortement au tatouage, alors que leur origine est pourtant différente : la stéganographie et la cryptographie. La stéganographie consiste à transmettre une information à travers un média quelconque, sans qu'une tierce partie ne s'aperçoive qu'un message est inséré. Dans le cas du tatouage, l'existence de la marque est supposée connue, l'objectif étant alors d'empêcher la tierce personne de décoder ou supprimer le message. La cryptographie consiste elle à modifier la représentation d'une information afin qu'elle ne soit pas utilisable en l'état par une tierce personne. L'information cryptée n'est donc pas interprétable en tant que média, et ce aussi longtemps qu'elle n'a pas été décryptée. Le tatouage étant en quelque sorte à la jonction de ces deux disciplines, des interactions et inspirations d'un domaine vers l'autre apparaissent fréquemment dans la littérature.

Il existe également un autre thème de recherche dérivé du tatouage : le tatouage pour

la transmission de données. Dans ce cadre, le média est considéré comme un canal de transmission, où l'on cherche à faire transiter (ou stocker) de l'information par tatouage, en respectant toujours la contrainte d'imperceptibilité. Les utilisations possibles du tatouage pour la transmission de données sont nombreuses, et s'écartent de l'aspect sécuritaire du tatouage numérique classique. C'est dans ce contexte que se situent les travaux de thèse présentés dans ce document, et plus spécifiquement dans le cadre du projet DReaM. L'objectif de cette thèse est de développer un système de tatouage pour la transmission de données à haut débit [1] appliqué à des signaux audio au format non compressé. Ce système doit satisfaire les besoins du projet DReaM, qui concerne la séparation de sources informée de signaux musicaux (c.f. chapitre 1) mais doit aussi pouvoir être utilisé pour d'autres applications audio, typiquement des applications visant à enrichir du contenu multimedia.

La première partie de ce manuscrit s'attache à présenter le contexte applicatif et scientifique de notre étude. Nous présentons tout d'abord le projet DReaM, ses objectifs principaux ainsi que la manière dont les travaux de thèse présentés dans cet ouvrage s'articulent en son sein. Cette vue d'ensemble du projet DReaM nous permet ainsi d'entrevoir les contraintes qui vont peser sur le système de tatouage et qui sont développées plus tard dans le manuscrit. Ensuite, nous présentons le contexte scientifique qui consiste en deux grandes parties, l'une sur le tatouage et l'autre sur le codage audio perceptuel. Le chapitre sur le tatouage présente les principes fondamentaux ainsi que les techniques usuelles de ce domaine, qui sont au cœur de presque tous les systèmes de tatouage existants. Le chapitre sur le codage audio, plus condensé, présente les grands principes de ce domaine, dont nous nous inspirons pour notre système de tatouage.

La seconde partie décrit une première implémentation basique de notre système de tatouage. Dans cette partie nous nous attachons d'abord à décrire de façon précise toutes les contraintes de notre système de tatouage, qui découlent de son utilisation dans le cadre du projet DReaM. Nous expliquons ensuite comment est constitué grossièrement le système de tatouage que nous proposons, en nous inspirant des codeurs audio perceptuels. Ensuite, nous décrivons cette première implémentation en détail en précisant les choix faits afin de respecter le cahier des charges fixé auparavant. Finalement nous terminons cette partie par une série d'expériences visant à montrer le bon fonctionnement du système présenté, et nous étudions ses performances par rapport au cahier des charges établi.

Dans la troisième partie du manuscrit, nous présentons une implémentation améliorée de l'implémentation basique. Plus précisément nous nous penchons tout particulièrement sur les points faibles de la première implémentation mis en évidence lors des expériences de la partie précédente, et nous nous attachons à présenter des solutions pour pallier ces différents problèmes. Nous terminons cette partie par une nouvelle série d'expériences cherchant à montrer les améliorations par rapport au système précédent, mais aussi par rapport à certains systèmes de la littérature dont les performances sont similaires au nôtre.

La quatrième partie s'écarte un peu de la ligne directrice du reste du manuscrit, en ce sens

1. Comme on le verra plus en détail par la suite, dans le cadre des recherches présentées dans ce manuscrit on s'intéresse à des débits de quelques centaines de kbits/s par canal (200 voire plus) pour des signaux audio non compressés, ce qui représente 30% (voire plus) du débit du signal hôte

qu'elle ne traite pas directement de tatouage : dans cette partie nous présentons un système conçu pour rendre parcimonieux la représentation temps-fréquence d'un signal audio. Cette problématique désormais classique de représentations parcimonieuses de signaux est en effet particulièrement intéressante dans le cadre du projet DReaM, et ce pour de nombreuses raisons que nous présenterons. Bien que n'étant pas du tatouage à proprement parlé, le pan de ce travail de thèse reste cependant cohérent avec les travaux des parties précédentes. En effet, nous verrons que le système de « parcimonisation » du signal est aussi issu des considérations sur les liens entre insertion / suppression d'informations dans le signal et codage perceptuel qui sont exploités dans les systèmes de tatouage des parties précédentes. Nous terminons cette partie par une présentation des résultats obtenus en terme de représentation parcimonieuse et un exemple d'application dans le cadre du projet DReaM.

Première partie

Contexte applicatif et scientifique

Chapitre 1

Le projet DReaM

Le projet DReaM (Disque Repensé pour l'écoute active de la Musique) est un projet ANR labellisé en octobre 2009 dans le cadre de l'appel d'offre CONTINT (CONTenus numériques et INTeractions), qui regroupe pour partenaires : l'institut Langevin Ondes et Images [1], Télécom ParisTech [2], le LaBRI [3], le GIPSA-lab [4], et l'industriel iKlax Media [5].

1.1 Objectifs du projet

L'objectif de ce projet est de développer de nouvelles technologies permettant à un auditeur écoutant un morceau de musique stéréophonique (à deux voies) d'interagir directement avec le morceau au niveau de ses macro-composantes musicales, par exemple en ayant la possibilité de modifier le volume d'un instrument [6] particulier du mélange, ou bien de modifier sa position dans l'espace, son timbre, voire de changer la hauteur de note ou lui ajouter un effet. Deux applications particulières de l'écoute active sont notables :

- Le karaoké généralisé, c'est-à-dire la suppression d'un ou plusieurs des instruments composant le signal de musique. Au delà des applications ludiques du karaoké, on peut par exemple concevoir une utilisation pédagogique dans le cadre de l'apprentissage de la musique. Les membres d'un groupe musical peuvent alors tous travailler leur morceau basé sur un même enregistrement en supprimant leur instrument, même lorsqu'ils ne sont pas en répétition. Le karaoké généralisé pouvant éventuellement conduire à la suppression de tous les instruments sauf un (*soloing*), on peut aussi simplement l'utiliser pour écouter un instrument seul lors d'un passage particulièrement apprécié d'un morceau de musique.
- Le remixage, c'est-à-dire la re-configuration personnalisée des paramètres de mixage d'une œuvre, entre autre la balance, la localisation des différents instruments, les effets sur les instruments, la compression…Cette fois-ci on peut imaginer un auditeur mélomane souhaitant adapter une œuvre afin de profiter au mieux de son équipement hi-fi et de son environnement d'écoute, ou les DJ amateurs et professionnels testant différentes interactions possibles avec le morceau.

Le but du projet DReaM est de chercher à donner accès à l'écoute active à des auditeurs qui n'ont pas forcément de connaissances particulières en traitement du signal ou en musique, c'est-à-dire sans modification majeure de leur mode d'écoute habituel. Une des idées principales de DReaM pour mener cet objectif à bien est de conserver un format traditionnel pour les signaux audio et non pas de développer un nouveau format multi-pistes, et ce pour plusieurs raisons. La première est liée à la difficulté de créer un nouveau format qui soit

1. http ://www.institut-langevin.espci.fr/
2. http ://www.telecom-paristech.fr/
3. http ://www.labri.fr/
4. http ://www.gipsa-lab.grenoble-inp.fr/
5. http ://www.iklaxmedia.com/
6. Le terme instrument est considéré ici au sens large, et peut désigner n'importe quelle source au sein du morceau de musique, ce peut être en particulier un instrument de musique mais aussi une voix voire une autre source sonore.

normalisé, ce processus étant très difficile et complexe et possédant une très forte inertie. La seconde est qu'une fois le nouveau format normalisé, il faut créer des logiciels ou modifier ceux existants déjà pour permettre la lecture du nouveau format. Ceci peut être relativement simple pour les logiciels utilisés dans un ordinateur, cependant pour du matériel dédié cela peut être beaucoup plus complexe. La troisième raison est qu'il est parfois difficile de rendre populaire un nouveau format de données, le public étant souvent réticent à l'idée de changer ou de modifier une partie de ses bases de données musicales ainsi que ses usages et habitudes. La dernière raison est liée à la réticence des maisons de disque à mettre à disposition du grand public les pistes individuelles. C'est pour ces raisons que le projet DReaM s'est d'abord focalisé sur une technique permettant l'écoute active avec des signaux musicaux au format non compressé standard, tel que porté sur les CD-Audio et les fichiers .wav (c'est-à-dire des signaux stéréophoniques, échantillonnés à 44.1 kHz, avec des échantillons codés en PCM 16 bits). En effet dans ce cas les problèmes liés à un nouveau format de données sont amoindris : même si l'utilisateur ne dispose pas sur tout son matériel du logiciel DReaM il peut tout de même écouter la musique comme il le faisait avant. Nous verrons de plus que ce choix permet aussi de répondre en partie au problème des pistes individuelles pour les maisons de disques. Il est intéressant de noter que des extensions des travaux dont nous allons parler rapidement par la suite sont à l'étude pour les signaux compressés (par exemple pour le format AAC de MPEG rendu omniprésent grâce notamment à iTunes).

1.2 Séparation de sources informée

Pour relever ce défi, le projet DReaM propose une nouvelle approche au problème de la *séparation de sources*, qui consiste précisément à extraire un ou plusieurs signaux (les sources) à partir de plusieurs mélanges de ces sources (les observations). Il s'agit ici d'exploiter la configuration particulière de la production musicale par rapport aux conditions habituelles des problèmes de séparation de sources, où deux différences majeures sont à noter. La première est le très faible nombre d'observations par rapport au nombre de sources. En effet, la séparation de sources est un problème très complexe qui nécessite généralement un nombre conséquent d'observations, si possible supérieur aux nombre de sources que l'on cherche à séparer si l'on souhaite avoir une séparation efficace de bonne qualité (on parle de mélange sur-déterminé). Or dans le cas de signaux audio stéréophoniques il n'y a que deux observations : la voie droite et la voie gauche ; et généralement nettement plus de deux instruments jouant simultanément, de l'ordre de la dizaine pour une large part des œuvres musicales dans de nombreux styles (on parle alors de mélange sous-déterminé). La seconde différence par rapport aux problématiques usuelles de séparation de sources est que le projet DReaM incorpore la partie mixage amont : on possède à l'origine les sources individuelles, que l'on mélange dans un mix stéréophonique (dont on maîtrise les paramètres), et c'est à partir de ce mix que l'on souhaite récupérer les sources originales avec une qualité satisfaisante. On cherche donc à contrebalancer le fait que le mélange soit sous-déterminé par le fait que l'on connaît les sources et les paramètres de mixage, plus

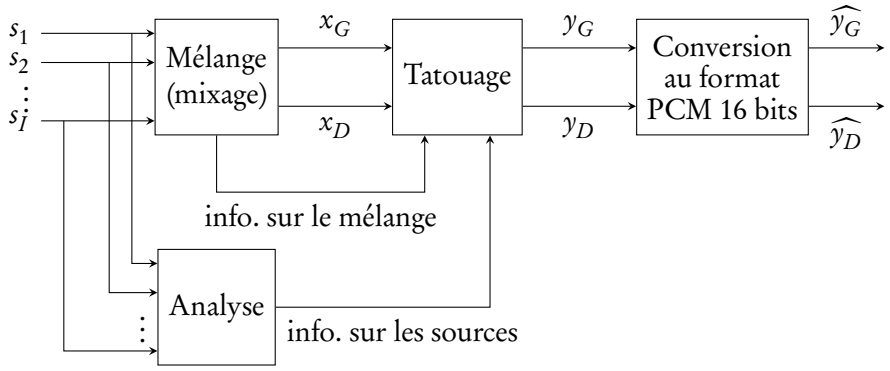


FIGURE 1.1 – Codeur mix actif DReaM

précisément en extrayant des paramètres relatifs à ces données qui peuvent permettre une séparation de bonne qualité (d'où l'origine du nom de la séparation de sources informée ou SSI). **Le projet DReaM propose alors de transmettre l'information nécessaire à la séparation par un processus de tatouage audio-numérique spécifique qui est l'objet de la majeure partie des travaux de thèse présentés dans ce manuscrit**. Le mix tatoué est bien sûr conservé au format non compressé standard évoqué précédemment. De cette façon, les utilisateurs peuvent profiter du mix de la manière habituelle même si la technologie DReaM n'est pas accessible sur tout leur matériel. Ceci permet d'éviter totalement le point évoqué précédemment sur la difficulté à rendre populaire un nouveau format (puisqu'il n'y a justement pas réellement de changement de format), et répond partiellement aux deux autres points sur la complexité du processus de normalisation et la rétro-compatibilité. Par rapport à la mise à disposition des pistes séparées par les maisons de disque, l'idée du projet DReaM est qu'il va être possible de cibler une qualité maximale de séparation dans le cas du *soloing* (un seul instrument joué) inférieure à la qualité originale, ce qui peut être un argument pour rassurer les maisons de disques et les encourager à investir dans l'écoute active. Les étapes décrites précédemment constituent un encodage du mix actif, dont un schéma récapitulatif est présenté figure 1.1.

Au niveau de la restitution du signal, le mix étant dans un format standard il peut être lu sur n'importe quelle plateforme habituelle de façon normale (y compris après fixation sur un support physique tel que le CD-Audio), ou alors de façon active en utilisant un logiciel spécifique DReaM. Dans ce logiciel DReaM, l'information tatouée est extraite puis exploitée lors d'un processus de séparation de sources informée. L'utilisateur peut alors modifier le mix initial portant le tatouage en appliquant des traitements plus ou moins complexes suivant les limites du logiciel DReaM. La figure 1.2 résume les différentes étapes que l'on vient de décrire et qui constituent le décodeur de mix actif.

La technologie comporte donc deux éléments clés :

– Un logiciel encodeur pour le traitement spécifique des signaux lors de la phase de production de l'œuvre musicale (mixage à partir des pistes studio). Ce logiciel utilise les

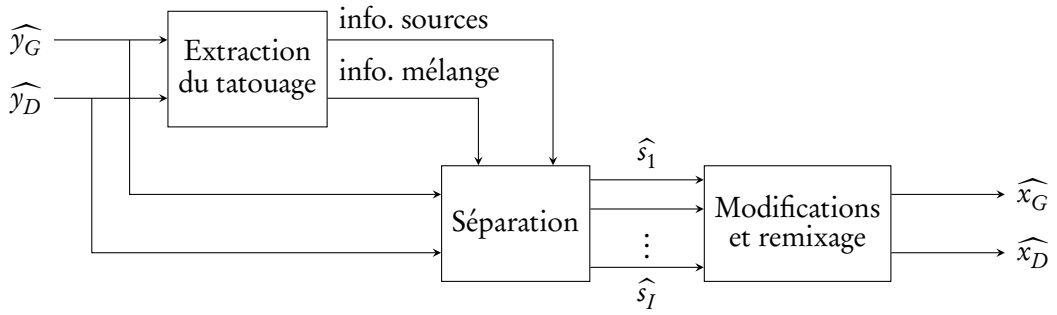


FIGURE 1.2 – Décodeur mix actif DReaM

pistes séparées et fournit le signal mixé stéréo tatoué, le tatouage contenant l'information permettant une séparation efficace des sources par le logiciel décodeur. Ce logiciel est destiné à être utilisé par les artistes et plus généralement par les professionnels de la production musicale.

– Un logiciel décodeur pour l'écoute active de la musique sur un ordinateur, un smartphone ou une tablette muni d'un système d'exploitation classique (Mac OS, Windows, Linux, Androïd), à partir des fichiers générés par l'encodeur DReaM. Ce logiciel, destiné au grand public, se présentera sous la forme typique des lecteurs multimédia avec une interface graphique conviviale pour activer et contrôler les fonctionnalités d'écoute active (boutons, ascenseurs…). Il est important de noter que, du fait de la compatibilité de format, les fichiers pourront être lus par n'importe quel lecteur traditionnel y compris les platines CD-audio (sans toutefois pouvoir bénéficier de l'écoute active).

1.3 Positionnement du présent travail

Chaque laboratoire partenaire du projet DReaM a contribué en développant une méthode différente de SSI. La méthode de l'institut Langevin repose sur un algorithme de reconstruction du type Griffin et Lim [SD13], celle de Télécom Paristech sur un filtrage de Wiener adapté (éventuellement multi-canaux) [LBR11], celle du LaBRI repose sur un filtrage par formation de voies [GM13], et celle du GIPSA-lab repose soit sur une inversion locale [PG11a], soit sur un codage des sources [PGB10a]. Pour toutes ces méthodes de SSI, le type d'information additionnelle est similaire. Il s'agit tout d'abord d'informations sur le protocole de mixage (par exemple des paramètres de filtres de mixage ou des matrices de mélange), et surtout des paramètres d'un modèle de représentation temps-fréquence des différents signaux sources (le modèle utilisé dépend des méthodes).

Ces différentes méthodes ont fait l'objet de plusieurs études comparatives, dont [LGS+12]. La performance de ces méthodes de SSI est caractérisée par un compromis entre qualité de séparation et quantité d'information additionnelle. Une qualité correcte est globalement

atteinte pour un débit d'information additionnelle de l'ordre de 2 à 10kb/s par source et par canal, ce qui représente, pour 10 sources, un débit de l'ordre de 100kb/s par canal environ. Lorsque ce débit augmente la qualité s'améliore et tend vers une valeur limite dépendante des modèles utilisés dans chaque méthode (qui peut être une qualité excellente si l'on code en plus le résidu). Augmenter le débit n'est donc pas forcément avantageux.

La partie tatouage du projet DReaM a été déléguée au GIPSA-lab, et le développement d'un système de tatouage adapté aux besoins du projet DReaM va être l'objet de la majeure partie de ce manuscrit. Il devra bien sûr s'appuyer sur une stratégie de tatouage efficace permettant un haut débit d'insertion qui s'obtiendra au mieux en substituant toutes les caractéristiques inaudibles du signal hôte. Il devra de plus permettre une flexibilité du débit d'insertion pour une meilleure synergie avec les méthodes de SSI. Par bien des points il peut donc s'apparenter à un schéma de codage audio perceptuel à débit paramétrable, à la fois dans la recherche de ces composantes inaudibles mais aussi dans son usage au sein de la chaîne d'écoute active. Contrairement aux applications de tatouage standards, le mix tatoué doit être manipulable comme le serait le flux d'un système de codage multi-pistes et permettre une écoute de qualité même lorsque seules quelques pistes en sont extraites.

Pour finir sur le positionnement de ce travail sur une note en lien avec des travaux extérieurs au projet DReaM, à notre connaissance il n'existe qu'une seule autre série de travaux portant sur l'utilisation du tatouage audio comme support d'une information visant à traiter le signal hôte lui-même : il s'agit de travaux de Samaali *et al.* [STHAM09, STHAM10, SMT12, Sam13], dans lesquels l'information de tatouage est utilisée pour corriger des distorsions subies par le signal hôte dans la chaîne de transmission, et notamment lors d'une étape de compression. Dans ces travaux, la finalité et le contenu de l'information transmise, ainsi que les spécifications (notamment de robustesse à la compression) sont donc différents de ceux du projet DReaM, et par conséquent la technique de tatouage est aussi différente. Nous reviendrons sur ces points par la suite.

Le contexte étant maintenant posé, nous pouvons présenter dans le chapitre suivant un état de l'art des deux piliers de notre étude que sont le tatouage audionumérique et le codage perceptuel.

Chapitre 2

Le tatouage numérique

Sommaire

2.1 Cadre applicatif

Nous avons vu en introduction générale de ce document les origines historiques du tatouage numérique. Si nous continuons cette chronologie, à partir de la fin des années 1970, quelques articles et brevets traitant de ce que l'on appelle aujourd'hui tatouage numérique apparaissent sporadiquement. Le terme *digital watermarking* en lui-même semble avoir été utilisé pour la première fois par Komatsu et Tominaga [KT88], mais le domaine du tatouage numérique ne prend son essor qu'à partir du milieu des années 1990, avec l'apparition de conférences dédiées et l'intérêt croissant que lui portent certains organismes ou compagnies.

Le tatouage connaît une forte avancée théorique à la toute fin des années 1990 avec la relecture d'un papier de Costa [Cos83], basé sur les travaux de communication numérique de Gel'fand et Pinsker [GP80]. Regardant ce papier sous un nouvel angle, certains auteurs comme Cox et Miller [CMM99] ou Chen et Wornell [CW01] présentent le tatouage comme un procédé de communication avec information additionnelle. En effet, jusqu'à présent dans la plupart des systèmes de tatouage, l'hôte était considéré comme un bruit et donc traité comme tel, c'est-à-dire de manière statistique puisqu'il s'agit en règle générale de signaux aléatoires. En réalité l'hôte est parfaitement connu au moment de l'insertion du tatouage et on peut donc effectivement tirer profit de cette information additionnelle pour faciliter l'insertion du tatouage. [1] Cette nouvelle vision du tatouage amène à distinguer les tatouages dits *aveugles* et les tatouages dits *informés*. Aujourd'hui la quasi-totalité des systèmes de tatouage est constituée de systèmes de tatouage informés.

Le tatouage numérique est né d'une volonté sécuritaire, dans un but de protection des droits d'auteur ou des droits de propriété. Cependant aujourd'hui, l'utilisation des techniques utilisées en tatouage numérique se retrouve aussi dans d'autres disciplines très proches voire qui se chevauchent : le *tatouage pour la transmission de données* et la *stéganographie*. La distinction entre ces disciplines est parfois difficile à faire, cependant on peut par exemple choisir la classification donnée dans [CMB01] :

- **Le tatouage** est la pratique qui consiste à modifier de manière imperceptible une œuvre afin d'y insérer un message qui fait référence à cette même œuvre. Le point clé dans cette vision du tatouage est que l'élément central est l'œuvre, c'est elle que l'on veut protéger (ou améliorer, modifier…) et donc l'information insérée doit être en rapport avec cette œuvre. La grande majorité des systèmes de tatouage s'intéressent à des questions de sécurité, et on peut distinguer deux grands types de tatouage, les tatouages robustes (*e.g.* [MBCM10]) et les tatouages fragiles (*e.g.* [WLDB08]). Les systèmes de tatouage robustes cherchent à transmettre une information qui a du sens, cette information doit donc être protégée des attaques volontaires ou non qui pourraient perturber le décodage. Dans le cas des systèmes de tatouages fragiles on cherche à assurer l'intégrité de l'œuvre, l'information transmise n'a pas exactement de sens en elle-même, mais son bon décodage doit prouver que l'œuvre n'a pas été modifiée.

1. Dans l'article [Cos83] de Costa, l'analogie pour le tatouage est celle d'une personne voulant écrire sur une feuille tâchée : il faut savoir écrire là où il sera facile pour quelqu'un de pouvoir lire le message.

- **La stéganographie** est la pratique qui consiste à communiquer un message souvent à haut débit en dissimulant l'existence même de la communication. On se situe sur un plan légèrement différent du tatouage : dans le cas de la stéganographie on souhaite qu'une tierce personne examinant l'œuvre soit incapable de dire si elle contient un message ou pas ; alors que dans le cas du tatouage si elle parvient à détecter qu'un message est inséré ce n'est pas important tant qu'elle n'arrive pas à le neutraliser. Un exemple de système stéganographique est présenté dans [CM03].
- **Le tatouage pour la transmission de données** est la pratique qui consiste à insérer de manière imperceptible de l'information dans une œuvre. Le message transmis peut avoir un sens par lui même sans être nécessairement rattaché à l'œuvre. La différence entre tatouage « classique » et tatouage pour la transmission de données est très fine et n'est pas toujours marquée dans la littérature, mais conduit à des mises en œuvre différentes (débit faible et forte redondance pour le tatouage contre débit élevé pour la transmission de données, détection du message pour le tatouage contre décodage du message pour la transmission de données). Les systèmes de tatouage pour la transmission de données peuvent être génériques (*e.g.* [CSTS05]), mais sont souvent spécifiques et optimisés pour une application précise. Les applications spécifiques qui utilisent le tatouage pour la transmission de données sont très variées, et concernent entre autre : les applications à contenu augmenté (*e.g.* [BAB+06]), l'aide à la stationarisation (*e.g.* [LJS05]), le débruitage (*e.g.* [Sam13]).

Les travaux de thèse présentés dans cet ouvrage ont pour but le développement d'un système de tatouage haute-capacité non sécuritaire pour signaux audionumériques. Le développement du système de tatouage s'est fait dans le cadre du projet DReaM, et son utilisation initiale est la transmission de données relatives au signal porteur (plus précisément le signal tatoué est un mix musical et les données insérées sont de l'information facilitant la séparation des sources constituant le mélange). En suivant la catégorisation introduite précédemment, le système développé serait alors un système de tatouage. Cependant ses aspects d'application non-sécuritaire et de haut débit ainsi que son utilisation possible pour des données non relatives au signal porteur nous amènent à le considérer comme un système de tatouage pour la transmission de données. L'état de l'art qui suit sera donc fortement orienté vers ce champs d'étude : le tatouage de signaux audionumériques pour la transmission de données.

2.2 Critères de performance des systèmes de tatouage

Un système de tatouage peut être décrit par un certain nombre de critères de performance, que nous allons présenter dans cette section. La liste proposée n'est bien entendu pas exhaustive et l'importance relative de chacune des caractéristiques dépend bien sûr de l'application considérée. De plus, les définitions présentées ici rejoignent généralement celles données dans [CMB01] et peuvent varier légèrement suivant les auteurs voire les applications.

2.2.1 Imperceptibilité

L'*imperceptibilité*, ou *fidélité*, désigne la ressemblance entre l'hôte tatoué et l'hôte original, conditionnée par la force d'incrustation choisie pour le tatouage. Cette différence est en général mesurée juste avant utilisation normale du contenu, c'est-à-dire après d'éventuels traitements tolérés sur le signal pour qu'il soit délivré ou stocké (par exemple compression). Un canal de mauvaise qualité permettra donc généralement des tatouages de plus fortes puissances, étant donné que l'hôte sera de toute façon dégradé naturellement (cela posera en revanche probablement des problèmes pour d'autres caractéristiques, comme la robustesse ou le débit d'insertion). Dans la grande majorité des cas, on exige une imperceptibilité « parfaite » pour le développement d'un système de tatouage. Cependant dans certains cas particuliers, il devra être nécessaire d'accepter une légère perte de qualité afin de gagner sur d'autres caractéristiques, comme la robustesse ou le coût de calcul informatique. [2]

Dans certains cas simples, ou dans des cas où le tatouage est très peu puissant, l'imperceptibilité est facilement caractérisée par des critères simples et faciles d'utilisation, tels que la norme L^2 du tatouage. Dans d'autres cas plus complexes, généralement lorsque la limite d'audibilité est approchée, l'évaluation de l'imperceptibilité est plus difficile et demande généralement l'utilisation d'outils complexes, voire la mise en place de tests subjectifs. Ce point sera discuté plus en détail en section 3.5.

2.2.2 Débit d'insertion / charge

Cette caractéristique représente la quantité d'information binaire insérée dans l'hôte, qui peut varier énormément suivant le type d'application. Le *débit d'insertion* ou *charge* varie de quelques bits/s pour des applications sécuritaires à quelques kbits/s voire dizaines de kbits/s pour des applications de transmission de données.

2.2.3 Robustesse

La *robustesse* d'un tatouage représente sa résistance à des opérations usuelles de traitement du signal appliquées sur l'hôte. Ces opérations peuvent être liées à :

- des traitements tolérés sur des contenus tatoués dépendants du scénario applicatif (compression, filtrage, ré-échantillonnage, transmission etc.),
- des attaques volontaires d'un contrevenant cherchant à mettre en défaut le décodeur.

Le recensement de ces opérations s'arrête dès que la qualité du signal devient trop dégradée pour une utilisation acceptable de ce dernier. Dans notre étude, seules des perturbations en lien avec la conservation du format CD-Audio seront prises en compte (quantification PCM), d'autant qu'aucun attaquant n'aurait d'intérêt à essayer de corrompre l'information transmise par notre système de tatouage, non sécuritaire.

2. Ce dernier point est généralement assez complexe à gérer, puisqu'il dépend de la complexité calculatoire, parfois difficile à mesurer, et surtout du matériel et du système d'exploitation.

2.2.4 Sécurité

La *sécurité* se réfère à la capacité du tatouage (ou de la technique de tatouage) à résister à des attaques volontairement destinées à neutraliser l'effet du tatouage. On distingue généralement trois grandes catégories d'attaques :

- Les attaques de détection sont des attaques dites passives, où l'attaquant accède dans une mesure plus ou moins grande aux données tatouées. Cela peut entraîner l'identification partielle du tatouage (par exemple être capable de reconnaître deux versions du tatouage sur deux hôtes différents), voire son décodage total.
- Les attaques de suppression cherchent à masquer partiellement ou totalement le tatouage afin de le rendre indétectable par les décodeurs.
- Les attaques d'insertion désignent le cas où l'attaquant produit une contrefaçon en tatouant un hôte qui ne devrait pas l'être.

La distinction, souvent assez floue, entre sécurité et robustesse se situe au niveau des types de traitements. Quand on utilise des traitements génériques (sous-échantillonnage, filtrage…), on parle plutôt de robustesse, mais lorsque la modification est liée plus ou moins fortement à la technique de tatouage elle-même, on parle plutôt de sécurité. [3]

2.2.5 Extraction aveugle ou informée

Suivant le système, l'extraction d'un tatouage peut requérir l'hôte original (non tatoué) en plus de l'hôte duquel on veut extraire le tatouage. Dans ce cas on parle d'*extraction informée* et dans le cas contraire on parle d'*extraction aveugle*. Notons ici qu'il est très important de ne pas confondre extraction informée et système de tatouage informé. Le premier terme signifie que l'hôte non tatoué est nécessaire à l'extraction, alors que le second fait référence à un système de tatouage tirant parti de la connaissance du signal hôte lors de l'insertion.

2.2.6 Coût informatique

Cette caractéristique englobe la puissance de calcul nécessaire pour faire l'insertion ou l'extraction et le temps de calcul nécessaire pour ces opérations. Pour certaines applications, ces contraintes sont peu problématiques alors que pour d'autres on peut par exemple avoir besoin d'un décodage en temps réel (ce qui signifie généralement traitement aussi rapide que la vitesse à laquelle la musique est jouée, avec un temps de latence initial assez faible).

2.3 Schéma de principe théorique

La figure 2.1 présente le schéma de principe théorique d'un système de tatouage pour la transmission de données. On adoptera la convention de notation suivante : u désigne un

3. On trouve notamment de nombreux travaux sur la sécurité où les études sont basées sur le principe de Kerckhoffs, c'est-à-dire que l'on doit toujours considérer que les attaquants connaissent le fonctionnement du système et ignorent uniquement la clé secrète.

scalaire, **u** un vecteur du domaine temporel, **U** un vecteur d'un domaine transformé et $U(n)$ sa n-ième composante.

Le système (figure 2.1a) est constitué de trois blocs, dont les deux principaux sont l'insertion 2.1b et l'extraction 2.1c du tatouage, le troisième représentant les perturbations (ou bruit) affectant le signal entre ces deux blocs. L'insertion du tatouage requiert en entrée le message **m** et l'hôte **x**, ainsi qu'une clé **k** éventuellement secrète. Cette étape d'insertion donne alors le signal hôte tatoué **y**, qui sera modifié par des perturbations, résultant en un signal hôte tatoué modifié $\widehat{\mathbf{y}}$. L'extraction quant à elle requiert le signal hôte tatoué, la clé **k** si elle a été utilisée lors de l'insertion et le signal hôte non tatoué **x** si l'extraction est informée.

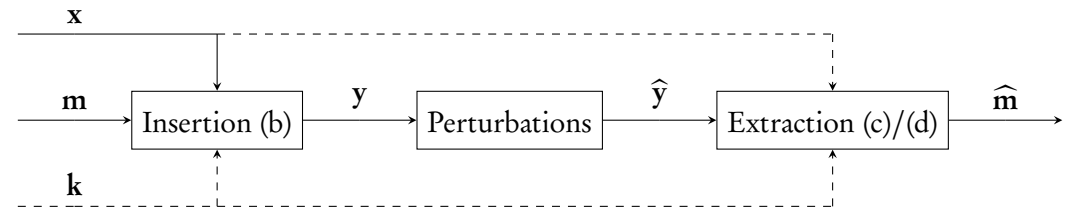
La première étape de l'insertion 2.1b est le *codage source* : le message à insérer **m** est codé dans un alphabet adapté à l'insertion (généralement un code binaire), puis éventuellement compressé, crypté ou adjoint de codes correcteurs d'erreurs, résultant en un code à insérer **c**. Parallèlement, le signal hôte **x** est transformé dans le *domaine d'insertion* grâce à la transformation $\mathscr{T}$, résultant en un vecteur $\mathbf{X} = \mathscr{T}(\mathbf{x})$. Le choix de ce domaine d'insertion est important : il peut être fréquentiel (transformée de Fourier ou rarement MDCT), cepstral, paramétrique (coefficients d'analyse LPC ou d'ondelettes), etc. Le code à insérer **c** est alors *mis en forme* dans le domaine d'insertion résultant en un tatouage **W**. Cette étape est parfois appelée codage canal dans la littérature car la correspondance entre un code **c** et un vecteur de tatouage **W** peut être modélisée par un dictionnaire de canal, éventuellement adapté en fonction de l'hôte. Dans le cas des tatouages sécuritaires ou robustes, cette mise en forme est souvent faite à l'aide d'une clé secrète **k**. Cette étape peut de plus être informée par le signal hôte, on parle dans ce cas de tatouage *informé* et dans le cas contraire de tatouage *aveugle* dont il a déjà été brièvement question en section 2.1. Le tatouage **W** est alors simplement sommé à l'hôte dans le domaine d'insertion, et le signal tatoué résultant $\mathbf{Y} = \mathbf{X} + \mathbf{W}$ est retourné dans le domaine de représentation original de l'hôte, résultant en un vecteur $\mathbf{y} = \mathscr{T}^{-1}(\mathbf{Y})$. Notons que par souci de simplicité on appelle parfois *fonction d'insertion* la fonction f définie par :

$$f(\mathbf{X}, \mathbf{c}, \mathbf{k}) = \mathbf{Y} \tag{2.1}$$

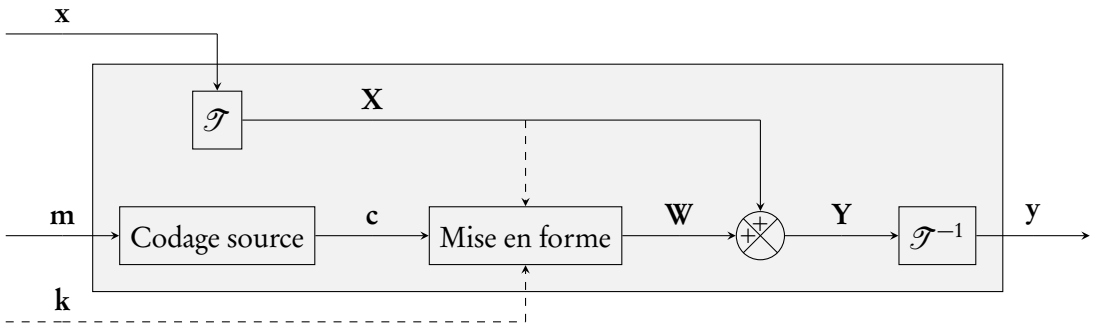
Cette fonction regroupe l'étape de codage canal et l'addition du vecteur de tatouage **W** à l'hôte **X**.

Une fois l'insertion réalisée et avant l'extraction du tatouage, le signal tatoué **y** peut être perturbé. Ces perturbations peuvent avoir des origines diverses, comme du bruit sur le canal de transmission, des détériorations du support de stockage, de la compression, des transformations du contenu ou encore des attaques visant le tatouage. Le signal résultant est noté $\widehat{\mathbf{y}}$. De façon similaire, nous notons donc avec des chapeaux toutes les variables calculées ou estimées à partir de $\widehat{\mathbf{y}}$.

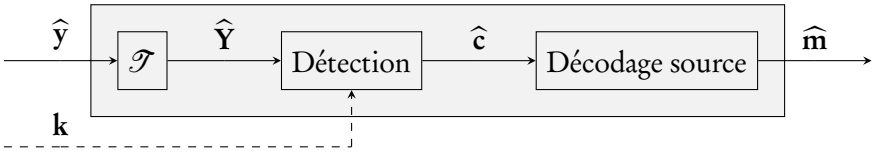
L'extraction du tatouage est en quelque sorte l'inversion de l'étape d'insertion, avec quelques légères différences si l'extraction est aveugle (figure 2.1c) ou informée (figure 2.1d). Dans le cas d'une extraction aveugle, le signal tatoué $\widehat{\mathbf{y}}$ est tout d'abord transformé dans le domaine d'insertion grâce à la transformation $\mathscr{T}$, résultant en un signal $\widehat{\mathbf{Y}}$. Le code $\widehat{\mathbf{c}}$ est ensuite extrait lors de l'étape de *décodage canal*, avec l'aide de la clé **k** si celle-ci a été utilisée lors de l'insertion. Finalement lors de l'étape de *décodage source*, le code $\widehat{\mathbf{c}}$ est décodé et l'on



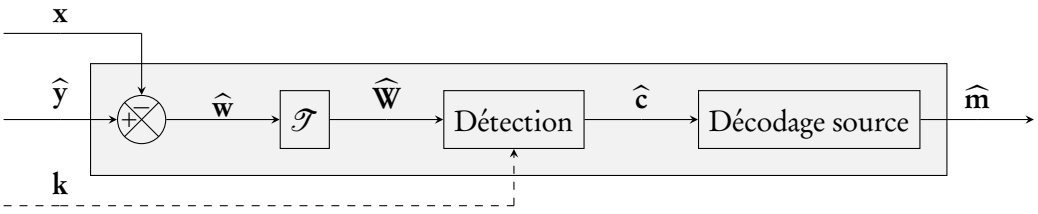
(a) Schéma global



(b) Insertion



(c) Extraction aveugle



(d) Extraction informée

FIGURE 2.1 – En (a), schéma de principe d'un système de tatouage. Le bloc d'insertion est détaillé en (b) et le bloc d'extraction est détaillé en (c) dans les cas d'une extraction aveugle et en (d) dans le cas d'une extraction informée. Les traits en pointillés entre les différents éléments indiquent des liens qui peuvent être présents ou non suivant le système de tatouage.

récupère le message $\widehat{\mathbf{m}}$. Dans le cas d'une extraction informée, on soustrait tout d'abord au signal tatoué reçu $\widehat{\mathbf{y}}$ le signal hôte non tatoué $\mathbf{x}$, ce qui donne l'image du tatouage reçu $\widehat{\mathbf{W}}$ dans le domaine temporel $\mathscr{T}^{-1}(\widehat{\mathbf{w}}) = \widehat{\mathbf{W}}$ [4] que l'on transforme dans le domaine d'insertion, résultant en un vecteur $\widehat{\mathbf{W}}$. Dans ce cas la détection est alors facilitée par rapport à une extraction aveugle et se déroule comme dans le cas d'une extraction aveugle, c'est-à-dire que le code binaire $\widehat{\mathbf{c}}$ est extrait lors de l'étape de *décodage canal*, et le message $\widehat{\mathbf{m}}$ est finalement récupéré lors de l'étape de *décodage source*.

Nous ne détaillerons pas les différentes méthodes de *codage source* (et de décodage associées), étant donné que celles-ci sont très variées et sont choisies pour chaque système de tatouage suivant des critères propres à l'application concernée. Dans la section suivante, nous allons détailler les principales techniques utilisées pour la mise en forme, couramment appelées techniques d'insertion, c'est-à-dire permettant de générer le vecteur de tatouage $\mathbf{W}$ à partir du code $\mathbf{c}$.

2.4 Techniques d'insertion

2.4.1 Étalement de spectre

L'*étalement de spectre* dénomme un ensemble de techniques originellement développées et affinées pour le domaine des télécommunications militaires, entre les années 1920 et 1960 environ [Sch82, PSM82, PSM84]. En effet, on cherchait alors à développer des systèmes de communication robustes à certains type d'attaques, et plus particulièrement les attaques de *jamming*. Une attaque est dite de *jamming* lorsque l'on essaye de perturber volontairement le canal de transmission (par exemple l'air quand on se situe dans le cadre des communications par ondes électromagnétiques) avec une puissance limitée. La meilleure stratégie pour faire face à ce type d'attaques est de répartir l'information sur toute la plage de fréquence disponible d'une façon inconnue de l'attaquant, d'où le nom d'*étalement de spectre* et les nombreuses recherches effectuées dans ce domaine. De plus, cette répartition de l'information sur une large bande de fréquence permet d'être intrinsèquement robuste à la plupart des opérations classiques de traitement du signal. On comprend alors facilement pourquoi les techniques d'étalement de spectre ont rapidement été adaptées au tatouage lorsque ce domaine a connu un essor. Les premières techniques de tatouage par étalement de spectre datent du milieu des années 1990, par exemple [CKLS97].

Au niveau de l'implémentation dans le cas du tatouage, l'étape de mise en forme (figure 2.1b) peut être schématisée dans la plupart des cas comme représenté figure 2.2. Le message binaire $\mathbf{c}$ à insérer est assigné de façon bijective à un motif à large bande $\mathbf{p}$, généralement une séquence pseudo-aléatoire. Cette association bijective est habituellement réalisée à l'aide de la clé $\mathbf{k}$. Le motif $\mathbf{p}(\mathbf{c}, \mathbf{k})$ est ensuite ajusté, généralement afin de respecter certains critères d'imperceptibilité, résultant en un vecteur de tatouage $\mathbf{W}$. Différentes méthodes pour cet ajustement sont présentes dans la littérature (cf. [CMB01]) :

4. Ceci n'est vrai que si les transformations $\mathscr{T}$ et $\mathscr{T}^{-1}$ sont linéaires, ce qui est généralement le cas.

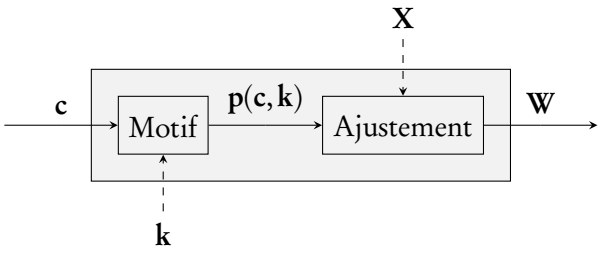


FIGURE 2.2 – Schématisation de la mise en forme dans le cas de l'étalement de spectre

- Pondération de $\mathbf{p}(\mathbf{c},\mathbf{k})$ par un facteur indépendant du signal hôte $\mathbf{X}$:

$$\mathbf{W} = \gamma \cdot \mathbf{p}(\mathbf{c},\mathbf{k}) \tag{2.2}$$

  e.g. [CKLS97]. Ce type de pondération simple est généralement réalisé afin d'assurer que l'erreur de tatouage (c'est-à-dire la différence entre le signal tatoué et le signal original, ici égal à $\mathbf{W}$) soit inférieure à une certaine valeur en moyenne quadratique.
- Pondération de $\mathbf{p}(\mathbf{c},\mathbf{k})$ par un facteur scalaire dépendant du signal hôte :

$$\mathbf{W} = \gamma(\mathbf{X}) \cdot \mathbf{p}(\mathbf{c},\mathbf{k}) \tag{2.3}$$

  e.g. [MF03]. Ce type de pondération vient généralement de modèles perceptifs plus complexes qu'une simple majoration de l'erreur quadratique, ce qui permet des tatouages plus puissants (et donc plus robustes ou de plus grand débit) sans pour autant baisser la qualité du signal tatoué.
- Pondération terme à terme de $\mathbf{p}(\mathbf{c},\mathbf{k})$ par un vecteur dépendant là encore de l'hôte :

$$\mathbf{W} = \gamma(\mathbf{X}) \odot \mathbf{p}(\mathbf{c},\mathbf{k}) \tag{2.4}$$

  où $\odot$ désigne la multiplication terme à terme de deux vecteurs. Un exemple de système utilisant ce type de pondération est décrit dans [WPD99]. Il s'agit d'une variante plus raffinée du type de pondération précédent, permettant généralement un meilleur compromis entre débit et robustesse mais résultant en une détection plus complexe.

L'extraction se résume quant à elle dans le cas le plus simple à une maximisation de la corrélation entre le signal reçu $\widehat{\mathbf{Y}}$ et les différentes séquences possibles :

$$\widehat{\mathbf{c}} = \arg\max_{c} \left\langle \mathbf{p}(\mathbf{c},\mathbf{k}) | \widehat{\mathbf{Y}} \right\rangle \qquad \text{(extraction aveugle)} \tag{2.5}$$

$$\widehat{\mathbf{c}} = \arg\max_{c} \left\langle \mathbf{p}(\mathbf{c},\mathbf{k}) | \widehat{\mathbf{W}} \right\rangle \qquad \text{(extraction informée)} \tag{2.6}$$

Cependant, il a été montré que cette méthode n'est optimale que dans le cas où l'erreur entre $\mathbf{p}(\mathbf{c},\mathbf{k})$ et $\widehat{\mathbf{Y}}$ est gaussienne, ce qui n'est généralement pas le cas en pratique. On peut alors

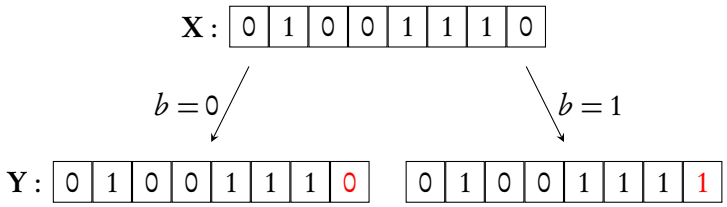


FIGURE 2.3 – Exemple d'insertion par LSB d'un bit b dans une composante codée sur 8 bits

utiliser des méthodes dérivées, comme la corrélation normalisée [CMB01] :

$$\widehat{\mathbf{c}} = \arg\max_{c} \frac{\left\langle \mathbf{p}(\mathbf{c},\mathbf{k}) | \widehat{\mathbf{Y}} \right\rangle}{||\mathbf{p}(\mathbf{c},\mathbf{k})||_2 \cdot ||\widehat{\mathbf{Y}}||_2}. \tag{2.7}$$

Les techniques par étalement de spectre permettent donc d'obtenir des schémas robustes, mais au prix d'un débit relativement faible (au maximum environ 300 bits par seconde) qui est incompatible avec les besoins du projet DReaM.

2.4.2 Techniques par quantification

Least Significant Bits

La technique par quantification la plus simple est la LSB (*Least Significant Bit*) développée au début des années 1990 [TNM90, TRS+93]. Le principe de base est de remplacer le bit de poids faible de chaque composante de $\mathbf{X}$ par les bits du code binaire à insérer $\mathbf{c}$ (cf. figure 2.3). Ce type de technique n'est généralement utilisé que dans les cas où les composantes de $\mathbf{X}$ représentent des nombres entiers codés sur N bits (*e.g.* la valeur d'un pixel d'une image codée en niveaux de gris ou la valeur d'un échantillon d'un signal de musique codé en PCM). Le remplacement du bit de poids faible génère alors une erreur facilement quantifiable.

Le schéma de base peut être adapté pour insérer K bits par composante, en modifiant les K bits de poids faible au lieu d'un seul. Dans ce cas, il peut être nécessaire d'ajouter ou de soustraire la valeur 2^K à la composante afin de minimiser l'erreur d'insertion (cf. figure 2.4). Quel que soit le nombre de bits insérés, le principe de décodage est évident puisqu'il s'agit de récupérer les bits de poids faible des valeurs tatouées.

Cette technique est très peu robuste. En effet presque tous les types de dégradations applicables sur le signal hôte (même à très faible puissance) vont modifier les bits de poids faible des valeurs. Ce type de technique est donc généralement utilisé dans des applications où des attaques ne doivent pas avoir lieu, en stéganographie, et dans les cas où l'on cherche justement à ne pas être robuste pour pouvoir détecter si l'hôte a subi des modifications (tatouages fragiles). De plus pour que ce type de technique soit efficace, la transformée $\mathscr{T}$ doit être exactement inversible (lors de l'implémentation), c'est-à-dire une bijection de $\mathbb{Z}^N$ dans lui-même. Cette contrainte fait que la plupart du temps les systèmes de tatouage par

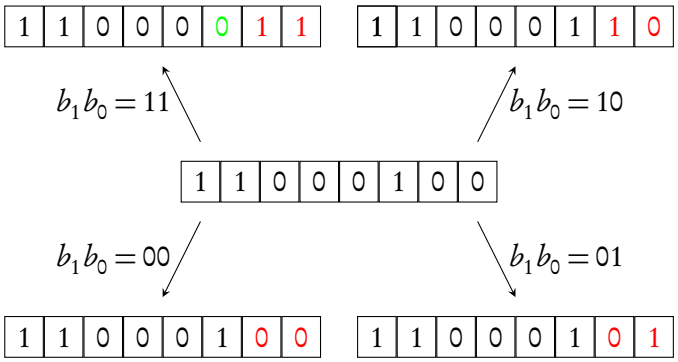


FIGURE 2.4 – Exemple d'insertion par LSB de deux bits b_1 et b_0. Dans le cas $b_1 b_0 = 11$, $2^2 = 4$ a été soustrait après remplacement des bits afin de diminuer l'erreur d'insertion de 3 à 1 en valeur absolue.

LSB se font dans le domaine temporel (*i.e.* $\mathscr{T} = \mathscr{I}$, la transformée identité), ou dans des domaines obtenus par des transformées dites *integer-to-integer* (ITI), par exemple [GYS06].

QIM scalaire pour message binaire

Les autres techniques d'insertion par quantification se basent sur la dénommée *Quantization-Index Modulation* (QIM), développée par Chen et Wornell à la fin des années 1990 [CW99, CW01]. Cette technique s'applique lorsque les coefficients de $\mathbf{X}$ prennent des valeurs *continues* (par opposition à la technique de LSB où les valeurs possibles sont intrinsèquement quantifiées). La QIM peut donc être vue comme une généralisation de la technique de LSB.

Afin de mieux expliquer le fonctionnement de la QIM, nous allons tout d'abord étudier le cas le plus simple possible, celui où nous cherchons à insérer 1 bit d'information $c \in \{0,1\}$ dans un coefficient $X \in \mathbb{R}$, sans clé $\mathbf{k}$. Dans ce cas précis la technique de QIM simple consiste à se munir de deux quantificateurs *entrelacés* (*i.e.* dont les représentants ne se superposent jamais), Q_0 et Q_1. Ces deux quantificateurs sont généralement scalaires uniformes (même si ce n'est pas obligatoire dans le principe de QIM) et définis à partir d'un quantificateur prototype Q de pas de quantification Δ :

$$Q : \mathbb{R} \longrightarrow \mathbb{R} \tag{2.8}$$

$$X \longmapsto \left[\frac{X}{\Delta}\right] \cdot \Delta \tag{2.9}$$

où $[.]$ désigne l'opérateur d'arrondi. Les deux quantificateurs Q_0 et Q_1 sont alors définis par :

$$Q_0(X) = Q\left(X + \frac{\Delta}{4}\right) - \frac{\Delta}{4} \tag{2.10}$$

$$Q_1(X) = Q\left(X - \frac{\Delta}{4}\right) + \frac{\Delta}{4} \tag{2.11}$$

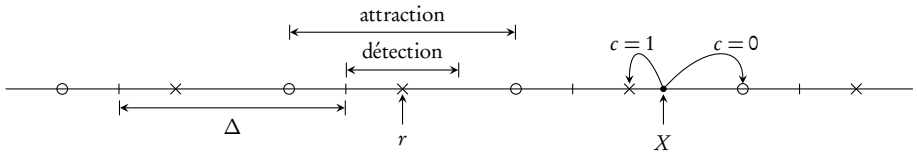


FIGURE 2.5 – Exemple d'insertion d'un bit d'information c dans un coefficient X, les représentants du quantificateur prototype Q sont représentés par des traits verticaux, ceux de Q_0 par des cercles et ceux de Q_1 par des croix. Les zones d'attraction (de largeur Δ) et de détection (de largeur $\Delta/2$) pour un représentant r donné sont aussi représentées. Figure reprise de [CW01]

L'insertion de $c \in \{0,1\}$ dans X se fait alors simplement en quantifiant X avec le quantificateur Q_c, comme représenté sur la figure 2.5 :

$$Y = Q_c(X) \tag{2.12}$$

En reprenant les notations de la figure 2.1, on a :

$$Y = Q_c(X) = X + (Q_c(X) - X) \tag{2.13}$$

Cette technique rentre donc bien dans le cadre du schéma général décrit en figure 2.1 en considérant que le vecteur de tatouage $\mathbf{W}$ est la différence entre le signal hôte et sa version quantifiée :

$$W = Q_c(X) - X \tag{2.14}$$

Les représentants des quantificateurs Q_0 et Q_1 forment des réseaux géométriques de $\mathbb{R}$:

$$\Lambda_0 = -\frac{\Delta}{4} + \Delta\mathbb{Z} \tag{2.15}$$

$$\Lambda_1 = \frac{\Delta}{4} + \Delta\mathbb{Z} \tag{2.16}$$

On appelle zone d'attraction d'un représentant r du quantificateur Q_c l'ensemble des antécédents de r par Q_c. On peut remarquer que l'ensemble des zones d'attraction des représentants d'un quantificateur Q_c définit une partition de $\mathbb{R}$. En effet chaque valeur possible de X peut être quantifiée par Q_0 et sera toujours quantifiée sur le même représentant. Les éléments de cette partition (les zones d'attraction) sont de plus les cellules de Voronoï des représentants, c'est-à-dire de Λ_c.

L'extraction de l'information insérée est effectuée en choisissant l'indice du réseau possédant le représentant le plus proche du coefficient reçu $\widehat{Y}$:

$$\widehat{c} = \underset{i\in\{0,1\}}{\arg\min}\left(\operatorname{dist}\left(\widehat{Y}, \Lambda_i\right)\right) \tag{2.17}$$

FIGURE 2.6 – Exemples de réseaux QIM pour $D = 4$, les représentants des différents réseaux sont des croix, des cercles, des carrés et des triangles

avec $\text{dist}\left(\widehat{Y}, \Lambda\right) = \min_{r \in \Lambda} \left|\widehat{Y} - r\right|$. On appelle zone de détection d'un représentant r l'ensemble des valeurs qui seront décodées comme étant r, c'est-à-dire l'ensemble des points pour lesquels r est le représentant le plus proche parmi les représentants de Q_0 et de Q_1. L'ensemble des zones de détections de tous les représentants définit alors une partition de $\mathbb{R}$. Les éléments de cette partition (les zones de détection) sont les cellules de Voronoï des représentants des deux quantificateurs, c'est-à-dire de $\Lambda_0 \cup \Lambda_1$.

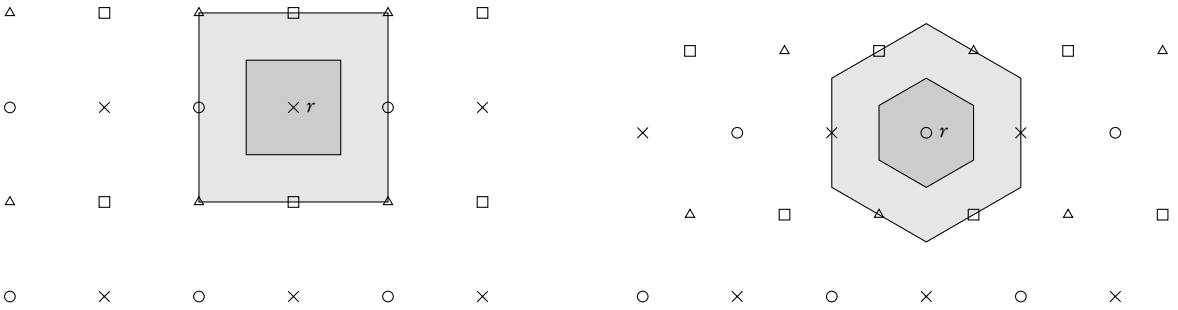
Comme on peut le voir sur la figure 2.5, tant que la perturbation avant l'extraction reste plus faible que $\Delta/4$ en valeur absolue, l'extraction se fait sans erreur. En effet dans ce cas la valeur perturbée reste dans la même zone de détection qu'avant perturbation. L'erreur introduite par l'insertion est limitée entre $-\Delta/2$ et $\Delta/2$, et en suivant le modèle classique (haute résolution) de Bennett pour la quantification [Ben48], cette erreur suit une distribution uniforme sur l'intervalle $\left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right]$. En fait, on peut montrer que l'hypothèse de distribution uniforme de l'erreur est exacte si l'on utilise des *dithered* quantificateurs [Sch64, ZF92], c'est-à-dire si pour chaque coefficient $X(n)$ de $\mathbf{X}$, les réseaux géométriques utilisés sont décalés aléatoirement et indépendamment de $\mathbf{X}$ d'une valeur $k(n) \in [-\Delta/2, \Delta/2]$. Le vecteur $\mathbf{k}$ résultant doit être connu lors de l'extraction et peut de plus servir de clé. Le second intérêt de ce décalage aléatoire est que dans les cas d'applications sécuritaires il permet de rendre plus difficile la détection du tatouage par un parti tiers.

QIM scalaire pour symbole : charge plus élevée

Ce schéma de base peut être généralisé afin d'insérer une charge plus élevée (c'est-à-dire une plus grande quantité d'information, voir section 2.2), par exemple un symbole c ayant D valeurs possibles. De façon analogue au cas binaire, l'ensemble de réseaux $\{\Lambda_d\}_{d \in [0, D-1]}$ est alors défini par :

$$\Lambda_d = \frac{(2d+1)\Delta}{2D} + \Delta\mathbb{Z}, \qquad d \in [0, D-1] \tag{2.18}$$

Un exemple de réseaux pour $D = 4$ est donné figure 2.6. L'erreur d'insertion est la même que précédemment et suit donc toujours la même distribution uniforme sur l'intervalle $\left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right]$. Cependant pour qu'il n'y ait aucune erreur d'extraction il faut que la perturbation soit inférieure à $\Delta/(2D)$ en valeur absolue. Le *dithering* peut s'appliquer de la même façon que dans le cas binaire.



(a) Réseaux cubiques (b) Réseaux hexagonaux

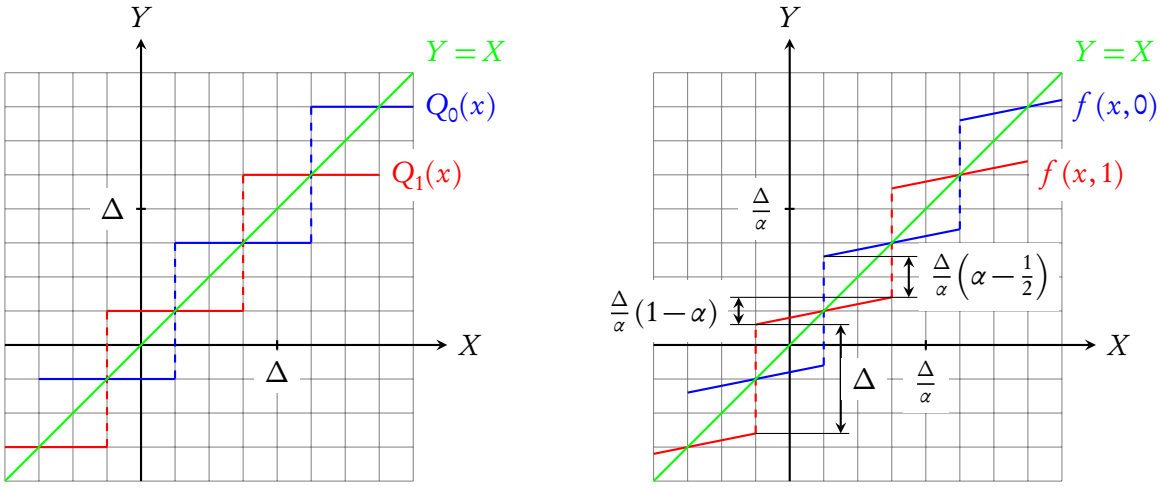
FIGURE 2.7 – Exemples de réseaux cubiques et hexagonaux pour un schéma QIM insérant 2 bits d'information dans un coefficient à 2 dimensions $\mathbf{X}$. Il y a donc 4 réseaux imbriqués $\{\Lambda_d\}_{0\leq d<4}$ qui sont représentés par des cercles, des croix, des triangles et des carrés. La zone gris claire (respectivement gris foncée) représente la région d'attraction (respectivement région de détection) d'un point r d'un des Λ_d. Les régions d'attraction des points de Λ_d sont les cellules de Voronoï de Λ_d, et les régions de détection sont les cellules de Voronoï de $\cup_{d=0}^{3}\Lambda_d$.

QIM vectorielle : dimension de l'espace d'insertion plus élevée

Le schéma de QIM peut aussi être généralisé en augmentant la dimension des quantificateurs, autrement dit en utilisant des quantificateurs vectoriels. On peut par exemple reprendre le premier cas simple étudié mais cette fois avec un coefficient $\mathbf{X} \in \mathbb{R}^2$ et un message à coder ayant 4 valeurs possibles $\mathbf{c} \in \{0,1,2,3\}$. Plusieurs choix de réseaux sont alors possibles, certains étant mieux adaptés que d'autres. La figure 2.7 en représente deux différents. Les critères les plus importants dans le choix du réseau sont généralement les suivants :

- Être régulier au niveau de la distorsion introduite, c'est-à-dire avoir des régions d'attraction proches d'une boule (par exemple, les réseaux hexagonaux de 2.7b répondent mieux à cette contrainte que les réseaux cubiques de 2.7a).
- Être robuste, c'est-à-dire avoir des réseaux Λ_i les plus éloignés possible les uns des autres.

La détermination de bons quantificateurs vectoriels est doublement complexe, du point de vue mathématique mais aussi de l'implémentation. Elle ne sera pas traitée ici mais de nombreux articles discutent de ce sujet, comme [MK05, KMKM00, ZSE02]. Nous verrons que dans nos systèmes de tatouage présentés aux parties II et III nous n'utilisons que la version scalaire de la QIM. En effet, nous verrons que l'interaction entre les différentes fréquences est traitée par un modèle psychoacoustique et ne doit donc pas être traitée à nouveau par un tatouage multi-dimensionnel qui est de plus beaucoup plus complexe à mettre en œuvre.



(a) QIM, 1 bit

(b) DC-QIM, 1 bit

FIGURE 2.8 – Allure des fonctions d'insertion des schémas QIM et DC-QIM pour un tatouage binaire.

DC-QIM : QIM plus robuste

L'inconvénient majeur de la QIM est sa mauvaise résistance au bruit. En effet, l'extraction est sans erreur tant que le bruit reste en dessous d'un certain seuil. Au-delà, les performances de décodage se dégradent fortement. Cependant, il est possible de modifier le schéma de base afin d'obtenir un schéma plus robuste, la *Distortion-Compensated* QIM (DC-QIM) [CW01].

Nous nous limitons ici aux principes de la DC-QIM scalaire, sachant que cette technique possède bien sûr les mêmes extensions que la QIM classique, que ce soit pour la charge insérée par coefficient ou la dimension des quantificateurs utilisés. En reprenant le premier cas simple d'insertion d'un bit d'information c dans un coefficient $X \in \mathbb{R}$, l'idée est de remplacer l'insertion par simple quantification (2.12) par l'opération suivante :

$$Y = f(X,c) = Q_c(\alpha X) + (1-\alpha)X \tag{2.19}$$

où $\alpha \in [0,1]$ est un paramètre à optimiser en fonction du bruit. Lorsque $\alpha = 1$, on retrouve la QIM originale, et lorsque $\alpha = 0$, on a $Y = X$ et l'insertion est dégénérée (il n'y a pas d'insertion). Les différences entre les fonctions d'insertion QIM et DC-QIM pour le cas scalaire binaire sont présentées figure 2.8. La différence par rapport à la QIM n'est pas évidente à saisir. Dans le cas de la QIM, la valeur originale de l'hôte est quantifiée, c'est-à-dire qu'elle est déplacée vers un représentant pour être égale à ce représentant. L'ensemble des valeurs possibles pour Y est donc un ensemble discret de points. Dans le cas de la DC-QIM, l'hôte est aussi déplacé vers un « représentant » (qui est l'invariant par la fonction d'insertion $f(X,c)$ le plus proche), mais seulement d'une distance dépendante de α. Or, l'invariant par

$f(X,c)$ le plus proche de X est $Q_c(\alpha X)/\alpha$:

$$f\left(\frac{Q_c(\alpha X)}{\alpha},c\right)=\frac{Q_c(\alpha X)}{\alpha} \tag{2.20}$$

Et on a de plus :

$$f(X,c)=X+\alpha\left(\frac{Q_c(\alpha X)}{\alpha}-X\right) \tag{2.21}$$

ce qui montre bien que l'insertion se fait en rapprochant X de l'invariant, d'une distance égale à α fois la distance avant insertion.

Dans le cas scalaire, la DC-QIM est souvent appelée Schéma de Costa Scalaire (SCS pour *Scalar Costa Scheme*) [EBTG03], en référence aux similarités avec le schéma présenté par Costa dans [Cos83]. Lorsque ce terme est utilisé, les notations changent généralement pour rejoindre le formalisme de Costa :

$$Y=X+\alpha\left(Q'_c(X)-X\right) \tag{2.22}$$

Le passage des notations DC-QIM aux notations SCS se fait grâce à la relation $\Delta=\alpha\Delta'$. La notation de la DC-QIM est généralement plus pratique que la notation SCS, car dans ce formalisme l'erreur maximale commise est toujours égale à $\Delta/2$, indépendamment de α (comme on peut le voir sur la figure 2.8).

Le choix de α détermine le comportement du système face à une perturbation donnée. Par exemple dans [Bas11], le paramètre est optimisé à des fins sécuritaires, et dans [EBTG03] il est optimisé numériquement pour maximiser la capacité du canal dans le cas d'un bruit blanc gaussien. Afin de bien se représenter l'effet du coefficient α sur la probabilité d'erreur, nous avons tracé en figure 2.9 la probabilité d'erreur (expérimentale) en fonction de α pour un décodage de DC-QIM affecté par un bruit blanc additif Gaussien de moyenne nulle et d'écart-type σ. Les paramètres sont $\Delta=1$, $\sigma_1=1/8$ et $\sigma_2=1.1\sigma_1$. On voit clairement qu'il existe un α qui minimise le taux d'erreur et que cet α optimal dépend du rapport Δ/σ.

Application aux signaux audio

Les techniques par quantification ont souvent été utilisées dans les systèmes pour la transmission de données dans la mesure où elles permettent un débit d'insertion élevé. Par exemple dans [IS04], les auteurs appliquent la LSB sur la phase de signaux stéréophoniques en utilisant le fait que le système auditif humain est peu sensible aux modifications de phases. Un débit d'une centaine de kbits/s est ainsi atteint. La LSB se retrouve aussi dans [CS02a] sur les échantillons temporels du signal avec un débit d'environ 170 kbits/s. Une application de quantification des coefficients MDCT [PG11a] laisse entrevoir des débits d'une centaine de kbits/s. Finalement lorsque la LSB est appliquée sur les coefficients IntMDCT (*Integer Modified Discrete Transform*, une transformée qui sera abordée en détails dans la suite de ce manuscrit) comme dans [GYS06], elle permet d'atteindre des débits de l'ordre de 120

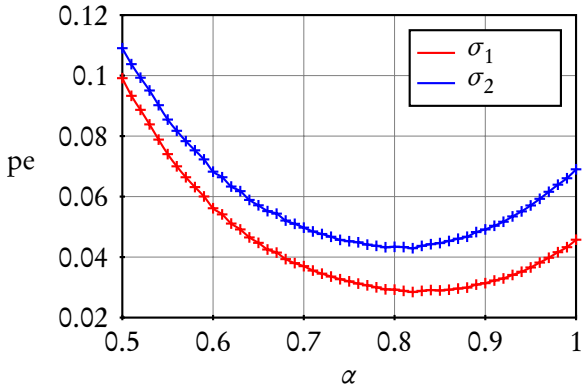


FIGURE 2.9 – Taux d'erreur pe de la DC-QIM en fonction de α pour un bruit blanc additif Gaussien.

kbits/s. Ces débits sont dans la limite basse des performances attendues pour le système de tatouage DReaM qui devra donc se concentrer sur l'amélioration du débit final.

Seul [CS02b] qui utilise la LSB sur les coefficients d'une transformée en ondelettes atteint des débits allant jusqu'à 400 kbits/s. Malheureusement, le respect de la contrainte d'inaudibilité n'est pas contrôlé du fait de l'absence de modèle perceptuel. La qualité du mix étant prépondérante dans le système DReaM, notre étude va donc porter sur l'adaptation des techniques de quantification à un environnement de tatouage fortement contraint par des considérations perceptuelles.

Chapitre 3

Le codage audio perceptuel

Sommaire

3.1 Introduction

L'apparition du disque compact audio (CD-Audio ou CD-A) en 1982 [IK98] fut un évènement décisif pour l'audionumérique, qui entraîna peu à peu le déclin du disque vinyle analogique, même si certains auteurs pensent que la qualité est moindre [Stu95]. Cependant avec le développement des systèmes multimédia, en particulier au niveau des systèmes en réseaux filaires et sans fils, le format utilisé dans les CD-A a commencé à devenir problématique. En effet les caractéristiques de ce format sont les suivantes :

- signal stéréophonique (2 canaux),
- fréquence d'échantillonnage de 44.1kHz,
- échantillons codés en PCM sur 16 bits.

Le débit résultant est d'environ 1.4Mbits/s ($2 \times 44100 \times 16$) [1], ce qui est trop élevé pour la plupart des systèmes usuels de communication sans fil. Cependant, les utilisateurs s'étant habitués à la qualité du CD-A (qu'on appellera qualité PCM dans le reste de ce document), celle-ci s'est imposée comme référence à atteindre pour les systèmes de compression. C'est dans ce cadre que le codage audio perceptuel s'est développé au milieu des années 1980, de façon plus ou moins indépendante des travaux sur le codage des signaux de parole [Joh07]. Le but du codage audio est d'avoir une représentation d'un signal audio plus compacte qu'avec le format PCM 16 bits, pour faciliter la transmission et le stockage, et qui puisse être décodée rapidement (généralement en temps réel, contrairement à l'encodage). Deux grands types de compression sont envisageables pour les signaux audio :

- **Suppression des redondances statistiques**. Les signaux audio ne sont effectivement pas totalement imprévisibles (au sens de la théorie de l'information), il est donc possible de trouver des codes adaptés qui réduisent le débit du signal. Ce type de compression ne modifie pas le signal, uniquement la manière de le représenter, on parle donc de *compression sans perte*.
- **Suppression et modification d'éléments sonores de manière indétectable**. En effet, comme nous allons le voir en section 3.2, le système auditif humain n'est pas « parfait » et deux signaux audio distincts peuvent être perçus de la même façon. Ce type de compression modifie le signal original, on parle donc de *compression avec perte*.

Dès le milieu des années 1980, de nombreuses travaux ont été publiés sur le codage de signaux audio de haute qualité (*i.e.* qualité de référence PCM 16 bits ou approchant) [Bra87b, Joh88].

Le but de ce chapitre n'est pas de faire une présentation exhaustive de tous les types de codeurs audio perceptuels (une description détaillée peut être obtenue par exemple dans [PS00]), mais de présenter un schéma général qui corresponde à la plupart des codeurs performants, comme MP3 [ISO98] et AAC [ISO09] de MPEG, ou AC-3 [ATS10] de Dolby. Ce schéma est présenté figure 3.1 et le principe général est le suivant. Le signal audio à coder est tout d'abord découpé en trames d'une durée variant de 2 à 50 ms. Chaque trame est ensuite transformée dans le domaine temps-fréquence (TF). Parallèlement une analyse de la trame basée sur un modèle psychoacoustique (modèle de perception des sons par l'oreille

1. Ce débit est celui des données audio uniquement, il ne tient pas compte des codes correcteurs d'erreurs et de la modulation *Eight-to-Fourteen* (EFM) ou des données additionnelles présentes sur un CD-A.

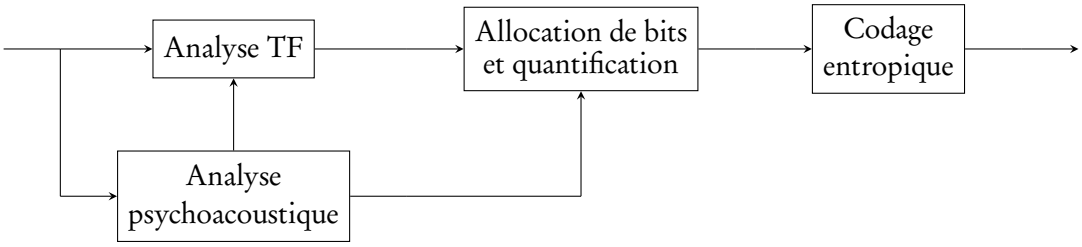


FIGURE 3.1 – Schéma de base d'un codeur audio perceptuel.

humaine) est effectuée, qui peut éventuellement guider l'analyse temps-fréquence. C'est de l'exploitation de cette analyse psychoacoustique que va provenir la majeure partie de la compression avec perte. Cette analyse va en effet permettre de savoir quelles dégradations sont applicables sur la trame courante sans impacter la qualité du signal audio reconstruit. L'étape suivante va donc être une étape de quantification et d'allocation de bits, c'est-à-dire que l'on va quantifier les coefficients de l'analyse temps-fréquence avec des quantificateurs d'une certaine précision (auxquels on alloue un certain nombre de bits). Le résultat de cette quantification va être alors codé paramétriquement et entropiquement (compression sans perte), et finalement les codes de chacune des trames vont être combinés pour donner un flux binaire qui est le signal audio codé. Dans la suite de ce chapitre, nous allons donc décrire plus en détails les grandes étapes du codage audio, qui sont l'analyse psychoacoustique, l'analyse temps-fréquence, la quantification et le codage entropique.

3.2 Analyse psychoacoustique

Le but de l'analyse psychoacoustique est de mesurer quelles dégradations peuvent être appliquées au signal audio afin de le modifier sans en altérer la qualité. Pour cela il faut tenir compte des imperfections du système auditif humain, décrit en détail dans [FZ06]. Nous allons nous contenter ici de décrire les éléments de psychoacoustique couramment utilisés dans les codeurs audio perceptuels, en commençant par les domaines de représentation adaptés à l'appareil auditif humain. Il est important de noter que dans le domaine de la psychoacoustique les différents effets sont généralement modélisés à partir des valeurs moyennes recueillies lors d'expériences réalisées sur un grand nombre d'individus, mais sont généralement assez reproductibles.

3.2.1 Domaines de représentation adaptés

Principe

L'étude de signaux audio dans le cas général se fait la plupart du temps dans le domaine fréquentiel voire temps-fréquence (TF), c'est-à-dire que l'on découpe le signal audio en

trames de quelques millisecondes ou quelques dizaines de millisecondes et que l'on en étudie le contenu fréquentiel à cette échelle locale. Lorsque l'on calcule une représentation fréquentielle d'un signal, les outils mathématiques utilisés la plupart du temps donnent un résultat échantillonné linéairement suivant une échelle en Hertz (comme on le verra dans la section 3.3). Or, la nature de l'oreille interne de l'homme (et plus particulièrement de la cochlée) fait que le système auditif humain ne fonctionne justement pas avec une base en Hertz. Il fonctionne d'une façon similaire à un banc de filtres, dont la bande passante augmente avec la fréquence. Nous avons par exemple une meilleure résolution fréquentielle pour des basses fréquences que pour des hautes fréquences (c'est-à-dire une meilleure aptitude à séparer deux sons de fréquences différentes). Afin de mieux prendre en compte cet aspect du système auditif humain, diverses représentations fréquentielles adaptées ont été développées à partir de mesures psychoacoustiques effectuées sur un grand nombre de sujets. Nous allons en présenter deux parmi les plus couramment utilisées, les bandes critiques proposées par Zwicker et l'ERB.

Bandes critiques

Au début des années 1960 des acousticiens, et en particulier Zwicker [Zwi61], réalisent des expériences qui tendent à montrer que le système auditif humain ne traite pas les fréquences indépendamment mais par bandes, dites *bandes critiques*, la position et la largeur de ses bandes n'étant pas fixe et pouvant varier. L'échelle dite des *bandes critiques* est alors obtenue par simplification en prenant les 24 ou 25 bandes critiques adjacentes telles que la première démarre à 20 Hz (limite basse du domaine de l'audible). La table 3.1 référence les bandes critiques telles que proposées et présentées par Zwicker. Il existe aussi une version continue, plus pratique à utiliser que les bandes critiques, on parle alors généralement d'échelle *Bark*. Plusieurs expressions mathématiques existent pour modéliser la conversion entre Hertz et Barks, les plus simples étant [PS00] :

$$b = 13 \arctan(0.00076 f) + 3.5 \arctan\left(\left(\frac{f}{7500}\right)^2\right) \tag{3.1}$$

$$f = \frac{52548}{b^2 - 52.56b + 690.39} \tag{3.2}$$

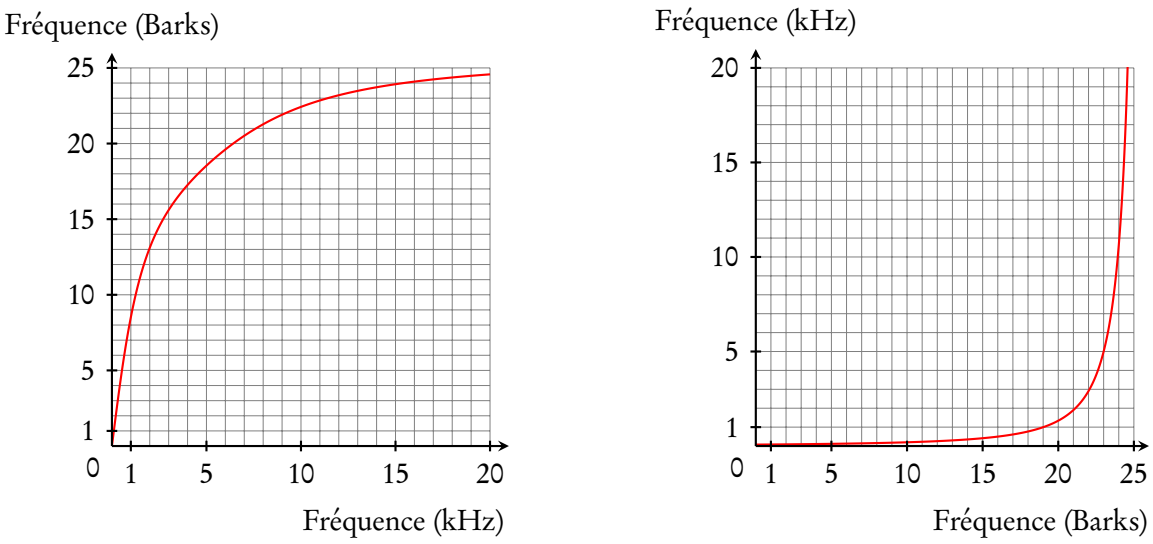
Ces deux fonctions sont représentées figure 3.2. Notons d'ailleurs l'allure pseudo-logarithmique pour la conversion de Hertz vers Barks.

ERB

Une autre représentation dans la même lignée que les bandes critiques de Zwicker est l'ERB (en anglais *Equivalent Rectangular Bandwidth*). On considère ici que le système auditif humain agit comme un banc de filtres rectangulaires, dont la bande passante varie en fonction de la fréquence centrale. Plusieurs articles détaillent des expériences visant à déterminer les caractéristiques de ce banc de filtres, dont on peut citer les premiers parus [Hou77, Pat76].

Bande critique	Fréquence centrale (Hz)	Fréquence de coupure (Hz)	Bande passante (Hz)
		20	
1	50	100	80
2	150	200	100
3	250	300	100
4	350	400	100
5	450	510	110
6	570	630	120
7	700	770	140
8	840	920	150
9	1000	1080	160
10	1170	1270	190
11	1370	1480	210
12	1600	1720	240
13	1850	2000	280
14	2150	2320	320
15	2500	2700	380
16	2900	3150	450
17	3400	3700	550
18	4000	4400	700
19	4800	5300	900
20	5800	6400	1100
21	7000	7700	1300
22	8500	9500	1800
23	10500	12000	2500
24	13500	15500	3500

TABLE 3.1 – Limites et fréquences centrales des bandes critiques selon Zwicker [Zwi61]



(a) Hertz/Bark (b) Bark/Hertz

FIGURE 3.2 – Fonctions de conversion Hertz/Bark et Bark/Hertz

De la même façon qu'avec les bandes critiques de Zwicker, il a été rapidement intéressant d'avoir une version continue de l'ERB, généralement appelée ERBS (*ERB Scale* en anglais). Comme pour l'échelle Bark, il existe plusieurs formules pour la conversion de Hertz à ERBS, la première étant probablement celle présentée dans [MG83] :

$$\mathrm{ERBS}(f) = 11.17 \cdot \ln\left(\frac{f + 0.312}{f + 14.675}\right) + 43 \tag{3.3}$$

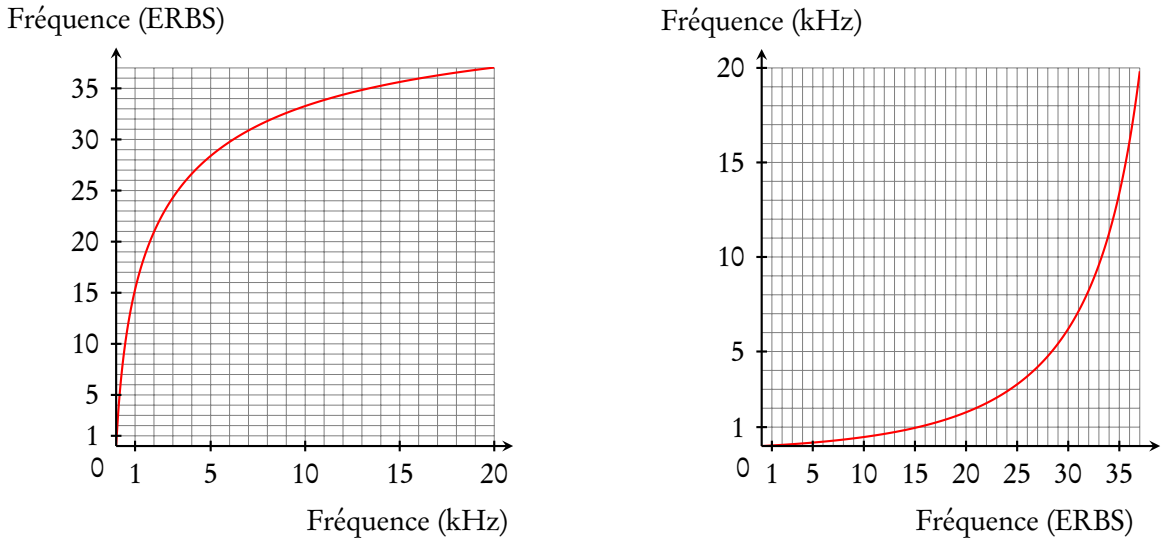
où f désigne la fréquence en kHz. Cette formule étant inversible, on obtient directement :

$$f = \frac{675.567}{47.035 - \exp(0.09\mathrm{ERBS}(f))} - 14.675 \tag{3.4}$$

Les courbes correspondant à ces fonctions sont représentées figure 3.3. Nous avons là aussi, bien sûr, une courbe pour la conversion de Hertz vers ERBS qui a une forme quasi-logarithmique. Les principales différences avec l'échelle des Barks de Zwicker est le nombre de bandes plus important, et la nature logarithmique conservée même dans les très basses fréquences.

3.2.2 Seuil d'audition absolu

Le but des codeurs audio étant de réduire le débit sans réduire la qualité, on va donc d'abord chercher à supprimer toutes les informations qui ne sont pas audibles. Un outil



(a) Hertz/ERBS

(b) ERBS/Hertz

FIGURE 3.3 – Fonctions de conversion Hertz/ERBS et ERBS/Hertz

couramment utilisé est le *seuil d'audition absolu*, qui représente pour chaque fréquence dans le domaine audible (généralement entre 20Hz et 20kHz) la pression acoustique nécessaire pour que l'on entende un son à cette fréquence pure. L'unité de mesure de l'intensité acoustique utilisée est généralement le $\mathrm{dB}_{\mathrm{SPL}}$, qui est défini comme la mesure de pression acoustique efficace en dB relative à une pression de référence de $20\,\mu Pa$:

$$p_{\mathrm{eff}}\,\mathrm{dB}_{\mathrm{SPL}} = 20\log_{10}\left(\frac{p_{\mathrm{eff}}}{20\cdot 10^{-6}}\right) \tag{3.5}$$

avec p_{eff} la pression en Pascal. Ce seuil est propre à chaque individu, varie avec l'âge (l'ouïe se dégrade avec le temps), et peut être affecté par notre environnement (détérioration de l'appareil auditif). Cependant certaines caractéristiques se retrouvent généralement chez tous les individus, comme le fait que l'on entende mieux les sons aux alentours de 3-4kHz. Un modèle possible du seuil d'audition L, là encore obtenu à partir de mesures psychoacoustiques sur un grand nombre de sujets, est donné par la formule suivante [Ter79] avec f la fréquence en kHz :

$$L(f) = 3.6\cdot f^{-0.8} - 6.5\exp\left(-0.6\,(f-3.3)^2\right) + 0.003\cdot f^4 \tag{3.6}$$

Le seuil d'audition absolu est aussi normalisé dans le document [ISO03] pour des valeurs entre 20Hz et 12,5kHz. La courbe correspondant à la formule ainsi que la courbe de la norme sont représentées figure 3.4. Pour les codeurs audio, l'idée est d'utiliser ce seuil d'audition absolu pour ne pas coder (ou plus exactement coder à 0) les composantes du signal qui sont sous cette courbe. Deux remarques importantes sont tout de même à faire :

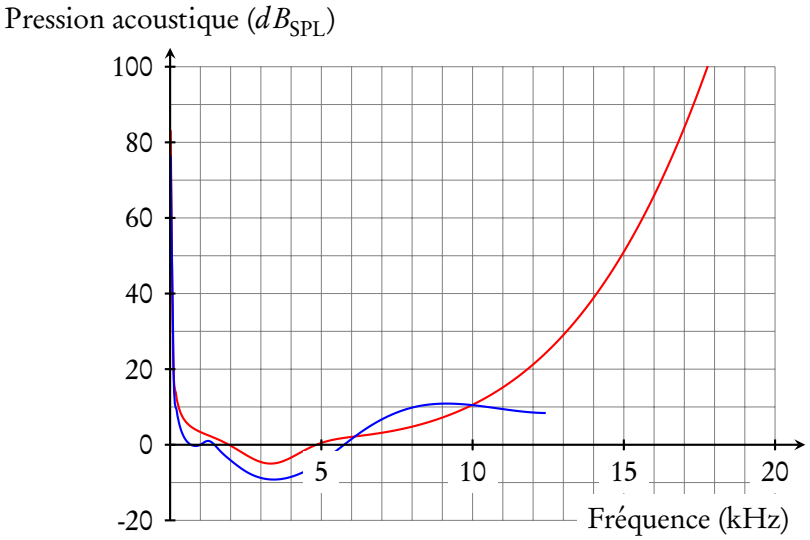


FIGURE 3.4 – Seuil d'audition absolu, en rouge avec la formule (3.6), et en bleu avec les valeurs du document normatif [ISO03]

- Le seuil d'audition absolu est calculé pour des fréquences pures. Le bruit ajouté par le codage audio est rarement constitué de sons purs, des ajustements sont donc nécessaires pour tenir compte de ce fait.
- Comme son nom l'indique, ce seuil est absolu. En effet comme on peut le voir sur la figure 3.4, une valeur du seuil de 0 dB_{SPL} a été choisie plus ou moins arbitrairement pour un son pur à 2 kHz dans le cas de la formule (3.6), et 1 kHz dans le cas de la norme [ISO03]. Cependant rien n'empêche l'utilisateur d'augmenter le volume du son auquel il écoute le signal audio. Il faut donc savoir à quel niveau mettre le signal audio PCM correspondant pour pouvoir le comparer à cette courbe. La convention la plus généralement prise par les codeurs audio est de faire correspondre le minimum du seuil d'audition (aux alentours de 3.5kHz) avec la puissance d'un signal de 1 bit d'amplitude.

3.2.3 Phénomènes de masquage

Le masquage réfère au fait que certains sons (dits *masqués*) peuvent être rendus inaudibles par la présence d'autres sons (dits *masquants*). Les codeurs audio vont donc chercher à prédire les composantes du signal audio qui sont masquées afin de ne pas les coder et de gagner en débit, et à quantifier les signaux en utilisant des quantificateurs les plus grossiers possibles (car nécessitant moins de bits pour coder les valeurs quantifier) tout en s'assurant que l'erreur de quantification soit masquée. On distingue généralement deux types de masquage (voir figure 3.6) :

- les masquages dits *simultanés*, lorsque le son masquant et le son masqué sont produits

en même temps (on parle aussi de masquage fréquentiel) ;
- les masquages dits *non-simultanés*, lorsque le son masqué est produit avant ou après le son masquant (on parle aussi de masquage *temporel*).

Masquage simultané

Les cas de masquage simultanés sont sensiblement différents suivant la nature du son masqué et du son masquant. Les études de la littérature s'attachent généralement à étudier et modéliser des cas simples, c'est-à-dire que les sons sont soit des sinusoïdes pures soit du bruit à bande étroite (de bande passante d'un Bark en règle générale). Ceci donne donc lieu à quatre grands cas :
- Son pur masquant un bruit (TMN pour *Tone Masking Noise* en anglais)
- Son pur masquant un son pur (TMT pour *Tone Masking Tone* en anglais)
- Bruit masquant un bruit (NMN pour *Noise Masking Noise* en anglais)
- Bruit masquant un son pur (NMT pour *Noise Masking Tone* en anglais)

Dans le cas du codage audio, l'erreur de codage est généralement un bruit, ou tout du moins sa nature est plus proche d'un bruit que d'un son pur. Malheureusement les études pour un signal masqué de type bruit sont peu nombreuses comparativement aux autres cas, et les résultats sont assez complexes. Dans tous les cas le principe du masquage simultané est que si deux sons sont joués simultanément, il est possible que le son le moins puissant soit masqué par le son le plus puissant. Pour que le phénomène de masquage simultané ait lieu, il faut que la différence de puissance entre les deux signaux soit assez élevée, et cette différence de puissance nécessaire dépend à la fois de la nature des signaux (son pur ou bruit) et de l'écart fréquentiel entre les deux sons. Autrement dit, plus un son s'éloigne en fréquence du son masquant, plus il doit être faible pour être masqué. Lorsque la puissance est en dB, cette décroissance est quasi-linéaire en fonction de la fréquence en Bark. On modélise donc le masquage fréquentiel par une fonction d'étalement triangulaire [MW98] dont le sommet correspond à la fréquence du son masquant. Un exemple de formule utilisée pour cette fonction d'étalement est la suivante :

$$SF(b) = 15.81 + 7.5(b + 0.474) - 17.5\sqrt{1 + (b + 0.474)^2} \tag{3.7}$$

où b représente le décalage en Bark par rapport à la fréquence centrale du son masquant et l'atténuation est donnée en dB (cf. figure 3.5). Néanmoins, si une seule fonction d'étalement est généralement utilisée pour les quatre types basiques de masquage, l'atténuation globale dépend du type de son masquant et de son masqué. En effet, on peut voir par exemple dans [PS00] qu'un bruit peut masquer un son pur plus faible de seulement 4 dB, alors qu'il faut parfois une atténuation de 24 dB pour masquer un bruit par un son pur. Lorsque l'on étudie une scène sonore complexe, ce qui est généralement le cas des codeurs audio traitant des signaux de musique, on fait l'hypothèse (plus ou moins réaliste) que les phénomènes de masquage fréquentiels sont additifs. Du point de vue de l'implémentation, on a donc généralement une convolution (ou un traitement approchant) dans le domaine des Barks du spectre (en dB) du signal par la fonction d'étalement, puis un ajustement de l'amplitude

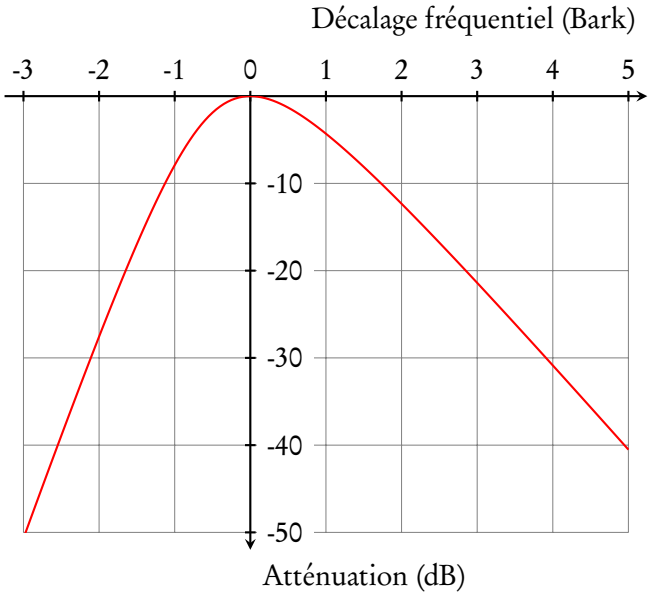


FIGURE 3.5 – Fonction d'étalement basique. La courbe représente l'atténuation en dB en fonction de la distance en fréquence entre le son masquant et le son masqué.

suivant la nature du son (son pur ou bruit). On considère ensuite que l'ajout d'un bruit (dans le cas des codeurs audio ce bruit sera l'erreur de codage) sous la courbe de masquage résultante sera inaudible, ce qui permet de choisir des quantificateurs qui soient les plus grossiers possibles tout en garantissant l'inaudibilité de la quantification.

Masquage non-simultané

On a vu précédemment qu'un son peut avoir un effet masquant sur un autre son simultané. Cependant la physique du système auditif humain, notamment les phénomènes d'intégration temporelle de l'information sonore, fait qu'un son peut aussi avoir une influence sur des sons qui lui sont postérieurs et même sur des sons qui lui sont antérieurs. La figure 3.6 présente de façon schématique les effets de ce que l'on appelle couramment le *post-masquage* et le *pré-masquage*. Du fait de la forte variance des résultats lors des différentes études menées, ces effets sont difficiles à modéliser et implémenter, et ils ne sont généralement pas utilisés directement pour la réduction du débit des codeurs audio. Cependant ils doivent être pris en compte pour assurer que le codeur audio ne dégrade pas la qualité du signal audio plus que prévu. En effet, on a vu précédemment que les codeurs audio travaillent par trame de quelques millisecondes ou dizaines de millisecondes. Or, les effets de la compression avec perte (qui viennent de la quantification de coefficients), sont généralement étalés sur la totalité de la trame temporelle. Prenons alors le cas d'une brusque variation de la puissance (et donc de l'effet de masquage fréquentiel). Si la variation est négative, c'est-à-dire si la puissance diminue, cela ne pose normalement pas de problème. Dans le pire des cas, la

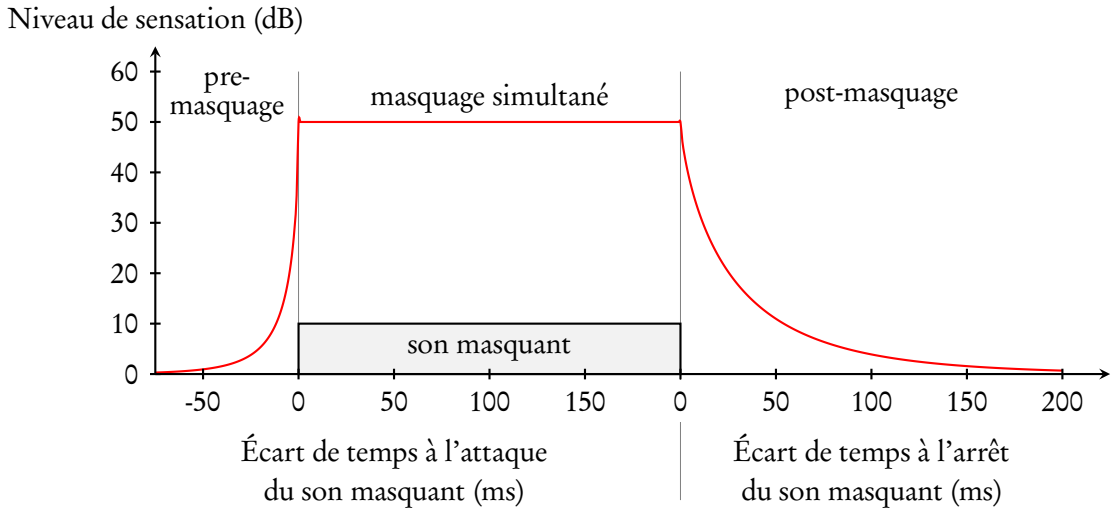


FIGURE 3.6 – Allure approximative des phénomènes de post-masquage et pré-masquage. Figure reprise de [FZ06]

transition se situe juste au centre d'une trame et l'erreur de codage est étalée temporellement sur toute la trame. Nous allons alors être en présence de post-masquage, puisque la moitié de la trame représente une durée ne dépassant pas quelques dizaines de millisecondes (étant donné les longueurs de trames utilisées en codage audio) comme on peut le voir sur la figure 3.6. Par contre, si l'on se situe dans la situation inverse (c'est-à-dire une augmentation de la puissance), le phénomène de pré-masquage étant plus limité, l'étalement de l'erreur sur la partie de la trame de plus faible puissance peut ne plus être inaudible. Quand ce bruit est audible on parle de *pré-écho*, qui doit soigneusement être pris en compte par les codeurs audio, sous peine d'entendre potentiellement du bruit avant chaque attaque de note. Afin d'empêcher ce phénomène, on peut par exemple contraindre l'augmentation du seuil de masquage, c'est-à-dire empêcher une trop brusque variation de ses valeurs d'une trame à une autre.

3.3 Analyse Temps-Fréquence

3.3.1 Introduction

Le principe général de l'analyse temps-fréquence est d'analyser le contenu fréquentiel du signal à une échelle temporelle locale où il peut être considéré comme étant stationnaire. La durée d'une trame est donc prise au maximum de quelques dizaines de millisecondes, comme il a déjà été mentionné. De plus cette taille est généralement cohérente avec la taille qui est utilisée pour l'analyse psychoacoustique détaillée précédemment, qui peut utiliser comme point de départ la même transformée temps-fréquence. Nous allons présenter ici

deux transformées temps-fréquence incontournables lorsque l'on étudie les codeurs audio perceptuels. Dans cette partie, pour tout vecteur $\mathbf{u}$ de longueur N nous notons $u(n)$ ses coefficients :

$$\mathbf{u} = (u(0), \ldots, u(N-1))^T \tag{3.8}$$

3.3.2 TFD

La *Transformée de Fourier Discrète* (TFD) est la transformée temps-fréquence (TF) usuelle dans un grand nombre d'applications audio (et dans d'autres domaines bien sûr), en particulier en analyse spectrale. Elle est dérivée directement de la transformée de Fourier : elle est la version discrète en fréquences de la transformée de Fourier à Temps Discret (TFDT), et elle est souvent utilisée avec du recouvrement et du fenêtrage. Plus particulièrement le fenêtrage est utilisé pour amoindrir le phénomène de Gibbs afin d'avoir une bonne analyse fréquentielle de la trame traitée, et le recouvrement est utilisé pour avoir une synthèse de bonne qualité, car l'absence de recouvrement entraîne généralement des artefacts audibles (notamment des clics) lorsque l'on modifie la TFD avant la synthèse (on parle aussi d'effet de bloc) [OS75, RS78].

Si $\mathbf{x}_t$ est une trame de longueur N, la définition de la transformée de Fourier discrète $\mathbf{X}_t$ de cette trame $\mathbf{x}_t$ est la suivante :

$$\forall k \in [0, N-1],\ X_t(k) = \sum_{n=0}^{N-1} x_t(n) e^{-j\frac{2\pi kn}{N}} \tag{3.9}$$

Cette transformée est inversible, et son inverse est donnée par :

$$\forall n \in [0, N-1],\ x_t(n) = \frac{1}{N} \sum_{k=0}^{N-1} X_t(k) e^{j\frac{2\pi kn}{N}} \tag{3.10}$$

Les facteurs donnés ici devant la TFD et son inverse (ici respectivement 1 et $1/N$) sont ceux usuellement utilisés. Cependant ils sont parfois modifiés, par exemple pour avoir une matrice de la TFD qui soit unitaire (il faut alors prendre des facteurs de $1/\sqrt{N}$ pour les deux expressions, directe et inverse).

Bien qu'elle soit la transformée TF usuelle, la TFD est rarement utilisée directement en compression pour obtenir les coefficients que l'on va quantifier, et ce pour deux raisons. La première est que ses coefficients complexes sont peu pratiques à manipuler du point de vue de la quantification, que ce soit dans une représentation module / phase ou dans une représentation partie réelle / partie imaginaire. La seconde raison, et probablement la plus importante des deux, est que comme nous l'avons énoncé dans le paragraphe précédent, s'il n'y a pas de recouvrement des trames adjacentes la quantification des coefficients fréquentiels produit souvent un effet de blocs (facilement audible, généralement par des clics sonores) lors de la reconstruction du signal au décodeur. Et si l'on effectue un recouvrement pour pallier ce problème, on a une représentation temps-fréquence qui a plus de coefficients réels que la

représentation temporelle, ce qui est contre productif et sous-optimal lorsque l'on veut faire de la compression. Nous reverrons certains aspects de ces considérations dans la section 5.2. Il faut néanmoins retenir que cette transformée reste à l'origine de toutes les autres transformées TF, du point de vue théorique mais aussi parfois au niveau de l'implémentation du fait de l'existence d'algorithmes de calcul de la TFD très efficaces en terme de vitesse de calcul. L'ensemble de ces algorithmes rapides de calcul de la TFD sont appelés communément algorithmes de transformée de Fourier rapide (TFR) ou FFT (pour l'anglais *Fast Fourier Transform*). Par exemple la manière la plus rapide connue actuellement pour générer la MDCT que nous allons développer ci-après est de décomposer cette transformée en plusieurs TFD.

3.3.3 MDCT

Présentation

La *Modified Discrete Cosine Transform* (MDCT) est une transformée réelle discrète avec 50% de recouvrement qui est aujourd'hui couramment utilisée dans le traitement du signal audionumérique et particulièrement dans le domaine de la compression audio. Elle est le fruit de recherches sur une propriété spécifique désirable pour les transformées temps-fréquence à vocation de codage, le *Time Domain Aliasing Cancellation* (TDAC) [PB86] (nous reviendrons sur cette propriété dans la suite de cette section). Cette transformée apparaît pour la première fois dans [PJB87] (bien que le nom de MDCT ne soit alors pas encore utilisé).

Soit $\mathbf{x}_t$ une trame de longueur N du signal $\mathbf{x}$. Puisque le recouvrement est de 50%, on a :

$$\mathbf{x}_t = \begin{pmatrix} x_t(0) \\ \vdots \\ x_t(N-1) \end{pmatrix} = \begin{pmatrix} x(0+tN/2) \\ \vdots \\ x(N-1+tN/2) \end{pmatrix} \tag{3.11}$$

Si l'on note en plus $\mathbf{h_a}$ la fenêtre d'analyse, la définition des coefficients MDCT de $\mathbf{x}_t$ est alors :

$$\forall k \in \left[0, \frac{N}{2}-1\right], X_t(k) = \frac{2}{\sqrt{N}} \sum_{n=0}^{N-1} x_t(n) h_a(n) \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{3.12}$$

Si l'on note de plus $\mathbf{h_s}$ la fenêtre de synthèse, la définition des coefficients de la transformée MDCT inverse (IMDCT) $\mathbf{y}_t$ est alors :

$$\forall n \in [0, N-1], y_t(n) = \frac{2}{\sqrt{N}} h_s(n) \sum_{k=0}^{\frac{N}{2}-1} X_t(k) \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{3.13}$$

Afin de garantir une reconstruction parfaite après recouvrement et addition (OLA pour *overlap-add* en anglais), les fenêtres d'analyse $\mathbf{h_a}$ et de synthèse $\mathbf{h_s}$ doivent satisfaire les

conditions de Princen-Bradley :

$$\forall n \in \left[0, \frac{N}{2}-1\right] \begin{cases} h_s(n)h_a(n) + h_s\left(n+\frac{N}{2}\right)h_a\left(n+\frac{N}{2}\right) = 1 \\ h_s(n)h_a\left(\frac{N}{2}-1-n\right) = h_s\left(n+\frac{N}{2}\right)h_a(N-1-n) \end{cases} \tag{3.14}$$

Dans la grande majorité des cas où l'on utilise la MDCT, les fenêtres d'analyse et de synthèse sont identiques :

$$\mathbf{h_s} = \mathbf{h_a} = \mathbf{h}, \tag{3.15}$$

et symétriques :

$$\forall n \in [0, N-1],\ h(N-1-n) = h(n) \tag{3.16}$$

Dans ce cas les conditions de Princen-Bradley se réduisent à :

$$\forall n \in \left[0, \frac{N}{2}-1\right],\ h^2(n) + h^2\left(n+\frac{N}{2}\right) = 1 \tag{3.17}$$

Les fenêtres les plus couramment utilisées satisfaisant ces conditions sont la fenêtre dite sinusoïdale et les fenêtres *Kaiser-Bessel Derived* (KBD). Les calculs relatifs à la MDCT (notamment la propriété de reconstruction parfaite) se situent en annexe A.

3.4 Quantification et codage entropique

Une fois la représentation temps-fréquence calculée, on peut distinguer deux grandes étapes dans la fin du processus de codage. La première est le choix des quantificateurs et la quantification de la représentation temps-fréquence, la seconde est le codage des valeurs quantifiées et des paramètres des quantificateurs utilisés avec un codage entropique.

Le type de quantificateur doit être choisi pour être le plus compatible possible avec les signaux audio. De nombreuses recherches montrent que les échelles les plus adaptées pour étudier les valeurs des coefficients temps-fréquence sont logarithmiques (ou très proches d'une loi logarithmique). La plupart des codeurs audio perceptuels utilisent donc soit des quantificateurs logarithmiques sur les coefficients temps-fréquence, soit des quantificateurs uniformes sur le logarithme (ou une fonction approchant le logarithme) de ces coefficients. Parmi la famille de quantificateurs disponibles, on choisit ceux dont la résolution est la plus faible possible (afin d'avoir une réduction d'information la plus grande possible) tout en s'assurant que l'erreur de codage reste inférieure au seuil de masquage résultant de l'analyse psychoacoustique [PS00].

Une fois le choix des quantificateurs effectué, il reste à coder ces informations afin de les insérer dans le flux binaire. La quantification est la phase de compression avec perte, et afin de maximiser le gain de débit, les codeurs perceptuels utilisent généralement une compression

sans perte additionnelle. Celle-ci est généralement réalisée grâce à des techniques de codage entropique, comme le codage de Huffman [CT06]. Le principe de ces techniques est de passer de codes de longueur fixe (c'est-à-dire où chaque mot est codé sur le même nombre de bits) à des codes de longueur variable, les mots les plus fréquents étant codés sur un nombre de bits plus faibles.

Il faut cependant faire très attention et ne pas croire que ces deux phases sont simples ou rapides, car cela dépend fortement des contraintes posées au codeur. En effet, la véritable contrainte que l'on pose généralement au codeur est une contrainte de débit. Dans le cas d'implémentation le plus strict, on dispose donc d'un nombre de bits fixé par trames (par exemple cette configuration est appelée CBR pour *Constant Bit Rate* en anglais dans le cas du AAC de MPEG). Ce type de réglage est généralement utilisé dans des cas de transmissions à débit fixe, par exemple pour la radio. Dans des cas plus complexes, on peut relâcher légèrement la contrainte, en répartissant les bits dans les trames suivant que l'une est plus psychoacoustiquement sensible qu'une autre (configuration ABR pour *Average Bit Rate* de AAC), voire relâcher complètement la contrainte (configuration VBR pour *Variable Bit Rate* de AAC). Quelle que soit la configuration choisie, le problème est alors un problème d'optimisation sous contrainte extrêmement complexe. En effet il faut adapter le choix des quantificateurs à la contrainte de débit en respectant au mieux le seuil de masquage du modèle psychoacoustique ; mais il faut le faire en tenant compte du codage entropique, dont on ne peut pas réellement prédire le taux de compression exact. Cette optimisation, appelée généralement processus d'allocation de bits, est donc généralement réalisée par des itérations du type analyse-synthèse, c'est-à-dire une recherche itérative de la configuration permettant de respecter la contrainte de débit et de suivre au mieux la contrainte liée au seuil de masquage. La mise en place de ce genre de techniques est très dépendante du codeur et ne sera donc pas discutée plus en détails. De plus, en ce qui concerne les technologies normalisées, si le flux binaire et le processus de décodage sont fixés dans les normes, le codeur est souvent plus libre et l'implémentation de telles boucles d'optimisation est laissée à l'initiative des concepteurs.

Le codage entropique est de moindre importance pour notre système de tatouage, puisqu'il n'y a pas de changement de format et donc de création de flux binaire dans ce cas. L'étape de tatouage dans laquelle le codage entropique peut être utile est le codage source, c'est-à-dire la mise en forme du message à transmettre $\mathbf{m}$ en un code $\mathbf{c}$ (voir figure 2.1). Cependant le type de compression sans perte à mettre en œuvre peut varier suivant les types de messages si l'on veut avoir une compression optimale et il dépend donc des données. Nous ne traiterons donc pas ce codage source en détail. Nous considérons que pour être optimal il doit être réalisé par l'utilisateur du système de tatouage en tenant compte de la statistique des données à tatouer.

3.5 Mesures de la qualité audio

3.5.1 Introduction

Dès l'apparition du codage audio perceptuel, il a fallu mesurer les performances des systèmes afin de pouvoir les améliorer et les comparer entre eux ou face à des bornes théoriques. Les deux principaux critères de performance pour un codeur audio sont bien sûr le débit, ou taux de compression, et la qualité audio. Le premier est assez évident et facile à calculer, cependant le second étant un critère subjectif il est beaucoup plus difficile de le quantifier. En effet, si l'on disposait de modèles psychoacoustiques parfaits, ceux-ci seraient utilisés directement dans l'étape d'analyse psychoacoustique des codeurs audio et la qualité audio désirée pourrait être choisie parfaitement. Or nous savons que les modèles psychoacoustiques utilisés, même les plus sophistiqués, ne sont pas parfaits et il faut donc mesurer la qualité des signaux codés. Le premier type de méthode est la mesure subjective de la qualité par des tests d'écoute. Deux protocoles normalisés par l'Union Internationale des Télécommunications (UIT ou ITU en anglais) seront présentés ici, un pour les signaux de très bonne qualité [IR97], et un pour les signaux de qualité intermédiaire [IR03]. L'avantage de ces méthodes est que si le nombre de participants aux tests d'écoute est assez élevé, les résultats sont normalement assez significatifs. Son inconvénient est qu'il est très lourd à mettre en place et demande des choix cruciaux, comme le choix des signaux test. Le second type de méthodes est la mesure objective de la qualité. L'idée est d'avoir un ensemble d'algorithmes qui permettrait de calculer automatiquement la perte de qualité entre un signal original et un signal codé. Nous présenterons un algorithme développé particulièrement pour le codage audio, l'algorithme PEAQ [IR01]. L'avantage de ce type de méthodes est qu'elles sont beaucoup plus rapides et pratiques à mettre en place que les méthodes subjectives, et peuvent être utilisées sur une plus grande base de données. L'inconvénient est qu'elles sont moins précises.

3.5.2 Mesures subjectives

Comme nous l'avons dit précédemment, le principe de ce type de mesures est de sélectionner quelques signaux test, d'une dizaine de secondes maximum, et de les faire évaluer par des auditeurs. Plus précisément les auditeurs vont attribuer soit une note comparative entre le signal codé et le signal original soit une note absolue au signal codé. Nous allons tout d'abord présenter le protocole MUSHRA [IR03], utilisé pour les signaux de qualité intermédiaire, puis nous allons présenter le protocole recommandé par l'UIT pour les signaux de très haute qualité [IR97].

Signaux de moyenne qualité

Le protocole proposé par l'UIT pour traiter les signaux de moyenne qualité (c'est-à-dire que l'on pourra facilement distinguer de l'original) est basé sur une méthode de test dite « doublement aveugle à stimuli multiples » (aussi appelé test MUSHRA pour *MUltiple*

Stimuli with Hidden Reference and Anchor en anglais), dont le principe est le suivant. On dispose d'un stimulus original, appelé référence, de n objets (qui sont des versions modifiées de ce stimulus pour lesquelles on souhaite avoir une mesure de la dégradation), ainsi que d'un signal de repère (*anchor* en anglais), qui est une version très dégradée de la référence (la recommandation propose un filtrage passe-bas à 3.5 kHz de la référence pour obtenir le repère). Le sujet a alors accès à un signal X, dont il sait que c'est la référence, et $n+2$ autres signaux, pour lesquels il ne connaît que les informations suivantes : parmi ces $n+2$ autres signaux se trouvent les n objets, le repère et à nouveau la référence. Le sujet doit alors évaluer la qualité en donnant une note suivant l'échelle de qualité continue (CQS pour *Continuous Quality Scale* en anglais) dont chacun des intervalles caractérise un niveau de qualité particulier : excellent, bon, assez bon, médiocre et mauvais. On leur assigne généralement une graduation allant de 0 à 100 ou de 0 à 5 pour permettre au sujet de se repérer plus facilement.

Le test d'écoute en lui-même se décompose en deux phases, une phase d'entraînement et une phase de notation. Lors de la phase d'entraînement, on fournit au sujet tous les signaux qui seront utilisés dans la phase de notation, sans lui donner aucune information sur ces signaux, afin qu'il se familiarise avec les différents types de dégradation qu'il peut rencontrer lors du test. La phase de notation quant à elle est décomposée en différents essais, chaque essai étant une réalisation d'un test de type MUSHRA, basé sur un signal de référence différent. La recommandation de l'UIT préconise un nombre d'essais pour chaque participant d'environ une fois et demi le nombre de configurations à traiter (n précédemment), avec un minimum de 5 essais, sachant que trop d'essais peuvent conduire à une fatigue et perte de concentration du sujet, ce qui peut entraîner des résultats moins pertinents qu'avec un nombre d'essais moindre.

Signaux de haute qualité

Le protocole proposé par l'UIT pour mesurer les modifications de faibles importances sur des signaux audio est un test d'écoute basé sur la méthode dite « doublement aveugle à triple stimuli et référence dissimulée » (que nous appellerons test ABC), dont le principe est le suivant. On dispose d'un stimulus original, appelé référence, et d'une version modifiée de ce stimulus, appelé objet. On souhaite alors faire évaluer subjectivement par un sujet la dégradation relative entre la référence et l'objet. Pour réaliser ceci, le sujet a accès à trois signaux, A, B et C. Il sait que le signal A est le signal de référence, et que parmi les signaux B et C, l'un est aussi la référence et l'autre l'objet, de manière aléatoire (*i.e.* B peut parfois être la référence, parfois l'objet). Le sujet va alors noter le signal B et le signal C comparativement au signal A, suivant la grille donnée dans la table 3.2. La différence entre la note que le sujet a donnée à l'objet et celle qu'il a donnée à la référence est alors calculée, et c'est cette note appelée SDG pour *Subjective Difference Grade* en anglais qui va être utilisée pour caractériser la dégradation perçue de l'objet par rapport à la référence.

Comme pour le test MUSHRA, le test ABC est généralement divisé en deux phases, une phase d'apprentissage et une phase de notation. La phase d'apprentissage est généralement

Niveau de dégradation	Note	ODG
Imperceptible	5,0	0,0
Perceptible mais non gênant	4,0	-1,0
Légèrement gênant	3,0	-2,0
Gênant	2,0	-3,0
Très gênant	1,0	-4,0

TABLE 3.2 – Signification des notes et de l'ODG en terme de niveau de dégradations

similaire à la phase de notation, à ceci près que les résultats ne sont pas enregistrés et les stimuli sont différents de ceux utilisés pour la phase de notation. Le but principal de la phase d'apprentissage est de rendre le sujet le plus familier possible avec les types de dégradations qu'il devra chercher à détecter lors de la phase de notation. La phase de notation se décompose en plusieurs essais, chaque essai étant une réalisation d'un test ABC. Il est important de noter ici que le sujet possède un accès non restreint aux trois signaux A, B et C, c'est-à-dire qu'il peut jouer chacun d'entre eux autant de fois qu'il le souhaite et dans n'importe quel ordre.

Ce type de test nécessite des sujets avec une bonne ouïe, et la recommandation de l'UIT propose des tests de post-sélection afin de valider ou non les sujets du test. Plus précisément, il est suggéré que pour chaque participant un test t unilatéral [2] soit effectué. Si l'hypothèse nulle est confirmée, alors généralement le sujet n'a pas correctement réussi à distinguer la référence de l'objet et il est éliminé. Si l'hypothèse est rejetée, c'est-à-dire que la moyenne est bien négative, alors on considère que le sujet a plutôt bien réussi à discerner les modifications et ses résultats sont conservés.

Afin d'avoir des résultats statistiquement représentatifs, la recommandation préconise une vingtaine de sujets, et pas plus d'une dizaine d'essais (de durées de 10 à 20 secondes) pour que la phase de notation ne dure pas trop longtemps.

3.5.3 Mesure objective : algorithme PEAQ

L'algorithme PEAQ, normalisé par l'UIT [IR01], est la synthèse de 6 méthodes développées dans les années 1990 dont l'objectif est de fournir une mesure objective de la qualité d'un signal après compression par un codeur audio perceptuel. Les six méthodes dont est issu l'algorithme sont :

- l'indice de perturbation DIX (*Distortion IndeX*) [TK96],
- le rapport bruit à masque NMR (*Noise-to-Mask Ratio*) [Bra87a],
- le système de mesure OASE (*Objective Audio Signal Evaluation*) [Spo97],

2. Un test de Student ou test t est souvent utilisé pour tester de façon statistique si une hypothèse de moyenne nulle pour une variable aléatoire est statistiquement cohérente. Voir par exemple [DD08] ou sur MathWorld™en cliquant ici

- la mesure perceptuelle de la qualité du son PAQM (*Perceptual Audio Quality Measure*) [BS94],
- le système PERCEVAL (*PERCeptual EVALuation of the quality of audio signals*) [PMMS92],
- la mesure perceptuelle objective POM (*Perceptual Objective Measurement*) [Col94],

ainsi que la *Toolbox Approach*. Le but de cet algorithme est de fournir une mesure objective de différence entre un signal de référence et un signal test (le signal de référence après compression par un codeur audio perceptuel), l'ODG (pour *Objective Difference Grade* en anglais) variant entre 0 et -4 suivant la dégradation comme indiqué table 3.2, et correspondant aux niveaux de dégradation utilisés pour les tests subjectifs vus précédemment. En vue de sa normalisation, plusieurs façons de combiner les différentes méthodes ont été testées sur de larges bases de données (par des organismes avec ou sans lien avec la normalisation de PEAQ), en les comparant à des SDG données par des sujets lors de tests d'écoute. La meilleure combinaison possible a été gardée pour donner l'algorithme PEAQ. Nous n'entrerons pas dans les détails ici, cependant il est important de savoir que cet algorithme a été configuré sur des bases de données de signaux compressés par des codeurs audio, et n'est donc originellement adapté qu'à ce type de signaux. L'utilisation sur d'autres types de signaux doit être vérifiée sur une base de données avant généralisation.

Notons que PEAQ n'est pas la seule méthode pour évaluer objectivement la qualité de signaux audio ; et nous pouvons notamment citer l'algorithme PEMO-Q [HK06] développé récemment et dont l'utilisation n'est pas restreinte aux signaux compressés.

Deuxième partie

1ère implémentation basique

Chapitre 4

Principes

Sommaire

4.1 Spécifications du système de tatouage

Dans cette partie, nous allons présenter une première implémentation de notre système de tatouage. Cette version basique ne contient que les éléments strictement nécessaires à son bon fonctionnement.

L'objectif est ici de développer un système de tatouage pour une utilisation dans le cadre du projet DReaM. Rappelons que le but du projet DReaM est d'offrir un service à l'utilisateur en lui permettant de remixer au moins partiellement un morceau de musique dans un format numérique non compressé (typiquement PCM-16 bits, 44.1kHz), qui sera par conséquent le format du signal hôte tatoué. Le tatouage est utilisé ici pour véhiculer des informations à un logiciel DReaM afin de permettre la séparation des sources qui composent le mix (instruments de musique, voix…). La première spécification est donc la suivante :

1. Le système de tatouage développé n'est pas sécuritaire et n'a donc aucune vocation à être résistant à des attaques volontaires autres que celles liées au format de stockage du mix tatoué. En particulier, ce format étant non-compressé, le tatouage n'a pas vocation à être robuste à une compression audio avec pertes.

De plus, l'information extraite doit être extrêmement fidèle à celle qui a été insérée. En effet si l'information transmise est erronée, la qualité du service à l'utilisateur risque d'être fortement amoindrie. La seconde spécification est donc :

2. Le système de tatouage doit permettre un taux d'erreur très faible.

Le projet DReaM souhaitant de plus permettre une réelle interaction entre l'utilisateur et le morceau de musique, il est alors souhaitable que cette interaction se fasse en temps-réel [1]. La troisième spécification est donc :

3. Le système de tatouage doit pouvoir permettre un décodage en temps-réel.

En outre, il peut être intéressant que l'information insérée grâce au système de tatouage puisse être décodée partiellement suivant les besoins, typiquement lorsque l'utilisateur décide de se placer à un moment quelconque du morceau. Pour cela, il faut que l'information tatouée soit fragmentée et insérée indépendamment dans des trames de quelques dizaines de millisecondes de durée, sur lesquelles un logiciel DReaM pourra se synchroniser [2] afin que l'utilisateur puisse se déplacer au moment de son choix dans un morceau, comme dans n'importe quel lecteur audio classique. La quatrième spécification est donc :

4. L'information insérée par le système de tatouage doit être fragmentée par trames de courte durée et chaque trame doit pouvoir être décodée séparément.

Finalement, plusieurs techniques de séparation de sources informée ayant été développées dans le cadre du projet DReaM (et dont l'étude indépendante ou comparative est totalement hors du cadre de cette thèse [LGS+12, MBB+12]), il faut permettre au système de tatouage

1. C'est-à-dire en réalité avec une latence très faible de quelques dizaines de millisecondes, quasiment imperceptible et peu gênante pour ce type d'application.

2. Dans ce chapitre on ne traitera que du découpage en trames. Les aspects de synchronisation seront traités dans la partie III.

de transmettre les paramètres de n'importe quelle technique de séparation. Chacune ayant besoin d'informations différentes et surtout de débits différents (pouvant varier de quelques dizaines de kb/s à quelques centaines de kb/s), la cinquième spécification est donc :

5. Le système de tatouage doit permettre d'insérer des données à des débits différents suivant l'application considérée, avec des valeurs allant de quelques dizaines de kb/s à quelques centaines de kb/s.

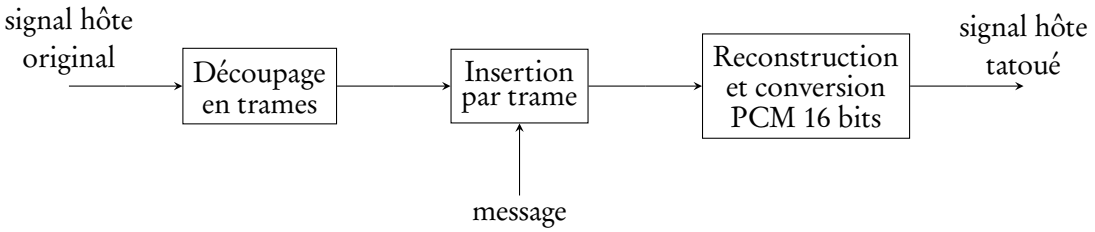
Ce débit est à comparer au débit du signal hôte, soit 705.6 kbits/s par voie pour le PCM 16 bits à 44.1 kHz. Le débit de tatouage pourra ainsi représenter une fraction non négligeable du débit du signal hôte.

4.2 Principes généraux

Comme il a été dit précédemment, le point de départ du système développé est l'adaptation des principes des codeurs audio perceptuels décrits dans le chapitre 3 à un système de tatouage. L'adaptation théorique à un système de tatouage va donc être basée sur un codeur, au sein duquel l'insertion sera réalisée, et un décodeur aveugle (au sens décrit en section 2.2), qui permettra de récupérer l'information insérée grâce au codeur. De façon similaire au codage perceptuel, l'insertion et le décodage seront réalisés par trames de courte durée (inférieure à 100ms). Un premier schéma de principe du codeur et du décodeur est représenté figure 4.1. Au codeur, un découpage du signal hôte en trames est tout d'abord réalisé, puis chaque trame est tatouée. Ensuite le signal hôte tatoué est reconstruit à partir de toutes les trames tatouées avant d'être reconverti au format PCM 16 bits, ce qui se traduit ici principalement par une quantification scalaire uniforme sur 16 bits des échantillons temporels du signal (format du CD-A). Au décodeur, le signal hôte tatoué est de nouveau découpé en trames, puis l'information tatouée est extraite de chacune de ces trames. Les informations décodées dans chacune des trames peuvent alors être regroupées si nécessaire pour reconstituer le message global.

Les deux blocs fondamentaux de notre système de tatouage sont alors les blocs d'insertion et d'extraction par trame. Le schéma de principe de ces deux blocs est représenté figure 4.2, et leur fonctionnement peut être décrit ainsi :

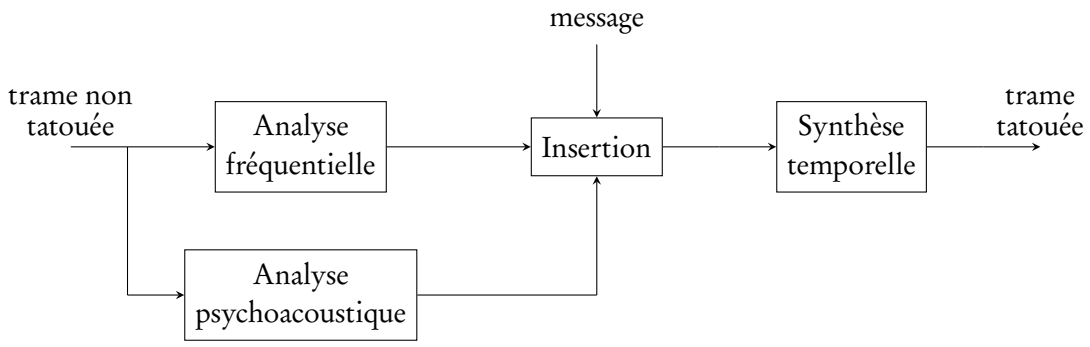
- Pour l'insertion d'une trame :
  - transformation dans le domaine fréquentiel (on parle d'analyse fréquentielle)
  - analyse psychoacoustique de la trame basée sur un modèle psychoacoustique
  - quantification de la représentation temps-fréquence, non pas pour réduire la quantité d'information à transmettre comme dans le cas du codage audio mais pour insérer de l'information. Ce choix d'une technique de tatouage par quantification (par opposition à une technique d'étalement de spectre par exemple) est déterminé par la nécessité d'une insertion à haut débit, conjugué avec une absence de contraintes fortes sur la robustesse.
  - retour dans le domaine temporel par transformation fréquentielle inverse (on parle de synthèse temporelle)



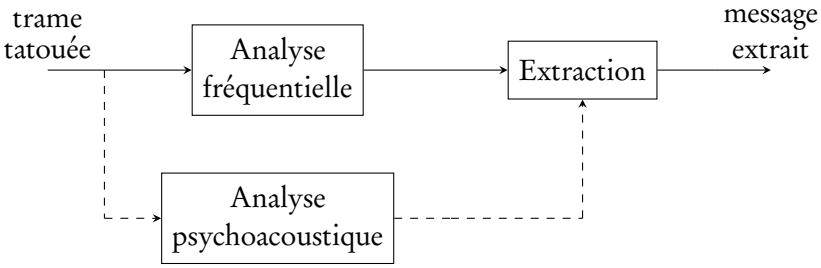
(a) Codeur

(b) Décodeur

FIGURE 4.1 – Schémas des codeurs et décodeurs



(a) Bloc d'insertion



(b) Bloc d'extraction

FIGURE 4.2 – Schéma des blocs d'insertion et d'extraction par trame

- Pour l'extraction :
  - analyse fréquentielle
  - extraction de l'information tatouée

Notons que dans le cas du codage audio perceptuel, les caractéristiques des quantificateurs (nombre de bits et facteurs d'échelle) sont codées dans le flux binaire du format compressé. Or, dans le cas de notre système de tatouage, il n'y a pas de tel flux binaire étant donné que le signal audio est converti au format PCM 16 bits. Dans notre cas il faut donc transmettre les paramètres des quantificateurs ou les recalculer dans le bloc d'extraction (cf. figure 4.2b), l'étape d'analyse psychoacoustique au décodeur n'étant nécessaire que si l'on doit recalculer les paramètres. Nous avons fait le choix de transmettre ces paramètres et non de les recalculer, pour plusieurs raisons. La principale est due au fonctionnement du modèle utilisé pour l'analyse psychoacoustique, qui donne le seuil de masquage à partir duquel sont calculés les paramètres d'insertion. En effet, puisque l'on cherche à insérer généralement une quantité d'information très importante dans le signal hôte, les différences entre le signal original et le signal tatoué seront importantes (même si elles ne sont pas audibles). En particulier le spectre pourra être notablement modifié par le tatouage ce qui aura comme conséquence que le seuil de masquage (et donc les paramètres qui en découlent) calculé à l'extraction sera différent de celui calculé à l'émission, entraînant un décodage erroné. La seconde raison pour cette transmission des paramètres est que cela permettra d'économiser un temps précieux lors de l'extraction puisque l'analyse psychoacoustique n'aura pas à être effectuée, ce qui peut être important si l'on cherche à utiliser le système pour des applications temps-réel.

Nous espérons alors que le système de tatouage ainsi développé pourra assurer les spécifications demandées. En effet, le caractère non sécuritaire et le taux d'erreur faible peuvent être réglés à la fois par le choix de la technique de tatouage par quantification et par le réglage des paramètres de cette technique. Le découpage du signal en trames permet bien quant à lui de fragmenter l'information et d'assurer ainsi l'accès « local » aux données tatouées ainsi que le décodage temps-réel, à condition que le décodage du tatouage soit assez rapide et donc pas trop complexe. Finalement, même si l'on n'atteint pas un débit de tatouage équivalent au gain obtenu par compression, qui dépasse parfois le Mo/s pour des signaux stéréophoniques, il semble possible d'atteindre des débits de l'ordre de la centaine de kbits/s en basant le tatouage sur une quantification dans le domaine fréquentiel.

Chapitre 5

Présentation détaillée

Sommaire

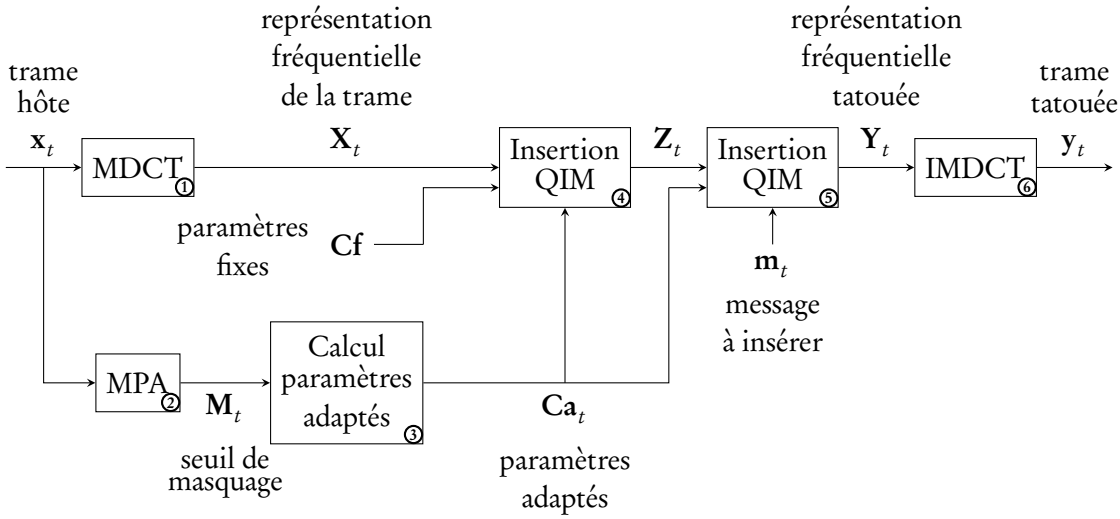
5.1 Vue d'ensemble du système

Nous avons vu dans le chapitre précédent le principe général de notre système de tatouage et nous allons maintenant le présenter plus en détail. Les schémas des codeurs et décodeurs sont identiques à ceux présentés en figure 4.1, autrement dit les traitements sont bien effectués trame par trame. Un schéma détaillé des blocs d'insertion et d'extraction qui décrit les processus effectués pour chaque trame est représenté figure 5.1.

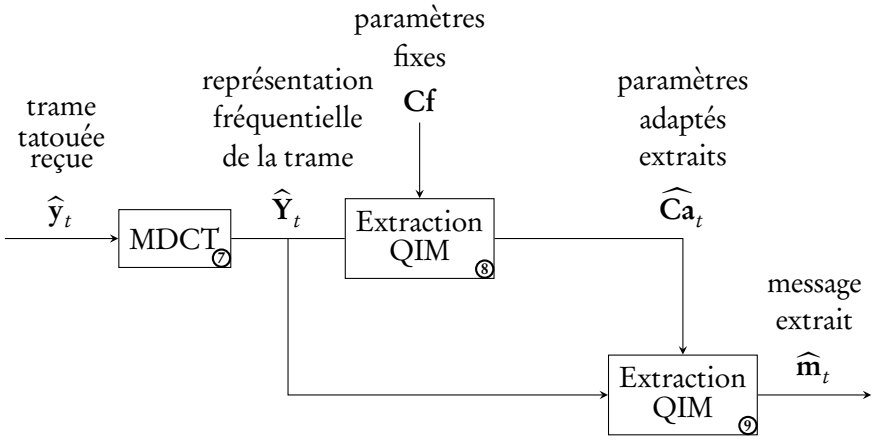
Lors de l'étape d'insertion d'une trame, une représentation fréquentielle de la trame est tout d'abord calculée (bloc ①), la transformée utilisée étant la MDCT. Nous verrons dans la suite de ce chapitre les raisons qui nous ont amené à choisir cette transformée pour notre système de tatouage (celles-ci sont en partie liées aux raisons qui conduisent beaucoup de codeurs audio perceptuels à l'utiliser, et en partie spécifiques au tatouage). Un seuil de masquage adapté à la trame est ensuite calculé grâce à un modèle psychoacoustique (bloc ②). Le modèle utilisé sera lui aussi détaillé dans ce chapitre, en notant tout de même que n'importe quel modèle qui fournit un seuil de masquage peut le remplacer. Suivant ce seuil de masquage, les paramètres des quantificateurs qui vont servir à l'insertion sont calculés (bloc ③) ; la nature exacte de ces paramètres sera discutée par la suite. Lors du chapitre précédent, nous avons expliqué que ces paramètres allaient être recalculés. Ces paramètres calculés sont donc tatoués dans lors d'une première étape d'insertion à paramètres fixes (bloc ④, ces paramètres fixes étant connus au décodeur), et nous choisissons d'effectuer cette première insertion dans les hautes-fréquences (HF). Le message est alors tatoué lors d'une deuxième étape d'insertion, cette fois-ci avec les paramètres adaptés calculés précédemment grâce au modèle psychoacoustique (bloc ⑤) dans la bande spectrale restante (basse fréquences, BF). Ces deux étapes d'insertion utilisent la technique de QIM (*Quantization Index Modulation*) scalaire, comme décrite en section 2.4.2. La représentation temps-fréquence de la trame tatouée est finalement transformée dans le domaine temporel grâce à l'inverse de la MDCT, l'IMDCT (bloc ⑥).

L'étape d'extraction d'une trame tatouée se déduit logiquement de l'étape d'insertion d'une trame. La trame est de nouveau transformée dans le domaine fréquentiel grâce à la MDCT (bloc ⑦). Une première extraction avec les paramètres fixes est effectuée afin de récupérer les paramètres des quantificateurs adaptés à la trame (bloc ⑧). Une fois les paramètres adaptés extraits, ils sont utilisés pour une deuxième extraction afin de récupérer le message inséré (bloc ⑨).

Ce chapitre est organisé comme suit. Nous présenterons tout d'abord en section 5.2 la transformée temps-fréquence, et plus spécifiquement pourquoi nous avons choisi la MDCT. La section 5.3 sera consacrée à la présentation du modèle psychoacoustique utilisé actuellement dans notre système, qui fournit le seuil de masquage à partir duquel sont calculés les paramètres adaptés pour l'insertion. Dans la section 5.4 nous expliquerons comment la technique de QIM est utilisée dans notre système. Finalement nous discuterons en section 5.5 des moyens mis en œuvre pour calculer les paramètres.



(a) Insertion



(b) Extraction

FIGURE 5.1 – Schéma de l'insertion et de l'extraction par trame

5.2 Transformée temps-fréquence : choix de la MDCT

5.2.1 Introduction

Dans cette partie nous allons principalement expliquer les raisons du choix de la MDCT (dont la description technique a été donnée à la section 3.3.3) comme transformation temps-fréquence pour notre système de tatouage. Plus spécifiquement, nous allons présenter comment les deux contraintes majeures pour notre système, l'inaudibilité et le fait que ce soit un système de tatouage à très faible taux d'erreur conduisent au choix de la MDCT, puis quels sont les avantages secondaires de cette transformée.

5.2.2 Inaudibilité

Tout d'abord, il faut que la transformée temps-fréquence permette au système de tatouage de respecter au maximum la contrainte d'inaudibilité. Autrement dit, lorsqu'une très faible quantité d'information est tatouée, la représentation temporelle de l'hôte tatoué doit être quasiment identique à la représentation temporelle de l'hôte original. En poussant le raisonnement, il apparaît donc nécessaire que la transformée temps-fréquence utilisée soit à reconstruction parfaite. On note $\mathbf{x}$ la représentation temporelle d'un signal, de dimension M et $\mathbf{X}$ sa représentation temps-fréquence, de dimension L. On note de plus :

$$\begin{aligned} \mathscr{U} : \mathbb{R}^M &\longrightarrow \mathbb{R}^L \\ \mathbf{x} &\longmapsto \mathbf{X} \end{aligned} \tag{5.1}$$

l'application qui à une représentation temporelle associe un plan temps-fréquence, et :

$$\begin{aligned} \mathscr{V} : \mathbb{R}^L &\longrightarrow \mathbb{R}^M \\ \mathbf{X} &\longmapsto \mathbf{x} \end{aligned} \tag{5.2}$$

l'application qui à un plan temps-fréquence associe une représentation temporelle. Il est très important de distinguer deux choses : les transformées qui agissent de manière globale (qui lient représentations temporelles et plans temps-fréquence), et les transformées locales (au niveau de la trame). Ceci n'est pas sujet à confusion lorsque l'on utilise la TFD, puisque l'on distingue alors TFD (pour une trame) et STFT (pour le signal en entier). Cependant dans la littérature le terme MDCT désigne parfois la transformée locale ou la transformée globale. Les opérateurs $\mathscr{U}$ et $\mathscr{V}$ désignent ici quant à eux les opérateurs qui relient la représentation temporelle dans son ensemble et le plan temps-fréquence entier. Nous avons expliqué précédemment qu'il faut que notre transformée temps-fréquence soit à reconstruction parfaite, ce qui se traduit par :

$$\mathscr{V} \circ \mathscr{U} = I_M \tag{5.3}$$

où I_M désigne l'identité dans un espace de dimension M. Ces applications étant supposées linéaires, on peut en déduire la relation suivante :

$$L \geq M \tag{5.4}$$

c'est-à-dire que la dimension du plan temps-fréquence doit être supérieure à celle de la représentation temporelle. Cette caractéristique est présente dans beaucoup de transformées temps-fréquence classiques, par exemple lorsque l'on calcule un plan temps-fréquence grâce à des TFD avec recouvrement.

Rappelons aussi qu'il doit être possible d'insérer une information conséquente dans certaines trames du signal, ce qui peut modifier substantiellement leur représentation fréquentielle. Or, il est bien connu que concaténer des trames temporelles dont les représentations fréquentielles ont été modifiées peut rapidement conduire à des problèmes de continuités au niveau des raccords, ce qui se traduit par des clics perceptibles. Afin d'éviter ceci il faut donc que la transformée temps-fréquence choisie soit à recouvrement, avec des fenêtres d'analyse et de synthèse appropriées à la suppression de ce type de défaut.

La contrainte d'inaudibilité implique donc deux propriétés :

- la transformée doit être à recouvrement,
- l'enchaînement transformée directe puis transformée inverse doit être égal à l'identité ($\mathcal{V} \circ \mathcal{U} = I_M$, ce qui implique $L \geq M$). Nous notons Ⓐ cette propriété pour simplifier les développements à venir.

Cet ensemble de caractéristiques est classique en analyse-synthèse, et de nombreuses transformées à recouvrement possèdent ces caractéristiques, par exemple la STFT (*Short Time Fourier Transform*) avec recouvrement, c'est-à-dire quand on génère un plan temps-fréquence en calculant la TFD de chaque trame avec recouvrement [OS75, RS78].

5.2.3 Système de tatouage à faible taux d'erreur

Cette deuxième contrainte est très différente de la précédente, et vient du fait que l'on est dans le cas d'un système de tatouage. Lorsque l'on fait de l'analyse-synthèse classique, le cadre est le suivant. On a tout d'abord un signal temporel pour lequel on calcule un plan TF ; on modifie ensuite ce plan TF et l'on re-synthétise un signal temporel. On enchaîne donc toujours directement la transformée directe $\mathcal{U}$, puis la transformée inverse $\mathcal{V}$. Or, dans le cas de notre système de tatouage, le contexte est assez différent. On part bien là aussi d'un signal temporel, dont on calcule le plan temps-fréquence grâce à la transformée $\mathcal{U}$. On modifie ensuite le plan temps-fréquence obtenu (en le tatouant), puis on transforme ensuite le signal dans le domaine temporel. Jusqu'ici pas de différence notable avec l'analyse-synthèse classique. La différence majeure se situe dans le fait que ce signal tatoué va alors être re-transformé du domaine temporel vers le domaine temps-fréquence pour effectuer l'extraction. Si l'on souhaite un faible taux d'erreur, il faut donc qu'après tatouage des coefficients temps-fréquence, l'enchaînement transformée inverse $\mathcal{V}$ (pour repasser dans le domaine temporel) puis transformée directe $\mathcal{U}$ (pour repasser dans le domaine temps-fréquence afin d'extraire le tatouage) soit égal à l'identité :

$$\mathcal{U} \circ \mathcal{V} = I_L \tag{5.5}$$

De la même façon que précédemment, on en déduit la relation suivante :

$$M \geq L \tag{5.6}$$

c'est-à-dire que la dimension de la représentation temporelle doit être supérieure ou égale à la dimension de la représentation temps-fréquence.

Le fait que l'on soit dans le cas d'un système de tatouage avec un faible taux d'erreur implique donc une contrainte duale de la propriété Ⓐ :

- l'enchaînement transformée inverse puis transformée directe doit être égal à l'identité ($\mathscr{U} \circ \mathscr{V} = I_L$, ce qui implique $L \leq M$). Toujours pour simplifier les développements à venir nous notons cette propriété Ⓑ.

Cette contrainte est spécifique au tatouage et n'apparaît pas dans le cas du codage audio, et il est important de noter qu'elle n'est pas respectée par la plupart des transformées (notamment la STFT avec recouvrement).

5.2.4 Conséquences

Les deux ensembles de propriétés obtenus précédemment impliquent la propriété suivante, que nous noterons Ⓒ :

$$L = M \tag{5.7}$$

Autrement dit :

$$\text{Ⓐ et Ⓑ} \Rightarrow \text{Ⓒ} \tag{5.8}$$

Remarquons que les trois propriétés sont permutables, c'est-à-dire que l'on a aussi :

$$\text{Ⓐ et Ⓒ} \Rightarrow \text{Ⓑ} \tag{5.9}$$

$$\text{Ⓑ et Ⓒ} \Rightarrow \text{Ⓐ} \tag{5.10}$$

en utilisant de manière immédiate le théorème du rang (voir par exemple [Lan02] ou sur MathWorld™en cliquant ici). On doit donc chercher une transformée qui possède deux des trois propriétés Ⓐ, Ⓑ ou Ⓒ et qui soit à recouvrement, c'est-à-dire une transformée bijective à recouvrement. Or la MDCT a été développée pour satisfaire les contraintes suivantes :

- reconstruction parfaite,
- être à recouvrement,
- être à échantillonnage critique, c'est-à-dire avoir le même nombre de coefficients dans le plan temps-fréquence que dans le domaine temporel.

La première condition découle simplement du fait que la MDCT a été développée pour faire de l'analyse-synthèse et du codage audio, et la seconde pour éviter les effets de blocs lors de la synthèse. La troisième condition est liée directement au codage audio perceptuel. En effet, dans ce cadre on cherche à compresser l'information au maximum (voir chapitre 3). Avoir une transformée avec de la redondance est donc contre productif. Ces trois contraintes correspondent justement aux contraintes de recouvrement, Ⓐ et Ⓒ définies précédemment. L'utilisation de l'équation (5.9) permet alors d'affirmer que la contrainte Ⓑ, primordiale pour notre système de tatouage, est bien vérifiée dans le cas de la MDCT, et ceci même si elle n'a pas été développée spécifiquement dans ce but.

5.2.5 Récapitulatif

Nous avons donc vu que, pour notre système de tatouage, il était nécessaire que la transformée temps-fréquence utilisée ait trois propriétés :

- être à recouvrement,
- Ⓐ,
- Ⓑ.

Les propriétés de recouvrement et Ⓐ sont dues à la nécessité de réaliser une bonne synthèse temporelle, et sont présentes dans la quasi-totalité des transformées temps-fréquence. En effet ces transformées ont pour la plupart été développées pour l'analyse-synthèse ou le codage audio, et dans ces deux domaines une bonne synthèse temporelle est indispensable. La contrainte Ⓑ quant à elle est spécifique au tatouage, et n'entre pas en ligne de compte pour le développement des transformées temps-fréquence. Cependant un jeu d'implications permet de retomber sur la condition Ⓒ d'échantillonnage critique, très intéressante pour le codage et qui a conduit avec la propriété Ⓐ et le recouvrement au développement de la MDCT. La MDCT est donc la transformée qui répond le mieux aux contraintes liées à notre système de tatouage, et c'est pour ces raisons que nous l'utilisons dans notre système de tatouage.

5.2.6 Autres intérêts de la MDCT

Un autre avantage de cette transformée est que les coefficients de la MDCT sont réels, par opposition aux coefficients de la TFD par exemple, qui sont complexes. Plus précisément, étant donné que nous utilisons une technique d'insertion par quantification, ce sont ces coefficients temps-fréquence qui seront quantifiés. Ce n'est pas directement le fait que ces valeurs soient complexes qui pose problème. En effet, une valeur complexe peut être assimilée bijectivement à un couple partie réelle / partie imaginaire, ou presque bijectivement (bijectivement à l'exception de la valeur complexe zéro) à un couple module / phase. En section 2.4.2, nous avons vu qu'il est possible d'utiliser la technique de QIM sur des variables à plusieurs dimensions en utilisation des réseaux géométriques. Le problème vient alors de la définition de ces réseaux. En effet, si l'on sait que l'amplitude et la phase de la représentation fréquentielle ou temps-fréquence d'un signal ne sont pas du tout perçues de la même manière par le système auditif humain, il est très difficile de les comparer et donc de définir des quantificateurs vectoriels bien adaptés.

Finalement la MDCT, comme la plupart des transformées temps-fréquence, peut être définie pour n'importe quelle longueur de trame N multiple de 2 tant que N est au moins égal à 4. Ceci est intéressant pour deux raisons :

- ce paramètre N est bien entendu susceptible de faire varier les performances du système de tatouage, notamment du point de vue de la contrainte d'inaudibilité ;
- dans le cadre du projet DReaM, les techniques de séparation de sources informée développées jusqu'à maintenant reposent sur des décompositions temps-fréquence, et adapter la longueur des trames du système de tatouage à celle de la technique de séparation de sources utilisée pourra faciliter le traitement en temps-réel.

Au niveau des fenêtres utilisées pour la MDCT, nous utilisons des fenêtres de types KBD (*Kaiser-Bessel Derived*) [TRS+93], réglées afin d'être pertinentes d'un point de vue psychoacoustique [1] tout en ayant la même fenêtre symétrique à l'analyse et à la synthèse.

5.3 Analyse psychoacoustique

5.3.1 Introduction

L'analyse psychoacoustique effectuée dans le système de tatouage présenté est menée grâce à un modèle psychoacoustique (MPA), dont le but est de fournir un seuil de masquage. Plutôt que d'utiliser le code d'un modèle psychoacoustique déjà développé, nous avons fait le choix d'en ré-implémenter un nous-mêmes afin de nous assurer de son adaptabilité et de son bon fonctionnement dans le cadre assez particulier de notre système de tatouage. Le modèle psychoacoustique que nous allons présenter est inspiré de celui présenté dans la norme MPEG-AAC [ISO98]. La différence majeure par rapport au modèle original est le calcul des partitions suivant la longueur de trame N (puisque deux longueurs de trames uniquement sont utilisées dans AAC). Ce modèle est simple mais prend en compte deux phénomènes prépondérants dans le cadre de l'analyse psychoacoustique : le masquage fréquentiel (ou masquage simultané) et le masquage temporel (ou masquage non-simultané), comme décrits dans la section 3.2. Le modèle présenté ne sera donc pas optimisé par rapport à d'autres modèles disponibles, cependant nous pourrons nous assurer qu'il est bien compatible avec une utilisation dans le cadre de notre système de tatouage, et il pourra être facilement modifié si le besoin s'en fait sentir. Nous allons tout d'abord présenter le modèle et une partie de son implémentation sans rentrer dans les détails, puis nous expliquerons rapidement pourquoi nous avons besoin d'un seuil de masquage pour le calcul des paramètres des quantificateurs.

5.3.2 Présentation

Le modèle psychoacoustique utilisé est représenté par le schéma de la figure 5.2. Tout comme le système de tatouage, le modèle psychoacoustique fonctionne par trames, qui sont les mêmes que celles utilisées pour l'insertion. Par souci de clarté, sur ce schéma chaque variable est munie d'une lettre en exposant. L'exposant **n** indique que la variable est temporelle, l'exposant **k** que la variable est fréquentielle avec des échantillons espacés linéairement selon une échelle en Hz, et l'exposant **p** que la variable est fréquentielle avec des échantillons espacés linéairement selon une échelle en Barks (voir section 3.2). Plus précisément, les échantillons sont calculés tous les tiers de Bark, et par analogie avec le MPA présenté dans la norme AAC de MPEG, on parle de partitions. Comme il a déjà été dit dans l'introduction, le modèle psychoacoustique est simple et la majeure partie des calculs est effectuée pour prendre en compte le phénomène de masquage fréquentiel, le phénomène de

1. Les fenêtres KBD sont réglables par un coefficient qui dans le cas du Dolbi AC-3 est réglé pour que l'effet de Gibbs résultant soit négligeable devant l'effet de masquage fréquentiel.

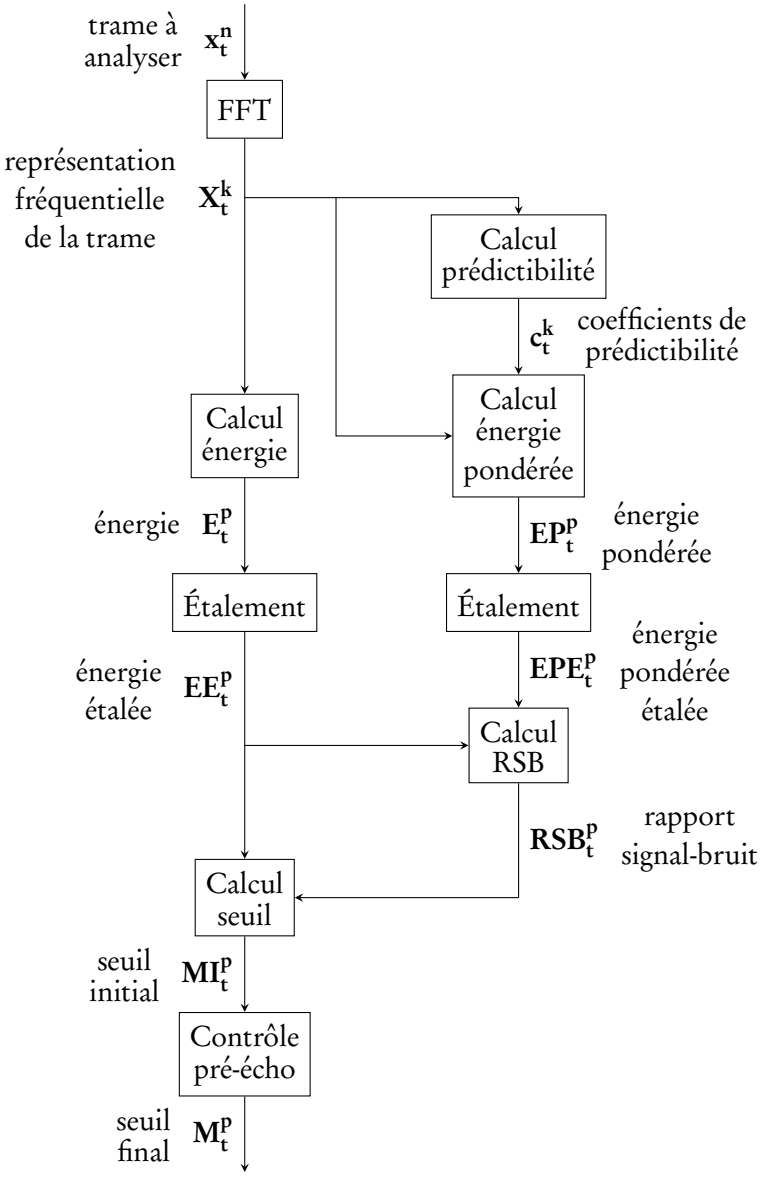


FIGURE 5.2 – Schéma de principe du modèle psychoacoustique

masquage temporel n'étant lui pris en compte que par un simple contrôle de pré-écho, ses mécanismes exacts étant beaucoup plus complexes et difficiles à modéliser.

Au niveau du fonctionnement plus détaillé de l'analyse psychoacoustique, la trame temporelle courante est tout d'abord pondérée par une fenêtre de Hann et transformée dans le plan temps-fréquence grâce à la FFT, et non la MDCT comme c'est le cas pour l'insertion. Ensuite, un calcul de prédictibilité est effectué pour chaque coefficient de la TFD. Cette mesure de prédictibilité est à relier avec ce qui a été présenté sur les modèles psychoacoustiques en section 3.2, et plus particulièrement sur le phénomène de masquage simultané. Nous avons en effet vu qu'il y a une atténuation différente pour le masquage suivant la nature du son masquant et du son masqué, et cette mesure de prédictibilité cherche à déterminer grossièrement la nature du masquant afin d'ajuster cette atténuation. La formule extraite de la norme MPEG-AAC pour le calcul du coefficient de prédictibilité pour chaque coefficient fréquentiel d'indice k est la suivante :

$$\mathbf{c_k}(k) = \frac{\left|\mathbf{X_t^k}(k) - \mathbf{X_{t,pred}^k}(k)\right|}{\left|\mathbf{X_t^k}(k)\right| + \left|\mathbf{X_{t,pred}^k}(k)\right|} \tag{5.11}$$

avec

$$\mathbf{X_{t,pred}^k}(k) = \left(2\left|\mathbf{X_{t-1}^k}(k)\right| - \left|\mathbf{X_{t-2}^k}(k)\right|\right)\exp\left(j\left(2\arg\left(\mathbf{X_{t-1}^k}(k)\right) - \arg\left(\mathbf{X_{t-2}^k}(k)\right)\right)\right) \tag{5.12}$$

Ensuite, deux calculs similaires ont lieu : un calcul d'énergie par partition et un calcul d'énergie pondérée par partition, dont les formules de calcul pour chaque partition sont les suivantes :

$$\mathbf{E_t^p}(p) = \sum_{k=0}^{N-1} \alpha(p,k)\left|\mathbf{X_t^k}(k)\right|^2 \tag{5.13}$$

$$\mathbf{EP_t^p}(p) = \sum_{k=0}^{N-1} \alpha(p,k)\mathbf{c_k}(k)\left|\mathbf{X_t^k}(k)\right|^2 \tag{5.14}$$

Dans ces formules, les coefficients $\alpha(p,k)$ sont des valeurs entre 0 et 1 qui pondèrent l'appartenance des coefficients d'indice k de la TFD à une partition p donnée. Pour une certaine partition p donnée, la plupart des coefficients $\alpha(p,k)$ vont être nuls, certains coefficients (adjacents) vont valoir 1 et un ou deux coefficients éventuellement (aux extrémités de la partition) vont avoir une valeur entre 0 et 1.

Suite à ces étapes, un calcul de RSB par partition est effectué à partir de l'énergie étalée $\mathbf{EE_t^p}$ et de l'énergie pondérée étalée $\mathbf{EPE_t^p}$. Ces deux énergies sont obtenues à partir des énergies calculées précédemment par convolution avec une fonction d'étalement **sf** (représentée figure 5.3), afin de commencer à prendre en compte le phénomène de masquage simultané :

$$\mathbf{EE_t^p} = \mathbf{E_t^p} * \mathbf{sf} \tag{5.15}$$

$$\mathbf{EPE_t^p} = \mathbf{EP_t^p} * \mathbf{sf} \tag{5.16}$$

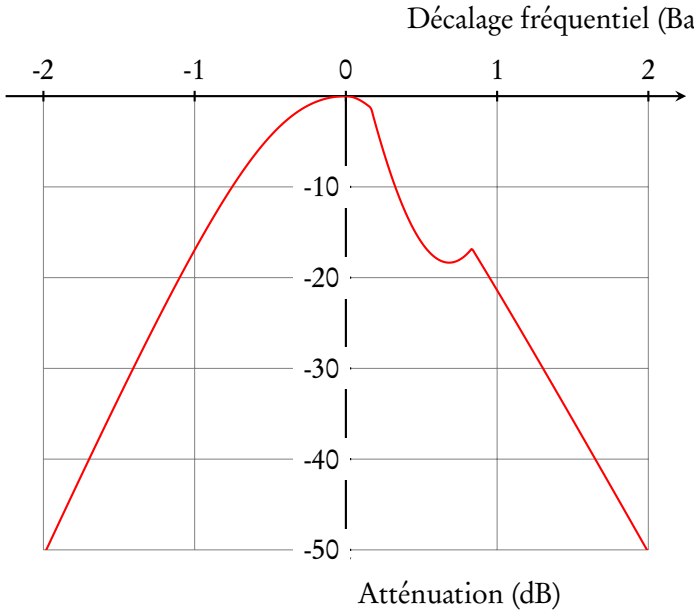


FIGURE 5.3 – Fonction d'étalement utilisée dans le MPA

Ce calcul de RSB est réalisé par la série de calculs suivante, extraite de la norme AAC de MPEG :

$$\mathbf{cb}_t^p(p) = \frac{\mathbf{EPE}_t^p(p)}{\mathbf{EE}_t^p(p)} \tag{5.17}$$

$$\mathbf{tb}_t^p(p) = -0.299 - 0.43\ln(\mathbf{cb}_t^p(p)) \tag{5.18}$$

$$\mathbf{tb}_t^p(p) = \max(0, \min(\mathbf{tb}_t^p(p), 1)) \tag{5.19}$$

$$\mathbf{RSB}_t^p(p) = \mathbf{tb}_t^p(p)\mathrm{TMN} + (1 - \mathbf{tb}_t^p(p))\mathrm{NMT} \tag{5.20}$$

avec TMN=18 et NMT=6 qui sont des constantes relatives à la nature des sons masquants et masqués (voir à nouveau la section 3.2), et permet d'avoir une atténuation variant de 6 à 18 dB suivant des sons masquant et masqué. Grâce à ce RSB, un seuil de masquage initial est calculé, en multipliant simplement l'énergie étalée par le RSB calculé :

$$\mathbf{MI}_t^p(p) = \mathbf{EE}_t^p(p) \cdot 10^{\mathbf{RSB}_t^p(p)/10} \tag{5.21}$$

Ce masque initial prend uniquement en compte le phénomène de masquage fréquentiel, la dernière étape consiste donc à prendre en compte le phénomène de masquage temporel. Si le modèle utilisé a pour but d'être assez précis dans le cas du masquage simultané, ce n'est pas réellement le cas pour le masquage non-simultané, en effet cette dernière étape consiste uniquement en un contrôle du pré-écho :

$$\mathbf{M}_t^p(p) = \min\left(\mathbf{MI}_t^p(p), k_{pe} \cdot \mathbf{MI}_{t-1}^p(p)\right) \tag{5.22}$$

où k_{pe} est une constante qui dépend de la longueur de trame. Par exemple dans la norme AAC de MPEG, un k_{pe} de 2 est utilisé pour les trames de longueur 2048 et un k_{pe} de 1 est utilisé pour les trames de longueur 256.

5.3.3 Utilisation du MPA dans le système de tatouage

Le seuil de masquage obtenu représente la puissance des modifications, dues au tatouage, qu'il est possible d'insérer dans la trame analysée sans que cela soit audible (théoriquement) pour l'utilisateur. La technique de tatouage utilisée étant la QIM, il va donc être assez simple de décrire la relation entre les caractéristiques des quantificateurs et la puissance maximale de l'erreur de tatouage qui peut être ajoutée à la trame.

En outre, ce seuil de masquage va nous permettre très facilement d'ajuster le débit de notre système de tatouage, de façon analogue à ce qui est fait dans les codeurs audio perceptuels. En effet, il va tout simplement suffire de translater le seuil de masquage (en décibels) pour modifier le débit de la trame courante. On pourra par exemple baisser le seuil de masquage de quelques décibels pour assurer une plus grande marge sur l'inaudibilité, ou tout simplement pour s'adapter au débit d'information à insérer, qui ne correspondra pas forcément exactement à la charge allouée naturellement par le modèle psychoacoustique.

5.4 Utilisation de la QIM

5.4.1 Introduction

Bien que la technique de QIM classique que nous utilisons pour cette implémentation simple ait été présentée théoriquement en section 2.4.2, nous allons décrire dans cette partie comment elle est utilisée dans le cadre de notre système de tatouage. Nous présenterons d'abord certains choix faits quant à la nature des quantificateurs utilisés pour l'insertion, puis nous présenterons en détail l'insertion et l'extraction par la technique de QIM.

5.4.2 Choix de types de quantificateurs

Si nous retournons rapidement dans le cadre des codeurs perceptuels, les types de quantificateurs utilisés sont généralement logarithmiques, ou alors ce sont des quantificateurs uniformes utilisés sur le logarithme des valeurs à quantifier (par exemple dans la norme AAC de MPEG [ISO09]), ce qui est équivalent. Cette utilisation de quantificateurs logarithmiques (ou procédé similaire) se justifie par le fait que les coefficients à quantifier ont une distribution gaussienne généralisée. Cependant dans notre cas, la situation est légèrement différente étant donné que la quantification a ici pour but d'insérer de l'information et non de compresser le signal. En effet, on cherche à maximiser la quantité d'information transmise pour une distorsion autorisée (donnée par le MPA), indépendamment du fait qu'un coefficient soit de valeur forte ou faible. Nous allons donc utiliser des quantificateurs uniformes. Nous pouvons aussi remarquer que si certains codeurs audio utilisent des quantificateurs vectoriels,

ce n'est généralement pas pour des raisons de psychoacoustique, mais plutôt pour des gains en taux de compression ou en temps de calcul. Les quantificateurs que nous utiliserons seront donc scalaires et non vectoriels.

5.4.3 Insertion et décodage d'un message

Pour insérer un symbole donné d'un alphabet $\mathscr{A}$ comportant $\#\mathscr{A}$ éléments dans un coefficient avec la QIM, il faut $\#\mathscr{A}$ quantificateurs entrelacés. Afin de simplifier les calculs, nous allons imposer que le message soit de type binaire, c'est-à-dire que le nombre de quantificateurs sera toujours une puissance de 2. Si l'on note $C_t(k)$ la charge insérée exprimée en nombre de bits dans le coefficient MDCT d'indice k de la trame t noté $X_t(k)$, et $\mathscr{S}_t(k)$ l'ensemble des quantificateurs utilisés pour réaliser l'insertion dans ce coefficient, on a alors :

$$\#\mathscr{A} = \#\mathscr{S}_t(k) = 2^{C_t(k)} \tag{5.23}$$

Les $\#\mathscr{A}$ quantificateurs sont entrelacés régulièrement afin de minimiser les erreurs de décodage.

Afin de pouvoir définir totalement les ensembles de quantificateurs $\mathscr{S}_t(k)$, il reste donc à définir le pas de quantification des quantificateurs. Le choix qui est fait ici est de fixer l'erreur de décodage, indépendamment du coefficient MDCT $X_t(k)$. Développons maintenant les implications de ce choix. Comme présenté dans la partie théorique sur la QIM, on se positionne sur un coefficient MDCT $X_t(k)$ et on se munit d'un quantificateur de référence, ou quantificateur prototype Q, de pas de quantification Δ_C. On omet ici la notation ${}_t(k)$ pour simplifier l'écriture tout en n'oubliant pas que nous nous situons à un coefficient MDCT donné. L'ensemble de quantificateurs $\mathscr{S}$ contient donc 2^C quantificateurs que l'on indexe par un entier entre 0 et 2^C-1 :

$$\mathscr{S} = \{Q_i\}_{0 \leq i < 2^C} \tag{5.24}$$

Chaque quantificateur est alors défini à partir du quantificateur prototype Q :

$$\forall i \in \left[0, 2^C - 1\right],\ Q_i(x) = Q\left(x - \frac{i}{2^C}\Delta_C\right) + \frac{i}{2^C}\Delta_C \tag{5.25}$$

La plus petite distance entre les régions d'insertion associées à deux valeurs de tatouage différentes est donc de $\Delta_C/2^C$. Afin de faciliter le choix des paramètres des quantificateurs, comme nous le verrons en détail dans la section 5.5, nous avons choisi de fixer la distance entre les régions d'insertion comme une constante Δ_Q ne dépendant pas de la charge C. On a alors :

$$\frac{\Delta_C}{2^C} = \Delta_Q \Leftrightarrow \Delta_C = 2^C \Delta_Q \tag{5.26}$$

L'insertion d'un mot c par QIM sur le coefficient MDCT se fait donc simplement en choisissant le quantificateur indexé par c, Q_c, et en remplaçant le coefficient par sa valeur

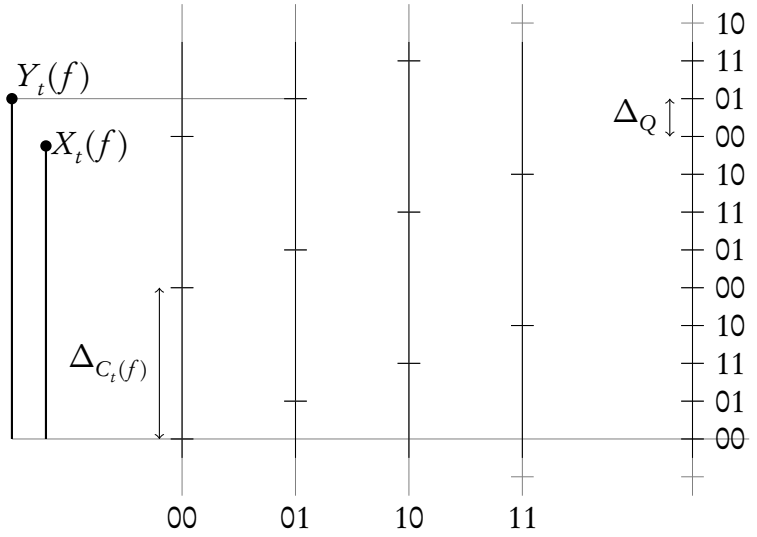


FIGURE 5.4 – Schéma explicatif pour l'insertion sur un coefficient MDCT par QIM, avec une charge de 2 bits. Les quantificateurs sont indexés par un code sur 2 bits et une grille globale est représentée à droite. À gauche se trouvent le coefficient MDCT original $X_t(k)$ et celui tatoué $Y_t(k)$. Dans cet exemple le message 01 est inséré.

quantifiée par Q_c :

$$\mathbf{Y}_t(k) = Q_c(\mathbf{X}_t(k)) \tag{5.27}$$

comme représenté sur la figure 5.4. Le décodage se fait simplement en cherchant le quantificateur Q_c qui possède le représentant le plus proche du coefficient MDCT dont on veut extraire le tatouage. La valeur décodée est alors l'indice du quantificateur, c :

$$\widehat{c} = \arg\min_c |Q_c(\mathbf{Y}_t(k)) - \mathbf{Y}_t(k)| \tag{5.28}$$

5.5 Calcul et transmission des paramètres

5.5.1 Introduction

Dans cette section nous allons expliquer quels sont les paramètres nécessaires pour effectuer l'insertion de données, comment ces paramètres sont calculés lors de l'insertion et comment ils sont transmis afin d'être réutilisés lors de l'extraction. Tout d'abord, nous allons présenter en détail le calcul de ces paramètres, basé sur les deux contraintes d'inaudibilité et de robustesse. Après cela nous expliquerons pourquoi nous avons eu besoin de travailler par sous-bandes et finalement nous détaillerons comment le débit peut être adapté grâce au seuil de masquage comme mentionné en section 5.3.

Comme nous l'avons vu dans la section 5.4 précédente, nous avons choisi de fixer la distance entre les régions d'insertion à une constante Δ_Q, de telle que sorte que les pas de quantification des quantificateurs soient égaux à $2^{C_t(f)}\Delta_Q$. Dans les prochaines sections nous allons détailler comment Δ_Q et les charges $C_t(f)$ sont calculées.

5.5.2 Robustesse

La première contrainte que nous allons développer est la robustesse. Nous avons déjà expliqué que notre système de tatouage n'est pas à vocation sécuritaire, cependant il faut prendre en compte la quantification PCM 16 bits qui a lieu entre l'insertion et l'extraction, afin de reconvertir le signal au format PCM 16 bits.

Nous nous plaçons ici dans le cas d'une trame t quelconque. La trame de signal hôte considérée $\mathbf{x}_t$ étant au format PCM 16 bits, nous considérons que les échantillons représentent des valeurs entières signées sur 16 bits :

$$\mathbf{x}_t \in [\![-2^{15}, 2^{15}-1]\!]^N \tag{5.29}$$

Cependant les échantillons temporels de la trame $\mathbf{y}_t$ de signal tatoué ne sont pas forcément entiers. La quantification PCM 16 bits va donc introduire un bruit lorsque l'on va reconvertir le signal au format PCM 16 bits.

On note $\mathbf{b}$ le bruit dû à la quantification PCM 16 bits et $\widehat{\mathbf{y}}_t$ la trame de signal tatouée convertie au format PCM 16 bits :

$$\widehat{\mathbf{y}}_t = \mathbf{y}_t + \mathbf{b} \tag{5.30}$$

Nous allons calculer ici l'influence de ce bruit. Nous commençons tout d'abord par effectuer l'hypothèse, assez réaliste, que les échantillons du bruit de quantification $b(n)$ suivent une loi uniforme centrée de support de largeur 1, la quantification étant faite au plus proche voisin entier.

$$b(n) \sim \mathcal{U}\left(-\frac{1}{2}, \frac{1}{2}\right) \tag{5.31}$$

Les coefficients MDCT $\widehat{\mathbf{Y}}_t$ de $\widehat{\mathbf{y}}_t$ étant obtenus par une transformation linéaire, si l'on note $\mathbf{B}$ les coefficients MDCT de $\mathbf{b}$, on a :

$$\widehat{\mathbf{Y}}_t = \mathbf{Y}_t + \mathbf{B} \tag{5.32}$$

En utilisant une version du théorème central limite avec hypothèse faible (conditions de Lyapunov ou Lindeberg [Fel71]), on peut alors montrer (voir annexe B) que les coefficients MDCT $B_t(k)$ suivent une loi normale de variance σ^2_{PCM} :

$$B_t(k) \sim \mathcal{N}\left(0, \sigma^2_{\text{PCM}}\right) \tag{5.33}$$

où σ^2_{PCM} représente la variance du bruit de quantification dans le domaine temporel. Ce bruit étant uniforme de support unitaire, sa variance vaut :

$$\sigma^2_{\text{PCM}} = \frac{1}{12} \tag{5.34}$$

Le bruit uniforme sur les coefficients temporels de variance σ^2_{PCM} se traduit donc par un bruit gaussien de même variance sur les coefficients MDCT, indépendamment de l'indice du coefficient considéré.

Étant donné que les zones d'insertion sont à une distance Δ_Q les unes des autres, indépendamment de la charge insérée, nous allons pouvoir fixer ce Δ_Q en fonction d'un taux d'erreur cible théorique souhaité. En effet nous pouvons en première approximation estimer que le taux d'erreur théorique p_{es} de décodage pour l'information tatouée sur un coefficient est de (voir annexe B.2) :

$$p_{es} = 1 - \text{erf}\left(\frac{\Delta_Q}{2\sqrt{2}\sigma_{\text{PCM}}}\right) \tag{5.35}$$

où erf désigne la fonction d'erreur usuelle :

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp\left(-t^2\right) \mathrm{d}t \tag{5.36}$$

On a donc :

$$\Delta_Q = 2\sqrt{2}\sigma_{\text{PCM}}\, \text{erf}^{-1}(1 - p_{es}) \tag{5.37}$$

Ce taux d'erreur p_{es} est à comprendre comme un taux d'erreur symbole, chaque symbole étant l'information insérée dans un coefficient MDCT. Une erreur résultant du décalage sur un niveau de quantification d'un quantificateur adjacent n'impliquera pas forcément une erreur de un bit. On peut cependant utiliser un code de Gray afin de minimiser le taux d'erreur binaire. Il est à noter que la charge étant variable suivant le coefficient MDCT considéré, la taille des symboles est elle aussi variable.

Maintenant que le paramètre Δ_Q est défini par le choix d'un taux d'erreur cible p_{es}, il reste à calculer les charges $C_t(f)$.

5.5.3 Inaudibilité

Nous allons montrer ici comment la contrainte d'inaudibilité va fixer les charges $C_t(f)$. En entrée du bloc de calcul des charges se trouve le seuil de masquage $\mathbf{M}_t$, calculé par le modèle psychoacoustique. Ce seuil de masquage représente la puissance maximale de bruit que l'on peut ajouter à la trame hôte sans que cela soit audible. L'erreur de tatouage maximale étant de $\Delta_{C_t(f)}/2$ (avec $\Delta_{C_t(f)} = 2^{C_t(f)}\Delta_Q$) et la contrainte d'inaudibilité imposant que :

$$\left(\frac{\Delta_{C_t(f)}}{2}\right)^2 \leq M_t(f) \tag{5.38}$$

il vient :

$$C_t(f) \leq \frac{1}{2}\log_2\left(\frac{M_t(f)}{\Delta_Q^2}\right)+1 \tag{5.39}$$

Les charges étant des nombres entiers positifs, et sachant que l'on cherche à maximiser le débit, on choisit donc :

$$C_t(f) = \max\left(0, \left\lfloor \frac{1}{2}\log_2\left(\frac{M_t(f)}{\Delta_Q^2}\right)+1 \right\rfloor\right) \tag{5.40}$$

où $\lfloor . \rfloor$ dénote l'opération de troncature (arrondi vers la valeur inférieure).

Étant donné que la distance Δ_Q est fixée au départ, afin de pouvoir effectuer une extraction correcte il suffit de transmettre les charges $C_t(f)$. De manière expérimentale, on constate que dans la quasi-totalité des cas les charges $C_t(f)$ ont des valeurs comprises entre 0 et 15, ce qui va permettre de les coder sur 4 bits. Si une valeur de 15 paraît très élevée, de telles valeurs sont tout de même rarement atteintes et uniquement dans le cas où la trame de signal est très puissante (rock, pop…).

Nous allons donc voir maintenant comment exactement les valeurs de ces charges sont transmises grâce à un travail en sous-bandes.

5.5.4 Utilisation des sous-bandes

Pour chaque trame de signal, coder les $N/2$ valeurs des charges calculées n'est pas possible. En effet comme on l'a expliqué précédemment en section 5.2, il y a autant de coefficients MDCT que d'échantillons temporels, et donc autant de valeurs de charges. Il faudrait donc transmettre 44100 valeurs de charges par seconde et par canal, codées sur 4 bits. Ceci représente un débit de 176.4 kb/s/c, ce qui est bien trop élevé. Nous allons donc une fois de plus emprunter une idée au codage perceptuel, le travail par sous-bande. L'idée est de définir des sous-bandes de coefficients MDCT au sein desquelles nous allons utiliser la même charge. Afin de faciliter l'utilisation du système de tatouage, nous avons de plus choisi de fixer la taille de ces sous-bandes à $N_b = 32$ coefficients. La charge $C'_t(b)$ choisie pour une sous-bande b est simplement prise comme étant la charge minimale des coefficients composant la sous-bande :

$$\forall b \in [\![1, N/N_b]\!],\ C'_t(b) = \min_{f \in [\![b(N_b-1), bN_b-1]\!]} C_t(f) \tag{5.41}$$

Dans chaque sous-bande b, on peut alors insérer $N_b \cdot C'_t(b)$ bits, que l'on peut voir comme $C'_t(b)$ mots de 32 bits. Dans ces conditions, il est suffisant de ne transmettre que $44100/N_b$ valeurs de charges par seconde par canal, codées sur 4 bits. Comparé aux 176.4 kb/s/c on redescend à environ 5.5 kb/s/c. Ce débit peut paraître élevé pour des systèmes de tatouage traditionnels, cependant dans notre cas où l'on vise des débits de l'ordre de la centaine de kb/s/c, cette perte de débit utile semble tout à fait acceptable.

Notons que plusieurs tests ont été faits sur le type de sous-bande à utiliser, par exemple des sous-bandes alignées sur les bandes critiques ou des fractions de bandes critiques. Cependant les meilleurs résultats en terme de compromis entre débit et inaudibilité ont été obtenus pour des sous-bandes de tailles égales. En outre, cela facilite le découpage du message pour l'insertion (et donc sa reconstruction à l'extraction), puisque chaque sous-bande contient un nombre entier de mots de 32 bits. Maintenant que nous savons quels paramètres doivent être transmis, nous allons voir comment les valeurs des capacités par sous-bande $C'_t(b)$ sont insérées.

5.5.5 Insertion des paramètres

Nous allons voir ici comment vont être insérés les 5.5 kb/s/c d'information nécessaire au bon décodage du message à transmettre, en rappelant que l'information doit pouvoir être décodée trame par trame. L'ensemble des coefficients C'_t relatifs à une trame donnée doit donc être inséré dans cette même trame. Le choix que nous avons fait dans ce premier système basique est alors d'insérer les valeurs des charges C'_t dans les sous-bandes localisées en hautes fréquences en utilisant dans cette zone les valeurs de charges fixes connues au décodeur. Ces sous-bandes ne serviront alors qu'à véhiculer cette information. Le choix des hautes-fréquences est le fruit de la réflexion suivante. Les valeurs des charges C'_t doivent être transmises à toutes les trames, indépendamment du contenu musical. Or, le seuil de masquage, lui, évolue suivant le contenu musical. Cependant il y a une caractéristique psychoacoustique que nous n'avons pas intégrée dans notre modèle psychoacoustique et qui est indépendante du contenu musical, c'est le seuil d'audition absolu, dont on a déjà parlé en section 3.2. Ce seuil est très élevé dans les très basses fréquences, sur une bande de fréquence très faible, et surtout en hautes fréquences, à partir d'environ 16 kHz, ce qui nous a amené à choisir les hautes-fréquences comme région d'insertion pour tatouer les valeurs des charges C'. Afin de s'assurer que nous restons bien en dessous du seuil d'audition absolu, nous choisissons d'avoir une charge maximum de 3 bits par coefficient dans cette région haute-fréquence. La figure 5.5 montre deux exemples de spectre et le seuil d'audition absolu pour illustrer ces propos. Étant donné que nous avons fixé la taille des sous-bandes à $N_b = 32$, le nombre de sous-bandes nécessaires va varier suivant la taille N des trames, comme on peut le voir sur le tableau 5.1. Prenons l'exemple des trames de longueur 2048 : elles contiennent 32 sous-bandes (1024 coefficients MDCT divisés par la taille des sous-bandes qui est de 32), ce qui fait 128 bits à insérer dans les hautes-fréquences (32×4 bits par valeur de charge). 2 sous-bandes en hautes fréquences sont utilisées pour encoder les paramètres de charges, les deux avec une insertion sur 2 bits. Finalement la bande des hautes fréquences s'étend de 20,67 à 22,05 kHz environ.

5.5.6 Adaptation du seuil

Comme il a été dit dans la présentation générale de notre système, le débit d'insertion de notre système de tatouage doit pouvoir être adapté au débit du message que l'on souhaite

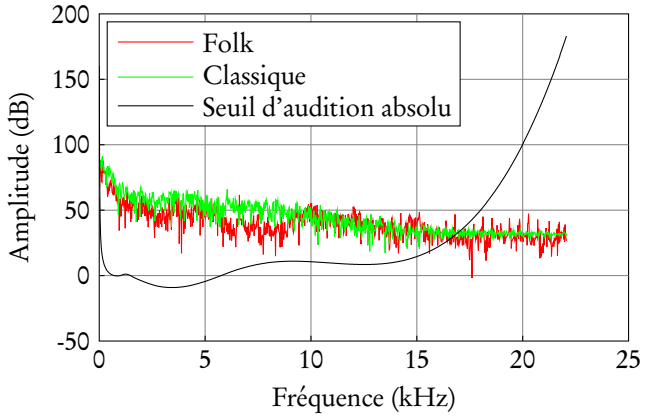


FIGURE 5.5 – Exemples de spectres de signaux de musique classique et folk, et comparaisons avec le seuil d'audition absolu

N	# sb total	# bits à insérer en HF	# sb HF utilisées	charges dans les sb HF	limites zone HF (kHz)
4096	64	256	3	2 3 3	21,02 - 22,05
2048	32	128	2	2 2	20,67 - 22,05
1024	16	64	1	2	20,67 - 22,05
512	8	32	1	1	19,29 - 22,05

TABLE 5.1 – Caractéristiques des sous-bandes pour les différentes longueurs de trame

insérer. En particulier, on souhaite que le débit d'insertion puisse être adapté trame par trame dans le cas particulier où l'information insérée dans une trame est relative à la trame elle-même. En effet si l'on peut régler le débit d'insertion trame par trame, on pourra par extension régler le débit d'insertion total, ce qui couvre la totalité des cas que l'on souhaite traiter pour le projet DReaM.

Afin de régler ce débit, on propose donc de jouer sur le seuil de masquage $\mathbf{M}_t$, et plus exactement de le translater d'un certain nombre de décibels α afin que les charges calculées conduisent au débit recherché. On note $\mathbf{M}_{t,\alpha}$ le seuil translaté de α dB :

$$\mathbf{M}_{t,\alpha} = 10^{\alpha/10}\mathbf{M}_t \tag{5.42}$$

Si l'on pose :

$$K_t(f) = \frac{1}{2}\log_2\left(\frac{\mathbf{M}_t(f)}{\Delta_Q^2}\right) + 1 \tag{5.43}$$

l'équation devient alors :

$$\begin{aligned} C_{t,\alpha}(f) &= \max\left(0, \left\lfloor \frac{1}{2}\log_2\left(\frac{\mathbf{M}_{t,\alpha}(f)}{\Delta_Q^2}\right) + 1 \right\rfloor\right) \\ &= \max\left(0, \left\lfloor K_t(f) + \frac{\alpha}{20}\log_2(10) \right\rfloor\right) \qquad (5.44) \\ &= \max\left(0, \left\lfloor K_{t,\alpha}(f) \right\rfloor\right) \qquad (5.45) \end{aligned}$$

Les valeurs de charges par sous-bandes sont alors toujours calculées de la même façon :

$$\forall b \in [\![1, N/N_b]\!],\ C'_{t,\alpha}(b) = \min_{f \in [\![(b-1)N_b, bN_b-1]\!]} C_{t,\alpha}(f) \tag{5.46}$$

Si l'on note :

$$K'_t(b) = \min_{f \in [\![(b-1)N_b, bN_b-1]\!]} K_t(f) \tag{5.47}$$

on a alors :

$$C'_{t,\alpha}(b) = \max\left(0, \left\lfloor K'_t(b) + \frac{\alpha}{20}\log_2(10) \right\rfloor\right) \tag{5.48}$$

La charge totale pour la trame t courante est finalement donnée par :

$$R_{t,\alpha} = \sum_{b=1}^{N/Nb} C'_{t,\alpha}(b) \tag{5.49}$$

Le problème est alors que l'on ne peut pas inverser ces expressions pour obtenir la valeur de α qui conduit au débit recherché, à cause des opérations de troncature et de maximum

qui ne sont pas inversibles. Cependant on peut remarquer que les $K_{t,\alpha}(f)$ sont linéaires en α, avec un coefficient de proportionnalité de $\log_2(10)/20$ indépendant de f. L'évolution de $C'_{t,\alpha}(f)$ en fonction de α suivra donc une courbe en escalier (avec phénomènes de saturation à 0 et 15 dûs aux opérations de minimum et de maximum). Par conséquent, si l'on choisit un seuil de translation :

$$\alpha_k = \frac{20}{\log_2(10)} k \tag{5.50}$$

avec k entier, on a :

$$\forall b,\ C'_{t,\alpha_k}(b) = \max\left(0, C'_t(b) + k\right) \tag{5.51}$$

c'est-à-dire que toutes les charges sont modifiées de k (sauf en cas de saturation à 0 ou 15). On peut alors comprendre qu'en faisant varier α entre α_k et α_{k+1}, les charges vont être progressivement augmentées de 1 bit, une par une, jusqu'à ce qu'elles aient toutes été augmentées de 1 bit. On peut de plus voir que l'ordre dans lequel les charges vont être incrémentées va dépendre de la valeur de $K_t(b)$. Plus la partie décimale de $K_t(b)$ va être grande, plus vite $C_{t,\alpha}(b)$ va être incrémentée (autrement dit, si la partie décimale de $K_t(b)$ est grande, il n'y aura besoin que d'une faible valeur de α pour incrémenter la charge $C'_{t,\alpha}(b)$, et si la partie décimale est faible, il faudra une plus grande valeur de α).

L'algorithme utilisé va donc être très simple, et on peut le décrire comme ceci :

1. calcul des $K'_t(b)$ pour une trame t
2. calcul des $C'_t(b)$ correspondants
3. tri des $K'_t(b)$ en fonction de b, suivant un ordre décroissant (resp. croissant) de la partie décimale si l'on souhaite augmenter (resp. diminuer) la charge, on obtient alors une permutation $\sigma_t \in \mathfrak{S}_{N/N_b}$
4. on incrémente (resp. décrémente) successivement $C'_t(\sigma_t(1))$ puis $C'_t(\sigma_t(2))\ldots$ jusqu'à atteindre le débit souhaité (en continuant à s'assurer que les charges soient comprises entre 0 et 15)

Cet algorithme à la fois très simple et très rapide permet donc de s'adapter à la charge recherchée, de la façon la plus correcte possible du point de vue psychoacoustique, c'est-à-dire en modifiant le bruit d'insertion de façon régulière sur toute la bande de fréquence.

5.6 Bilan

Nous avons montré dans ce chapitre comment développer un système de tatouage adapté aux besoins du projet DReaM en s'inspirant des bases du codage audio perceptuel. Plus précisément nous avons montré comment utiliser un modèle psychoacoustique pour guider une insertion par quantification des coefficients temps-fréquence de la transformée la plus adaptée à nos contraintes, la MDCT. Finalement nous avons détaillé une méthode utilisant

une double insertion et des sous-bandes afin de maximiser le débit d'insertion, et nous avons expliqué comment adapter l'insertion à une charge donnée de la façon la plus cohérente possible avec le modèle psychoacoustique.

Chapitre 6

Premières expériences

Sommaire

6.1 Introduction

Dans ce chapitre nous allons présenter et détailler les expériences qui ont été réalisées afin de valider notre système de tatouage pour une utilisation dans le cadre du projet DReaM ou d'autres applications similaires. Plus particulièrement, nous allons montrer que ce système de tatouage simple permet une transmission à haut débit, avec un faible taux d'erreur et avec un décodage possible en temps-réel (cet aspect étant très sensible au langage et au matériel utilisé, il ne sera abordé que superficiellement).

Ce chapitre est organisé de la manière suivante. Tout d'abord, nous allons présenter en section 6.2 les différentes bases de données audio que nous avons utilisées pour faire nos expériences. Dans la section suivante 6.3, nous discutons de la pertinence de l'approximation de la probabilité d'erreur fournie par la formule (5.35) en comparant l'expression théorique aux probabilités d'erreur obtenues dans la pratique. Ensuite, dans la section 6.4, nous présentons des courbes de qualité audio en fonction du débit d'insertion, nous détaillons l'impact de la longueur des trames N sur ces courbes et nous discutons des débits d'insertion atteignables suivant les types de signaux musicaux considérés. Étant donné que de très nombreuses mesures de la qualité audio en fonction du débit d'insertion et d'autres paramètres sont nécessaires pour évaluer les performances de notre système de tatouage, il paraissait peu raisonnable de réaliser des tests subjectifs pour chacun des réglages possibles. Pour pallier ce problème, nous avons eu recours à des algorithmes d'évaluation objective de la qualité de signaux audio, notamment l'algorithme PEAQ décrit en section 3.5. Cet algorithme ayant été développé dans le but spécifique d'évaluer les performances en terme de qualité des codeurs audio, avant de l'utiliser nous avons dû nous assurer que la métrique renvoyée par cet algorithme était appropriée à notre système de tatouage. En section 6.5 nous présentons donc les expériences qui nous ont conduit à considérer l'algorithme PEAQ comme fournissant des résultats satisfaisants pour évaluer la qualité de notre système de tatouage, et cet algorithme sera donc utilisé dans les expériences suivantes. Dans cette même section nous présentons aussi une propriété intéressante du modèle psychoacoustique utilisé.

6.2 Bases de données utilisées

La principale base de données que nous utilisons, BD1, est formée de 96 signaux stéréophoniques de 30 secondes, au format PCM 16 bits (44.1 kHz, 16 bits), pour un total de 48 minutes. Cette base de données est extraite de morceaux de musique commerciaux, de styles variés (principalement pop, rock, classique, jazz, folk, reggae, latino et rap). Cette base de données est utilisée pour toutes les expériences basées sur des mesures objectives, comme les mesures objectives de qualité obtenues grâce à l'algorithme PEAQ, ou les mesures de taux d'erreur.

Cependant pour les expériences subjectives, en l'occurrence les tests d'écoute réalisés pour valider l'utilisation de l'algorithme PEAQ, utiliser une base de données de 48 minutes est difficilement possible. Afin de mener à bien ces expériences, nous avons donc créé une seconde base de données, BD2, formée de 8 signaux de 10 secondes extraits de la base de

données BD1. Ces signaux ont été choisis afin d'être à la fois représentatifs de différents styles musicaux, et les plus problématiques possible pour le système de tatouage (fortes attaques de notes, échos).

6.3 Taux d'erreur

Nous avons ici cherché à vérifier si l'approximation du taux d'erreur donnée par l'équation (5.35) était proche du taux d'erreur pratique. Rappelons que ce taux d'erreur n'est pas binaire, mais une sorte de taux d'erreur symbole, un symbole étant l'information insérée dans un coefficient MDCT donné. La charge insérée à chaque coefficient MDCT étant variable, chaque symbole est de taille variable et donc on ne peut pas facilement relier le taux d'erreur binaire du système avec ce taux d'erreur symbole.

Cette expérience visant uniquement à étudier la pertinence de l'équation (5.35), les charges utilisées pour décoder les données sont celles qui ont été calculées au codeur, $\mathbf{Ca}$, et ne sont pas celles extraites normalement au décodeur, $\widehat{\mathbf{Ca}}$. La seule perturbation prise en compte est donc celle venant de la quantification PCM 16 bits des échantillons temporels lors de la conversion au format PCM 16 bits, ce qui est bien ce que l'on cherche à modéliser quand on utilise l'équation (5.35). Cependant, lorsque l'on procède à un décodage normal, ce sont les valeurs des charges $\widehat{\mathbf{Ca}}$ décodées dans les hautes fréquences, qui sont utilisées. S'il n'y a pas d'erreur dans le décodage de ces charges, il n'y a pas de problème particulier. Cependant s'il y a des erreurs dans le décodage, ce peut être extrêmement gênant pour la récupération du message inséré. En effet, lorsqu'une erreur de décodage est commise dans les basses fréquences, le message décodé est erroné dans le sens où certains mots de 32 bits sont corrompus. Ceci est gênant mais pas forcément rédhibitoire, et on peut pallier ce problème en ajoutant par exemple des codes correcteurs d'erreurs si besoin. Mais, si l'erreur est commise en hautes fréquences, alors le décodage en basses fréquences qui va s'ensuivre ne va pas résulter en le bon nombre de mots de 32 bits, c'est-à-dire que l'on va décoder un nombre de mots de 32 bits différent de celui originellement inséré. Ceci pose un gros problème, puisque l'on ne peut pas savoir où sont apparus (ou disparu) les mots en trop ou en défaut. On appelle ce genre de problème des problèmes de synchronisation interne, et l'on discutera dans l'implémentation améliorée (partie III) de méthodes pour traiter ce genre de situations. Dans cette partie nous nous concentrons sur les erreurs de décodage dues à la quantification PCM 16 bits et nous supposons les valeurs de charges correctement transmises.

La base de données utilisée pour cette expérience est la base BD1, pour différentes tailles de trames, ce qui représente environ $2.4 \cdot 10^8$ coefficients MDCT (certains coefficients n'étant pas tatoués, leur charge étant nulle, ils ne sont donc pas comptés ici). Les résultats pratiques et théoriques sont présentés dans la table 6.1. Comme nous pouvons le voir, dans tous les cas la valeur théorique cible est soit dans l'intervalle de confiance à 95% calculé pour les valeurs pratiques, soit très proche. L'intervalle de confiance choisi est l'intervalle de Agresti et Coull pour les lois binomiales [AC98], pour son bon comportement dans le cas d'un

TES cible	Intervalle de confiance à 95% 512	1024	2048	4096
10^{-2}	$[0.992, 0.994]\,10^{-2}$	$[0.996, 0.998]\,10^{-2}$	$[0.997, 0.999]\,10^{-2}$	$[0.999, 1.001]\,10^{-2}$
10^{-3}	$[0.973, 0.981]\,10^{-3}$	$[0.983, 0.991]\,10^{-3}$	$[0.992, 1.000]\,10^{-3}$	$[0.993, 1.001]\,10^{-3}$
10^{-4}	$[0.945, 0.970]\,10^{-4}$	$[0.961, 0.986]\,10^{-4}$	$[0.974, 0.998]\,10^{-4}$	$[0.984, 1.009]\,10^{-4}$
10^{-5}	$[0.879, 0.955]\,10^{-5}$	$[0.882, 0.958]\,10^{-5}$	$[0.927, 1.005]\,10^{-5}$	$[0.910, 0.987]\,10^{-5}$
10^{-6}	$[0.854, 1.101]\,10^{-6}$	$[0.798, 1.039]\,10^{-6}$	$[0.834, 1.080]\,10^{-6}$	$[0.831, 1.076]\,10^{-6}$

TABLE 6.1 – Taux d'erreur symbole cible et intervalle de confiance à 95% pour différentes tailles de trames

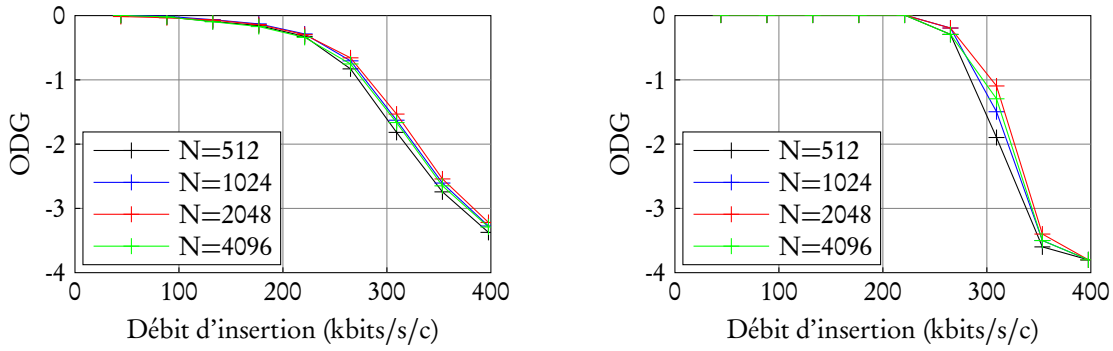
nombre d'épreuves très élevé [BCD01]. On remarque que même si l'intervalle de confiance ne contient pas toujours la valeur cible, elle est néanmoins très proche, ce qui tend à valider l'utilisation de la formule approchée utilisée pour choisir Δ_Q en fonction du taux d'erreur que l'on souhaite avoir.

Dans la suite, à moins qu'il ne soit précisé autrement, le taux d'erreur cible utilisé pour les expérimentations sera de 10^{-6}.

6.4 Courbes débit-qualité

Nous avons vu dans la présentation que notre système de tatouage possède la particularité d'avoir un débit d'insertion adaptatif. Nous allons donc étudier le comportement de la qualité audio en fonction du débit d'insertion. En outre, nous avons aussi vu que la longueur des trames utilisées pour le système de tatouage peut être réglée, c'est pourquoi nous présentons ces résultats pour différentes tailles de trames. La qualité audio est mesurée grâce à l'algorithme PEAQ (présenté en section 3.5) et les longueurs de trame étudiées sont 512, 1024, 2048 et 4096, valeurs utilisées couramment pour la MDCT, et aussi plus généralement pour des signaux audio échantillonnés à 44.1 kHz. En effet les différentes durées de ces trames à cette fréquence d'échantillonnage (respectivement 11.6, 23.2, 46.4 et 92.9 ms) sont souvent considérées comme permettant un bon compromis entre résolution fréquentielle et aspect stationnaire, tous deux nécessaires à une bonne analyse fréquentielle. Afin de pouvoir combiner les résultats des différents éléments de la base de données BD1, nous avons calculé l'ODG (c.f. table 3.2) des signaux tatoués pour 9 valeurs de débit d'insertion, les multiples de 44.1 kb/s/canal (de 44.1 à 396.9). Les données insérées sont aléatoires, et les résultats sont présentés sur les figures 6.1 et 6.2.

La première figure 6.1 présente la moyenne et la médiane des ODG calculées pour la base de données BD1. Tout d'abord, on remarque que les courbes suivent une même tendance indépendamment de la longueur de trame considérée. La moyenne décroît en pente douce jusqu'aux alentours de 220 kb/s/c, ce débit correspondant à une ODG d'environ -0,3, puis la pente devient rapidement plus raide pour rester à peu près constante à partir de 270 kb/s/c.



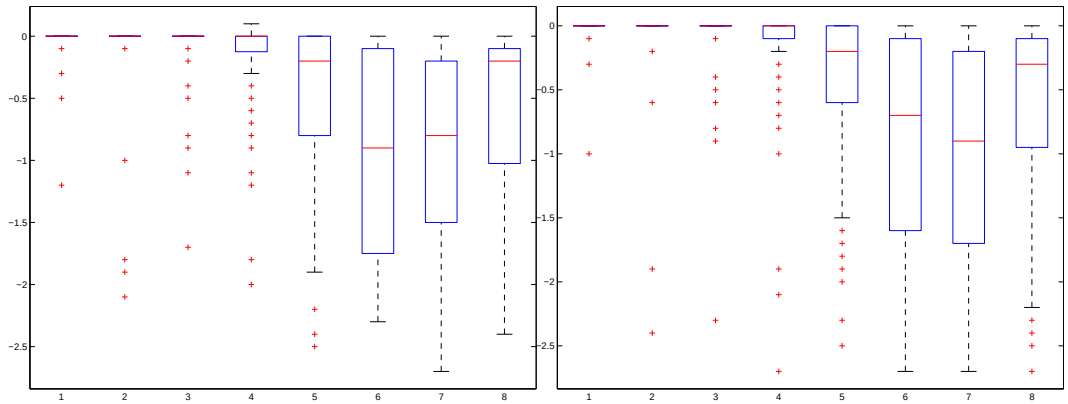
(a) Moyenne

(b) Médiane

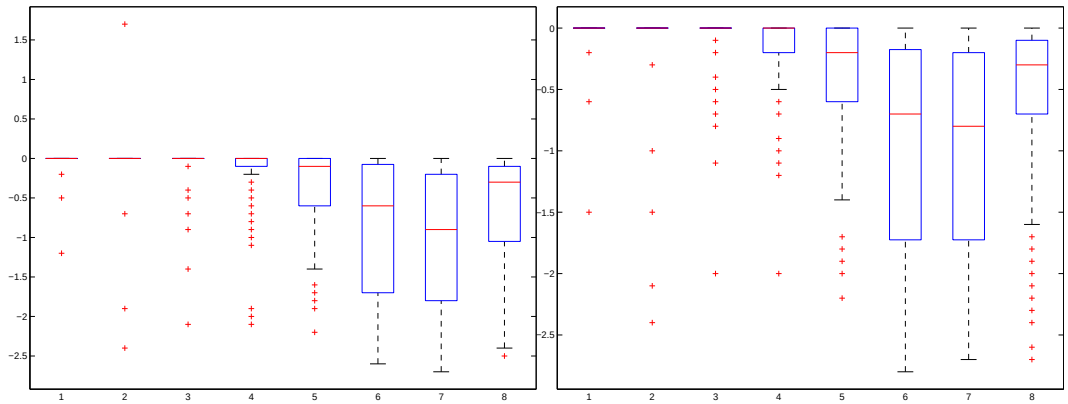
FIGURE 6.1 – Moyenne et médiane des ODG sur la base de données BD1, pour différentes longueurs de trames

La médiane quant à elle est constante avec une ODG de 0 jusqu'à environ 220 kb/s/c, puis décroît assez brutalement et un peu moins régulièrement que la moyenne. Cette allure globale, qui est similaire pour la médiane et la moyenne, est assez encourageante puisqu'elle tend à montrer que la qualité est bien reliée au débit d'insertion pour notre système : plus on insère d'information, plus on modifie le signal et plus on risque de dégrader la qualité. La différence entre la moyenne et la médiane semble montrer une assez forte variation au sein des signaux de la base de données. Ceci est tout à fait compréhensible, puisque les signaux musicaux sont très différents les uns des autres, ils ont chacun un débit limite différent à partir duquel les dégradations dues au tatouage deviennent audibles. Les différences qui vont affecter les résultats ici sont la puissance du signal, puisque cette grandeur joue un rôle important dans le modèle psychoacoustique, et aussi la répartition fréquentielle de cette puissance. Au niveau des différences entre les longueurs de trames, elles sont très faibles, avec cependant ce qui semble être un léger avantage pour une longueur de 2048. Ce résultat est aussi encourageant, puisque 2048 est une longueur souvent utilisée pour la MDCT dans le cas des signaux audio échantillonnés à 44.1 kHz, c'est en particulier la longueur des trames longues utilisée dans la norme MPEG AAC. Le fait que les variations soient aussi faibles entre les différentes longueurs de trames est probablement dû au fait que le modèle psychoacoustique possède un contrôle du pré-écho très fort. C'est-à-dire que dans le cas d'attaques franches de notes, le seuil de masquage autour de l'attaque va être volontairement affaibli afin d'être sûr que le phénomène de pré-écho ne soit pas audible, au prix d'un débit légèrement réduit dans cette zone.

Le premier jeu de figures ne permet cependant pas de s'assurer que les différents signaux de la base de données suivent bien tous cette même tendance, ou au moins que les courbes d'ODG sont bien décroissantes en fonction du débit d'insertion. C'est pourquoi dans le second jeu de figures 6.2, nous représentons la statistique de la perte de qualité entre deux débits d'insertion successifs. Si l'on note $\mathbf{d}$ le vecteur à 9 éléments des débits d'insertion



(a) N=512 (b) N=1024



(c) N=2048 (d) N=4096

FIGURE 6.2 – Statistique des gains d'ODG suivant l'incrément de débit d'insertion pour l'implémentation améliorée

auxquels les ODG ont été calculées, on définit le vecteur de différence de qualité audio **p** par :

$$\forall k \in [0,8],\ p(k) = \mathrm{ODG}(d(k+1)) - \mathrm{ODG}(d(k)) \tag{6.1}$$

Le but est de voir si la différence de qualité est bien négative. Pour représenter la statistique de ces vecteurs de différence de qualité, nous utilisons une représentation en boîte à moustaches pour chacun des huit intervalles (notés de 1 à 8 sur la figure). Chaque boîte à moustaches représente la variation au sein des éléments de la base de données BD1. Plus exactement, la barre rouge au centre d'une boîte représente la médiane ou second quartile Q_2, les extrémités des boîtes représentent les premiers et troisième quartiles Q_1 et Q_3 [1], les croix rouges représentent les valeurs statistiquement aberrantes [2] et les moustaches représentent les valeurs maximales et minimales des données qui ne sont pas considérées comme étant aberrantes. On remarque tout d'abord qu'à part deux valeurs, toutes les différences de qualité sont bien négatives, ce qui est cohérent et montre bien que même en regardant signal par signal, la qualité audio est une fonction décroissante du débit d'insertion. Des valeurs de plus en plus étalées (représentées par des boîtes et/ou des moustaches de plus en plus grandes), lorsque l'on progresse dans les intervalles, montre encore une fois que certains signaux restent à une ODG nulle pendant que d'autres commencent à voir leur qualité se dégrader, suivant leurs caractéristiques psychoacoustiques. Pour s'en convaincre totalement, un jeu de courbes débit / qualité pour certains morceaux de la base de données BD1 (longueur de trame 2048) est présenté figure 6.3, en précisant le style musical de chaque élément.

Ces deux premiers jeux de courbes nous montrent que notre système de tatouage adaptatif permet de tatouer des signaux audio commerciaux avec une quasi-transparence au niveau de la qualité à des débits se situant généralement aux alentours de 200kb/s/c et pouvant aller jusqu'à plus de 250 kb/s/c suivant le contenu audio.

6.5 Validation de l'algorithme PEAQ et du MPA

Comme nous l'avons déjà expliqué en section 3.5, l'algorithme PEAQ a été développé pour avoir des mesures objectives de la qualité des codeurs audio. Bien que notre système de tatouage ait de grandes similarités de fonctionnement et de modification des signaux avec les codeurs audio perceptuels, il n'est cependant pas garanti que les mesures d'ODG données par l'algorithme PEAQ soient adaptées à notre système. Nous avons donc réalisé deux expériences, l'une pour valider l'utilisation de PEAQ et l'autre pour tester le MPA utilisé.

1. Le premier quartile Q_1 est la valeur des données telle qu'un quart des données est inférieur à Q_1, et le troisième quartile Q_3 est la valeur des données telle qu'un quart des données est supérieur à Q_3.

2. Les données aberrantes sont ici les données qui sont plus faibles que Q_1 (resp. plus élevées que Q_3) d'au moins une fois et demi l'écart inter-quartiles ($Q_3 - Q_1$), ce qui est un critère classique.

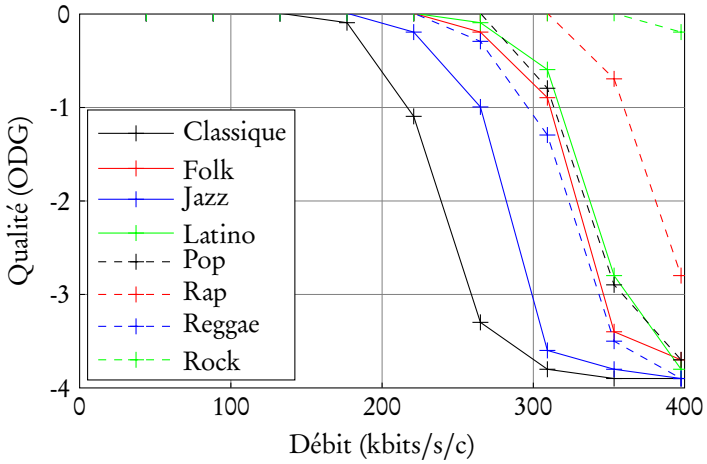
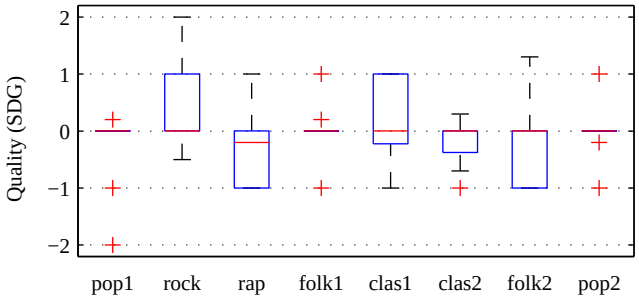


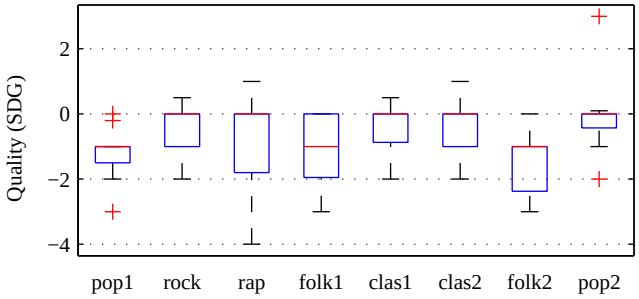
FIGURE 6.3 – Exemple de courbes débit/qualité pour des signaux de styles musicaux différents faisant partie de la base de données BD1

6.5.1 Première expérience

Afin de valider l'utilisation de cet algorithme dans le cadre de notre système de tatouage, nous avons donc réalisé un test d'écoute. Étant donné que l'utilisation principale de notre système de tatouage est prévue dans le cas d'une très bonne qualité audio, proche de la limite d'inaudibilité, c'est dans cette gamme que nous allons concentrer notre test. Nous avons à ces fins utilisé la base de données réduite BD2, et pour chacun des extraits de cette base, nous avons recherché le débit maximal à ODG nulle et le débit qui donne l'ODG le plus proche possible de −1 (pour un total de 16 signaux tatoués, chaque signal ayant une version tatouée avec une ODG cible de 0 et une version tatouée avec une ODG cible de −1). Nous avons ensuite fait passer un test d'audition en suivant les recommandations de l'ITU pour détecter des différences entre des signaux de haute qualité [IR97]. Nous avons choisi une longueur de trame de 2048, étant donné que cette valeur semble donner des résultats légèrement meilleurs que les autres tailles testées. Le test était donc de type ABC, c'est-à-dire que pour chacun des 16 cas générés (1 signal parmi les 8 de BD2 et une ODG cible, 0 ou −1), l'auditeur pouvait écouter la référence X (le signal non tatoué), et les signaux A et B, autant de fois qu'il le souhaitait. Pour les signaux A et B, à chaque fois était tiré au hasard lequel des deux était égal à la référence X et lequel des deux était le signal tatoué. Le sujet devait alors donner une note (c.f. table 3.2) à chacun des signaux A et B, sachant qu'un des deux était le signal original. Il disposait tout d'abord d'une phase d'apprentissage, pour une durée selon sa convenance, durant laquelle il pouvait écouter plusieurs signaux (ne faisant pas partie de la base de données BD2) modifiés ou non par le système de tatouage afin de repérer et de s'habituer aux déformations caractéristiques du système. Conformément aux recommandations, nous avons effectué une post-sélection des sujets pour garder uniquement ceux qui ont réussi à noter en moyenne correctement les signaux (c'est-à-dire ceux qui ont



(a) MDCT, ODG=0.



(b) MDCT, ODG=−1.

FIGURE 6.4 – Résultat du test d'écoute subjectif

généralement mis une meilleure note pour le signal original que pour le signal tatoué), à l'aide d'un test t. Parmi les 20 personnes qui ont passé l'expérience, uniquement 11 ont été sélectionnées par le test t, ce qui indique que les différences sont généralement très difficiles à détecter, ce qui est normal puisque les ODG visées étaient très faibles (0 et −1).

La figure 6.4 comporte deux graphiques en boîtes à moustaches, un pour les ODG cibles de 0 et l'autre pour les ODG cibles de −1. Chaque boîte à moustache représente la statistique, suivant les participants et pour un signal particulier de la base de données BD2, de la SDG (c'est-à-dire la différence entre la note attribuée au signal tatouée et celle attribuée au signal original). Pour le cas d'une ODG cible de 0, on remarque que mis à part pour l'extrait de rap, la note médiane des sujets est toujours 0. De plus, si l'on met de côté les valeurs aberrantes, aucune SDG n'est plus faible que −1, ce qui nous porte à croire que les modifications pour une ODG cible de 0 sont bien inaudibles, sauf peut-être pour les détenteurs d'une très fine oreille, et pour ces personnes les modifications ne sont pas gênantes. Pour le cas d'une ODG cible de −1, les résultats sont moins constants suivant les extraits, bien qu'ils semblent osciller autour de la valeur −1. De plus, la SDG est généralement plus élevée que l'ODG cible de −1, ce qui laisse penser que dans la plupart des cas l'algorithme PEAQ appliqué à notre système de tatouage est plus sévère qu'un auditeur moyen.

Cette expérience nous conforte donc dans l'utilisation de l'algorithme PEAQ pour le cas de notre système de tatouage, notamment lorsqu'il vise à ne pas ou très peu modifier la

Extrait	Débit basique MPA (kb/s/c)	Débit à ODG=0 (kb/s/c)
pop1	270	260
rock	356	360
rap	270	260
folk1	268	260
clas1	265	240
clas2	164	140
folk2	234	230
pop2	253	240

TABLE 6.2 – Débit pour le réglage basique du MPA et débit maximal à ODG nulle pour les éléments de BD2

qualité du signal hôte.

6.5.2 Deuxième expérience

Nous allons décrire maintenant un seconde expérience qui a été effectuée afin de montrer une propriété intéressante du modèle psychoacoustique utilisé dans notre système de tatouage. Pour cette expérience, nous avons utilisé la base de données réduite BD2, et nous avons d'abord effectué une insertion avec le débit de base proposé par le modèle psychoacoustique (c'est-à-dire non translaté), puis nous avons cherché le débit maximal qui donne une ODG de 0 (avec une précision de 10 kb/s/c). Les résultats sont représentés dans le tableau 6.2. On peut voir ici que le modèle psychoacoustique, bien que très simple, n'est pas trop imprécis puisqu'il permet d'avoir immédiatement une approximation du débit limite pour lequel le tatouage est inaudible, ce qui est très intéressant pour l'application du projet DReaM où l'on cherche à ne pas modifier la qualité du mix. Notons aussi les très bons résultats obtenus par le système de tatouage. En effet, on atteint des débits aux alentours de 250 kb/s/c pour presque tous les morceaux, et le morceau de rock permet même une insertion à 350 kb/s/c qui, on le rappelle, représente 50% du débit du signal audio.

Troisième partie

2ème implémentation améliorée

Chapitre 7

Améliorations du système de tatouage

Sommaire

7.1 Introduction

Nous avons vu dans la partie précédente une implémentation relativement simple d'un système de tatouage inspiré des codeurs audio perceptuels. Ce système offre de très bonnes performances par rapport au cahier des charges que nous avons fixé suivant les besoins du projet DReaM. Cependant, plusieurs points critiques peuvent être améliorés, comme nous allons le montrer avec cette deuxième implémentation.

Tout d'abord, nous avons vu dans la première implémentation que la source de bruit qui affecte le tatouage est intrinsèque à notre système, puisqu'elle provient de la conversion au format PCM 16 bits. Notre signal hôte original peut être vu comme un signal à valeurs entières (les échantillons PCM 16 bits), mais la MDCT et son inverse sont des transformées à valeurs réelles. Ainsi, une fois les coefficients MDCT modifiés par le tatouage, lorsque la IMDCT est appliquée les valeurs ne sont pas forcément entières. Une amélioration envisagée est alors d'utiliser une transformée bijective à valeurs entières qui ait des propriétés similaires à la MDCT, afin d'éliminer le bruit intrinsèque à notre système.

Ensuite, nous avons vu qu'un problème important dans la première implémentation est l'absence d'une synchronisation interne explicite. En effet, à cause de la double extraction en cascade au décodeur, la taille du message décodé (en l'occurrence le nombre de mots de code de 32 bits) peut être différente de la taille du message inséré : on ne sait alors plus où se situent les mots de code manquants ou ajoutés. Afin de pallier ce problème, nous proposons une technique de synchronisation interne simple, qui consiste à ajouter à l'information à insérer une faible quantité d'information supplémentaire afin de réaliser cette synchronisation. Ce problème de synchronisation interne n'apparaît normalement que dans le cas où le système de tatouage est utilisé pour transmettre un message de taille conséquente réparti au mieux sur la totalité du morceau suivant l'analyse psychoacoustique. Si le système est utilisé pour transmettre de l'information de façon synchrone, c'est-à-dire quand l'information insérée dans une trame est relative à cette même trame uniquement, le problème est moins grave. En effet, cela n'a pas de répercussion sur les trames suivantes. Dans ce cas là, il peut néanmoins être intéressant de pouvoir synchroniser directement sur un début de trame quand on se déplace dans le morceau. Ceci est particulièrement bienvenu pour le projet DReaM, puisque dans certaines techniques de séparation de source informée, le traitement est effectué trame par trame, avec de l'information additionnelle transmise relative à chaque trame. Ainsi, un auditeur peut avoir envie de passer une partie du morceau, et donc il est intéressant de commencer la séparation dès que l'utilisateur s'est déplacé dans le morceau. Nous allons donc présenter un système de synchronisation qui va permettre de localiser les débuts de trames pour permettre un décodage à partir de n'importe quel moment du morceau (à une demi-largeur de trame près).

La dernière modification que nous allons introduire va se situer au niveau de l'information additionnelle. Nous avons vu que, dans l'implémentation basique, une bande de fréquences est réservée uniquement à la transmission des charges utilisées en basses fréquences. Ceci ne semble pas être un choix optimal, d'autant que d'un point de vue psychoacoustique, on sait que dans les très hautes fréquences il est possible de modifier de manière conséquente

(et donc dans notre cas d'insérer beaucoup d'information dans) les hautes fréquences. Or dans l'implémentation basique nous n'insérons qu'un maximum de 3 bits par coefficient dans ces hautes fréquences. Nous allons donc montrer comment il est possible d'utiliser ces hautes fréquences pour transmettre simultanément les charges et l'information utile, grâce à la possibilité de décodage hiérarchique de la QIM.

Nous allons dans un premier temps détailler les modifications apportées au système de tatouage présenté dans la partie précédente. Nous allons tout d'abord décrire la transformée temps-fréquence qui va remplacer la MDCT classique, puis les méthodes de synchronisation proposées, et finalement la nouvelle méthode utilisée pour transmettre les charges de façon plus optimale. Dans le chapitre suivant nous retracerons les expériences qui viennent décrire les performances de ce nouveau système de tatouage amélioré. Et c'est par une étude comparative des résultats avec un système de tatouage haut débit construit suivant des contraintes similaires que nous terminerons cette partie.

7.2 Nouvelle transformée temps-fréquence

7.2.1 Introduction

Le but de la nouvelle transformée temps-fréquence que nous allons utiliser dans cette implémentation améliorée est de supprimer le bruit inhérent à la première implémentation, dû à la quantification PCM 16 bits. Les échantillons de l'hôte non tatoué peuvent être considérés comme des entiers entre -2^{15} et $2^{15}-1$. Lorsque la MDCT est appliquée (dans son ensemble, pour obtenir un plan temps-fréquence), nous avons des coefficients reéls. Nous avons vu que l'application qui à une représentation temporelle associe son plan temps-fréquence MDCT est bijective pour des réels. Lorsque les échantillons temporels sont des entiers, c'est-à-dire lorsque le signal temporel est un vecteur de $\mathbb{Z}^K$, l'espace d'arrivée du plan temps-fréquence est donc un sous-espace de $\mathbb{R}^K$ isomorphe à $\mathbb{Z}^K$. Cependant, lorsque l'on modifie les coefficients MDCT par la technique de QIM pour insérer l'information, rien ne nous dit que l'on reste dans ce sous-espace (qui est très complexe et semble difficile à identifier). Par conséquent, lorsque l'on retourne dans le domaine temporel après tatouage on ne retrouve pas forcément des valeurs entières (c'est pour cela qu'il faut quantifier le signal tatoué pour le garder au format PCM). D'où l'idée d'utiliser une transformée qui soit une bijection de $\mathbb{Z}^K$ dans lui même, et qui approche la MDCT. Une telle transformée existe, en l'occurrence la IntMDCT [GHKB02], qui est une approximation entière de la MDCT classique.

L'IntMDCT est une transformée bijective de $\mathbb{Z}^K$ dans $\mathbb{Z}^K$, qui fait partie des transformées ITI (*integer-to-integer*) bijectives. Les première transformées ITI développées ont été des ondelettes [DS96], et la technique centrale d'approximation, le *lifting-step*, a été adaptée ensuite à d'autres transformées [GHKB02, BVdE92]. Il existe plusieurs méthodes qui permettent de calculer des transformées ITI à partir d'une transformée usuelle, que nous ne développerons pas puisqu'en dehors du cadre de cette thèse. Nous allons simplement présenter la transformée IntMDCT que nous utilisons pour notre système de tatouage et

l'algorithme utilisé pour la calculer, extrait de [GHKB02].

Le calcul de l'IntMDCT n'est pas simplement une approximation directe de la formule de MDCT présentée en section 3.3.3. En effet si l'on représente cette formule sous forme d'un produit matriciel avec une matrice $M \in \mathscr{M}_{N,N/2}$, alors cette matrice n'étant pas carrée, il ne va pas être possible d'approximer une transformée bijective directement. Il faut prendre en compte le recouvrement temporel (ou OLA pour *overlap-add* en anglais) car c'est uniquement l'application qui à une représentation temporelle associe un plan temps-fréquence qui est bijective, et non la MDCT considérée pour une trame seule.

Nous allons tout d'abord expliquer une technique générale pour obtenir des approximations ITI que nous allons utiliser pour la IntMDCT. Nous allons pour cela utiliser la décomposition de la MDCT en DCT-IV, vue en section 3.3.3, et montrer que l'IntMDCT se décompose en deux approximations ITI, une pour la DCT-IV et une pour le repliement-fenêtrage. Finalement nous discuterons des implications de cette transformée par rapport à la technique d'insertion par QIM dans notre système de tatouage.

7.2.2 Technique d'approximation ITI

Il existe plusieurs techniques pour obtenir des approximations ITI de transformées. Le principe général de celle que nous utilisons pour l'IntMDCT (issu de [GHKB02]) est de décomposer la matrice de la transformée que l'on veut décomposer (cette matrice devant être orthogonale) en un produit de plusieurs matrices simples, qui peuvent être :

- des matrices de permutation, avec éventuellement changement de signe de certains éléments,
- des matrices diagonales par bloc, chaque bloc étant une matrice du même type que décrit précédemment ou une matrice de rotation 2×2.

Pour les permutations avec éventuellement changement de signe, il n'y a rien à approximer : en effet un vecteur d'entiers multiplié par ce type de matrices ou son inverse restera entier. Pour les autres types de matrice, il suffit de savoir construire une approximation ITI d'une matrice de rotation 2×2. C'est pour ces matrices que nous allons utiliser le *lifting-step*, technique développée à l'origine pour les transformées en ondelettes ITI [DS96]. Le principe est assez simple, il se base sur la décomposition suivante des matrices de rotation 2×2 :

$$R_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \frac{1-\cos(\theta)}{\sin(\theta)} & 1 \end{pmatrix} \begin{pmatrix} 1 & -\sin(\theta) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \frac{1-\cos(\theta)}{\sin(\theta)} & 1 \end{pmatrix} \tag{7.1}$$

Cette décomposition n'est bien sûr valable que si $\sin(\theta) \neq 0$. Si $\sin(\theta) = 0$ la matrice est une matrice de permutations avec éventuellement changement de signe, il n'y a donc pas besoin de décomposition. Une fois cette décomposition faite, des approximations ITI vont être réalisées. Pour expliquer cette technique, prenons l'exemple d'une matrice :

$$L_a = \begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix} \tag{7.2}$$

avec $a \in \mathbb{R}$. Les matrices utilisées dans la décomposition des matrices de rotations sont du même type que cette matrice L_a ou que sa transposée, cas pour lequel les explications suivantes sont toujours valables. Nous avons immédiatement :

$$L_a^{-1} = \begin{pmatrix} 1 & 0 \\ -a & 1 \end{pmatrix} = L_{-a} \tag{7.3}$$

qui est elle aussi du même type. La matrice L_a représente l'opérateur suivant :

$$\begin{aligned} \mathscr{L}_a : \mathbb{R}^2 &\longrightarrow \mathbb{R}^2 \\ (x,y) &\longmapsto (x, y+ax) \end{aligned} \tag{7.4}$$

Nous allons alors définir une approximation ITI bijective de $\mathscr{L}_a$:

$$\begin{aligned} \mathrm{int}\mathscr{L}_a : \mathbb{Z}^2 &\longrightarrow \mathbb{Z}^2 \\ (x,y) &\longmapsto (x, y+[ax]) \end{aligned} \tag{7.5}$$

où $[.]$ désigne l'opération d'arrondi à l'entier le plus proche. Cette décomposition est cohérente car :

$$\begin{aligned} \mathrm{int}\mathscr{L}_{-a} \circ \mathrm{int}\mathscr{L}_a(x,y) &= \mathrm{int}\mathscr{L}_{-a}(x, y+[ax]) && (7.6) \\ &= (x, y+[ax]+[-ax]) && (7.7) \\ &= (x,y) && (7.8) \end{aligned}$$

Autrement dit :

$$\mathrm{int}\mathscr{L}_a^{-1} = \mathrm{int}\mathscr{L}_{-a} \tag{7.9}$$

ce qui est similaire au cas des matrices L_a et L_{-a}. Notons que les opérateurs ITI définis ici ne sont pas linéaires, et n'ont donc pas de forme matricielle. Ils peuvent cependant être décomposés en un produit matriciel suivi d'une opération d'arrondi sur le résultat.

7.2.3 Approximation ITI de la MDCT

Nous avons vu en section 3.3.3 que la MDCT considérée pour une trame n'est pas inversible, car il faut tenir compte du recouvrement. Nous reprenons ici les notations de la section 3.3.3, et nous avons la décomposition de la matrice de MDCT suivante :

$$\mathbf{M}_{\mathrm{MDCT}} = \mathbf{DFHa} \tag{7.10}$$

La matrice de DCT-IV **D** étant orthonormale, nous pouvons en calculer une approximation ITI en utilisant une décomposition (en l'occurrence celle donnée dans [Wan84]) et les techniques d'approximation décrites précédemment. Le produit **FHa** n'est lui pas approprié directement pour une approximation ITI puisque la matrice n'est même pas carrée. Il va

donc falloir prendre en compte le recouvrement de 50% des trames. Considérons deux trames consécutives $\mathbf{x}_1$ et $\mathbf{x}_2$ que nous notons en sous-vecteurs de longueur $N/4$:

$$\mathbf{x}_1 = \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \mathbf{s}_3 \\ \mathbf{s}_4 \end{pmatrix} \qquad \mathbf{x}_2 = \begin{pmatrix} \mathbf{s}_3 \\ \mathbf{s}_4 \\ \mathbf{s}_5 \\ \mathbf{s}_6 \end{pmatrix} \tag{7.11}$$

On a alors :

$$\mathbf{FHax}_1 = \begin{pmatrix} -\mathbf{RHa}_3\mathbf{s}_3 - \mathbf{Ha}_4\mathbf{s}_4 \\ \mathbf{Ha}_1\mathbf{s}_1 - \mathbf{RHa}_2\mathbf{s}_2 \end{pmatrix} \tag{7.12}$$

$$\mathbf{FHax}_2 = \begin{pmatrix} -\mathbf{RHa}_3\mathbf{s}_5 - \mathbf{Ha}_4\mathbf{s}_6 \\ \mathbf{Ha}_1\mathbf{s}_3 - \mathbf{RHa}_2\mathbf{s}_4 \end{pmatrix} \tag{7.13}$$

On remarque ici que la partie commune aux deux trames apparaît à la fois dans la partie supérieure de la matrice $\mathbf{FHax}_1$ et dans la partie inférieure de $\mathbf{FHax}_2$. Or ces deux parties peuvent être obtenues par le produit de la partie commune et d'une matrice $\mathbf{O}$:

$$\mathbf{O}\begin{pmatrix} \mathbf{s}_3 \\ \mathbf{s}_4 \end{pmatrix} = \begin{pmatrix} -\mathbf{RHa}_3\mathbf{s}_3 - \mathbf{Ha}_4\mathbf{s}_4 \\ \mathbf{Ha}_1\mathbf{s}_3 - \mathbf{RHa}_2\mathbf{s}_4 \end{pmatrix} \tag{7.14}$$

$$\mathbf{O} = \begin{pmatrix} -\mathbf{RHa}_3 & -\mathbf{Ha}_4 \\ \mathbf{Ha}_1 & -\mathbf{RHa}_2 \end{pmatrix} \tag{7.15}$$

On peut alors facilement montrer (voir annexe A.5) que cette matrice $\mathbf{O}$ est orthogonale si les fenêtres d'analyse et de synthèse sont identiques ($\mathbf{Ha} = \mathbf{Hs} = \mathbf{H}$), et on peut ensuite calculer une approximation ITI de cette matrice $\mathbf{O}$.

Pour résumer, nous avons décomposé le calcul de la MDCT d'un signal temporel de la façon suivante, schématisé figure 7.1 :

1. Chaque demi-trame MDCT est multipliée par la matrice $\mathbf{O}$.
2. Les parties supérieures et inférieures des vecteurs résultants sont réarrangées.
3. Multiplication finale par une matrice de DCT-IV.

L'étape 2 n'est qu'un réarrangement de vecteurs, qui est parfaitement inversible, et nous avons expliqué que les étapes 1 et 3 peuvent être approximées par des transformée ITI, ce qui donne au final l'IntMDCT. N'oublions pas que pour obtenir une reconstruction parfaite de la totalité du signal il faut prendre en compte les deux demi-trames des extrémités, comme nous l'avons vu en section 3.3.3.

7.2.4 Implications sur la QIM

Dans le cas de la première implémentation basique, nous avons expliqué comment les charges et la distance minimale entre les régions d'insertions Δ_Q étaient calculées, en fonction

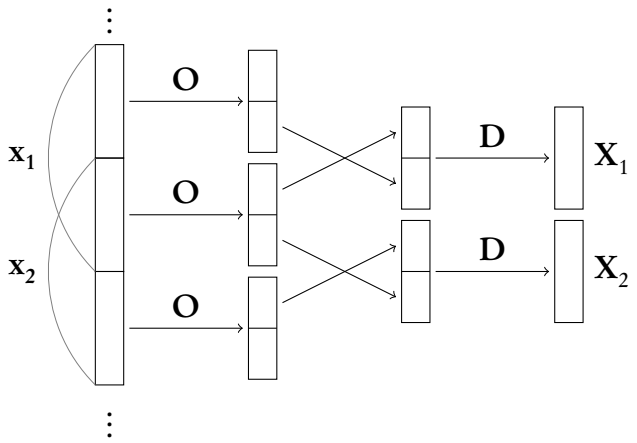


FIGURE 7.1 – Schéma de décomposition de la MDCT exploité pour le calcul de la IntMDCT

des contraintes d'inaudibilité et du taux d'erreur symbole cible. Étant donné qu'il n'y a pas de bruit intrinsèque au système de tatouage dans le cas de l'IntMDCT, la distance minimale entre les régions d'insertion Δ_Q n'est pas fixée par le taux d'erreur symbole cible, mais simplement par le fait que l'IntMDCT est une transformée ITI. C'est la différence avec l'implémentation basique. Autrement dit les coefficients IntMDCT tatoués doivent rester des entiers et donc $\Delta_Q = 1$. La formule pour déduire la valeur des charges suivant la valeur du masque reste quant à elle inchangée. Par mesure de comparaison, dans le cas de la MDCT un taux d'erreur symbole cible de 10^{-6} correspond à $\Delta_Q = 2,8$; et $\Delta_Q = 1$ correspond à un taux d'erreur symbole cible de $0,08$ environ ce qui est trop élevé pour quelque utilisation que ce soit.

7.3 Synchronisation

7.3.1 Introduction

Dans cette section nous allons présenter les deux techniques de synchronisation mises au point pour notre système de tatouage suivant les deux cas possibles d'utilisation.

Le premier est le cas que nous allons qualifier d'**insertion synchrone**. Dans cette configuration, on considère que l'information est insérée avec une cohérence locale, c'est-à-dire que l'information additionnelle nécessaire pour séparer les sources dans une trame donnée est insérée dans cette même trame. Ce cas est intéressant pour les contraintes de temps-réel, et peut aussi permettre à l'utilisateur de se déplacer au sein du morceau comme bon lui semble, à l'instar d'une écoute classique. Le problème du déplacement est qu'il faut alors se synchroniser sur des débuts de trames, afin de pouvoir extraire l'information tatouée.

Le second cas est celui que nous appelons **insertion asynchrone**. On considère alors

que l'information de séparation n'est pas locale mais globale, et elle doit être insérée comme un ensemble en utilisant tout le morceau. Elle doit donc être décodée entièrement avant utilisation. Le problème n'est alors plus de se synchroniser sur des débuts de trames, mais de se prémunir contre un mauvais décodage des valeurs de charges, qui entraîne le décodage d'un mauvais nombre de bits et donc la corruption de la quasi-totalité de l'information insérée (puisque l'on ne sait pas où se situent les bits supplémentaires ou en défaut). Dans la section précédente, nous avons vu que l'IntMDCT donnait une extraction sans erreur, car elle permettait de se débarrasser du bruit du à la quantification PCM. Il peut cependant arriver que quelques bits soient corrompus dans un fichier (par exemple lors de la copie d'un CD), et il vaut mieux se protéger contre ce genre de problèmes. La distinction entre ces deux cas va au-delà du projet DReaM, et il est intéressant de se doter de ces deux possibilités pour une utilisation du système de tatouage pour d'autres applications de contenu enrichi, qui peuvent utiliser du contenu local ou non.

7.3.2 Insertion synchrone

Pour l'insertion synchrone, c'est-à-dire dans le scénario où nous cherchons à nous resynchroniser sur un début de trame, nous utilisons le principe classique de checksum (utilisé par exemple dans [GYS06]). Le principe d'utilisation d'une checksum est d'ajouter de la redondance à un message qui permette de déterminer s'il a été modifié ou non. La différence avec les codes correcteurs est que les checksums ont pour but uniquement de détecter les erreurs, et non de les corriger. Dans notre cas, nous allons rajouter dans la zone d'insertion à paramètres fixes (la zone hautes fréquences dans le système précédent) une checksum en plus des valeurs de charges. Cette checksum va être calculée sur cette même information insérée à paramètres fixes. Afin de garder un système qui transmet un nombre entier de mots de code de 32 bits, nous allons utiliser une checksum elle aussi sur 32 bits (on rappelle que 32 est la taille des sous-bandes, cette checksum nécessitera donc d'augmenter de 1 dans une sous-bande la charge consacrée à l'insertion à paramètre fixe). La répartition de l'information additionnelle sera détaillée dans la section suivante. Lorsque nous nous déplaçons à un endroit aléatoire d'un morceau tatoué, nous testons toutes les trames possibles en nous déplaçant d'un échantillon à chaque fois jusqu'à trouver une trame pour laquelle la checksum est correcte. Il existe cependant des risques (très faibles, comme on le verra dans les expériences) pour qu'une trame incorrecte donne une checksum correcte. Afin de réduire cette probabilité, on peut tester la checksum sur plusieurs trames consécutives. En effet l'opération d'extraction à paramètre fixe, puis de calcul et de comparaison de la checksum est très rapide, et si la machine est assez puissante on peut la tester sur plusieurs trames consécutives sans perdre le temps-réel.

Dans les démonstrateurs et logiciels DReaM développés jusqu'à présent, notons que l'opération de synchronisation dans ces cas est réalisée par le logiciel, en effectuant tout simplement des déplacements au sein du morceau par multiples de $N/2$, afin d'être toujours sur un début de trame. Pour le cas $N = 4096$, la plus grande longueur de trame considérée, $N/2$ représente une durée de 46 ms à 44.1 kHz, ce qui est une précision largement suffisante

pour se déplacer au sein d'un morceau de musique. Ce système de checksum reste cependant très intéressant si l'on veut extraire un passage du morceau. Dans ce cas l'intérêt est que le tatouage reste utilisable même si le signal n'a pas été découpé exactement suivant la taille des trames, par exemple suite à une opération dite de *cropping*, information qui n'est pas nécessairement disponible pour l'utilisateur.

7.3.3 Insertion asynchrone

Nos nous situons ici dans le cas où nous avons un message de taille conséquente à insérer de façon globale dans le signal hôte, sans lien particulier entre l'information insérée dans une trame et la trame elle-même. Ce message est donc réparti de la manière la plus adéquate possible du point du vue psychoacoustique dans la totalité des trames du signal. Dans ce cas, il est particulièrement important d'extraire la même quantité d'information que celle insérée. Dans notre système, ce type d'erreur est dû à une mauvaise extraction des charges insérées en hautes fréquences, soit à cause des erreurs dues au bruit inhérent dans le cas de la MDCT, soit à cause de données temporelles corrompues dans le cas de la MDCT ou de l'IntMDCT, par exemple lors de la copie d'un fichier. Le but du système de synchronisation interne est donc de repérer où ces erreurs ont eu lieu, et quels ont été leurs effets, c'est-à-dire combien de mots de code de 32 bits ont été ajoutés ou supprimés par erreur, et leur localisation.

L'idée va être de transmettre une information additionnelle dans chaque trame, qui va consister en deux marqueurs : le premier va être la quantité d'information déjà extraite (dans les trames précédentes), et le second, la quantité d'information qui reste à extraire (dans les trames suivantes). Nous notons ces marqueurs $m_{t,1}$ et $m_{t,2}$ respectivement. Au décodage, une fois que l'information a été extraite dans toutes les trames, le système de synchronisation va tout d'abord calculer la somme S_t de $m_{t,1}$, $m_{t,2}$ et du nombre de mots de codes de 32 bits décodés C_t :

$$S_t = m_{t,1} + m_{t,2} + C_t \tag{7.16}$$

Pour les trames où aucune erreur n'a eu lieu, cette somme S_t doit être identique, et égale au nombre total de mots de code de 32 bits insérés. Le système de synchronisation va donc chercher la valeur S_o de S_t ayant la plus forte occurrence, et va considérer que cette valeur est le bon nombre de mots de codes de 32 bits insérés. Les trames erronées sont facilement repérées puisque ce sont celles où $S_t \neq S_o$. En observant les valeurs de $m_{t,1}$ et $m_{t,2}$ des trames alentours le système de synchronisation peut alors déduire quel était le nombre de mots de codes de 32 bits réellement insérés dans les trames mal décodées. S'il ne peut pas corriger les erreurs, il peut cependant les remplacer par une valeur arbitraire (par exemple des zéros).

L'inconvénient de cette méthode est qu'elle nécessite un débit conséquent. En effet si l'on prend un exemple plausible d'un morceau de musique stéréophonique de 5 minutes, tatoué avec un débit d'insertion de 200 kb/s/c, cela représente environ 4,6 millions de mots de codes de 32 bits, ce qui nécessiterait $2 \times 23 = 46$ bits d'information additionnelle à transmettre par trame, ce qui est une quantité non négligeable. Afin de pallier ce problème, les marqueurs $m_{t,1}$ et $m_{t,2}$ vont être transmis mais modulo une certaine valeur q. Nous allons choisir q

N	# sous-bandes	$C_{\max}$	# bits pour coder $m_{t,1}$ ou $m_{t,2}$	# trames consécutives erronées min détectable
512	8	$2^6 < 120 < 2^7$	16	2^{10}
1024	16	$2^7 < 240 < 2^8$	16	2^9
2048	32	$2^8 < 480 < 2^9$	16	2^8
4096	64	$2^9 < 960 < 2^{10}$	16	2^7

TABLE 7.1 – Table récapitulative de la méthode de synchronisation interne

arbitrairement, avec le même raisonnement que pour le système d'insertion synchrone, c'est-à-dire que nous allons utiliser 32 bits pour coder $m_{t,1}$ et $m_{t,2}$, ce qui fait 16 bits chacun et donc $q = 2^{16}$. Nous avons ainsi la même quantité d'information insérée à paramètre fixe dans les deux systèmes de synchronisation. Ce système de synchronisation n'est donc plus parfait puisque l'information est transmise modulo q, ce qui peut entraîner des erreurs, dans des cas que nous allons détailler maintenant.

Tout d'abord, rappelons que les valeurs des charges insérées à paramètre fixe sont comprises entre 0 et 15. Le nombre maximal de mots de codes de 32 bits transmis dans une trame $C_{\max}$ est donc égal au nombre de sous-bandes multiplié par 15. Une erreur dans une trame peut donc entraîner au maximum un défaut ou un excès de $C_{\max}$ mots de codes de 32 bits. Pour que les erreurs ne soient pas détectables par le système de synchronisation, il faut donc que plusieurs trames consécutives soient erronées, et que le nombre de mots de codes en défaut ou en excès dû à ces erreurs soit plus grand que q. Le nombre de trames minimum pour que cela se produise est donc dans le pire des cas de $\lceil q/C_{\max} \rceil$ (où $\lceil . \rceil$ désigne l'opération d'arrondi à la valeur supérieure). Les valeurs suivant N sont résumées dans le tableau 7.1. Par exemple pour $N = 2048$, il faudrait au minimum $2^8 = 256$ trames consécutives erronées pour que l'erreur ne puisse être détectable (correspondant à environ 6 secondes, cette durée est similaire pour toutes les longueurs de trame). En réalité, les erreurs dans les trames génèrent bien moins que $C_{\max}$ mots de codes en défaut ou en excès, ce qui fait que beaucoup plus de trames erronées consécutives sont nécessaires pour que cela soit indétectable.

7.4 Modification de la transmission des charges

7.4.1 Introduction

Dans l'implémentation basique que nous avons présentée dans la première partie nous avons réservé un certain nombre des sous-bandes localisées dans les fréquences les plus élevées, pour insérer les charges qui serviront à extraire l'information insérée dans les basses fréquences. Cependant ceci n'est pas optimal : l'insertion dans ces hautes-fréquences est réalisée en deçà de ce qui est possible psychoacoustiquement parlant, puisque l'oreille humaine est très peu sensible dans ces très hautes fréquences (en l'occurrence aux alentours de

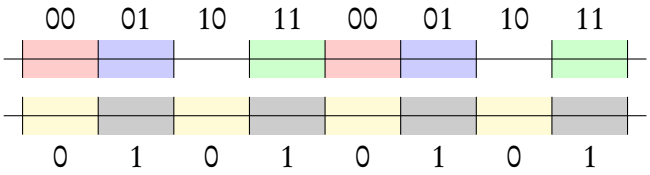


FIGURE 7.2 – Illustration du principe de décodage hiérarchique de la QIM. Régions d'attraction pour une insertion QIM sur 2 bits en haut et sur 1 bit en bas

18 kHz). D'où l'idée d'utiliser une partie seulement de ces hautes fréquences pour l'insertion des charges, et le reste conformément au modèle psychoacoustique pour avoir la possibilité de transmettre plus de mots de code de 32 bits. Ceci va être réalisé grâce à la QIM qui permet un décodage hiérarchique, comme nous allons le montrer dans un premier temps. Nous détaillerons ensuite comment nous l'utiliserons dans notre implémentation améliorée.

7.4.2 Décodage hiérarchique de la QIM

Le principe de décodage hiérarchique de la QIM est très simple et peut s'énoncer comme suit dans le cas de notre système de tatouage. Si l'on insère dans un coefficient hôte X une marque m sur C bits, alors si $C' \leq C$ on peut décoder les C' bits de poids faible de la marque m sans connaître C. Ceci est dû au fait que nous avons fait le choix d'avoir une distance entre les différents représentants constante égale à Δ_Q indépendamment de la valeur de C. Ce principe est illustré sur la figure 7.2, où l'on a représenté les différentes régions d'attraction pour une insertion QIM sur $C = 2$ bits et les régions d'attraction des quantificateurs pour une insertion sur $C' = 1$ bit. Si l'on note $R_{C,c}$ la région d'attraction du code c pour une insertion sur C bits, il est facile de voir :

$$R_{C',0} = R_{C,00} \cup R_{C,10} \tag{7.17}$$

$$R_{C',1} = R_{C,01} \cup R_{C,11} \tag{7.18}$$

On peut donc bien décoder le bit de poids faible de l'insertion sur C bits, sans connaître la valeur de C. Ceci est possible avec n'importe quelle valeur de C et C', à condition bien sûr que $C' \leq C$, et que les quantificateurs soient ordonnés suivant l'ordre du code binaire normal (cela ne fonctionne pas par exemple s'ils sont ordonnés suivant le code de Gray). C'est cette possibilité de décodage hiérarchique que l'on va utiliser dans l'implémentation améliorée de notre système afin d'optimiser la transmission des charges.

7.4.3 Nouvelle répartition des charges

Afin de transmettre les charges de manière plus efficace, nous allons utiliser la propriété de décodage hiérarchique de la QIM que nous avons décrite précédemment. C'est-à-dire qu'au lieu d'utiliser des charges fixes pour les bandes hautes fréquences et d'utiliser uniquement

N	# sous-bandes	# bits pour coder charges	# bits synchronisation	répartition bits de poids faible en HF
512	8	32	32	2
1024	16	64	32	3
2048	32	128	32	2 3
4096	64	256	32	3 3 3

TABLE 7.2 – Répartition des bits de poids faible dans les sous-bandes hautes fréquences suivant la longueur de trame N

ces fréquences pour transmettre les valeurs des charges des bandes de basses fréquences, nous allons calculer leurs charges à l'aide du modèle psychoacoustique comme pour les bandes basses fréquences. Nous utiliserons uniquement les poids faibles de ces charges dans les bandes hautes fréquences (avec une configuration fixe et connue au décodeur pour chaque valeur de longueur de trame N) pour transmettre les charges de basse fréquence. Il faudra bien évidemment que les charges calculées soient supérieures au nombre de bits dont on a besoin pour coder les charges, mais on sait que ce nombre de bits reste relativement faible et la faible acuité auditive humaine dans la région des plus hautes fréquences permet généralement l'insertion d'un nombre de bits significativement plus grand.

Le tableau 7.2 illustre la répartition de l'utilisation des bits de poids faible en haute fréquence pour l'insertion à paramètre fixe suivant la valeur de N. On voit par exemple que pour $N = 2048$, il y a 32 sous-bandes (de 32 coefficients), ce qui nécessite 128 bits pour coder les valeurs des charges (32 valeurs de charges, une pour chaque sous-bande, codée sur 4 bits). L'information de synchronisation nécessite 32 bits, pour un total à insérer de 160 bits. Les bits de poids faible utilisés sont alors les 3 bits de poids faible de la dernière sous-bande, et les 2 bits de poids faible de l'avant dernière sous-bande (on a bien $(2+3)\times 32 = 160$). La figure 7.3 représente un exemple de répartition des charges dans chacune des sous-bandes toujours pour une longueur de trame 2048 avec un système de synchronisation interne. On peut voir en rouge les bits de poids faible dans les hautes fréquences qui sont utilisés pour transmettre les charges et les données de synchronisation. Au décodeur, une première extraction QIM va donc avoir lieu avec des valeurs fixes de charges, respectivement 2 et 3 pour l'avant dernière et la dernière sous-bande. L'information extraite va ensuite être reconstituée pour donner les informations de synchronisation et les valeurs des charges de toutes les sous-bandes, qui vont permettre la deuxième extraction QIM avec des valeurs de charges adaptées à chaque sous-bande et variable suivant la trame.

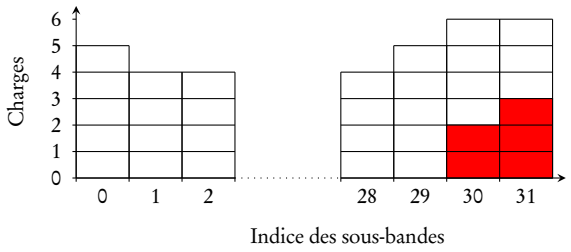
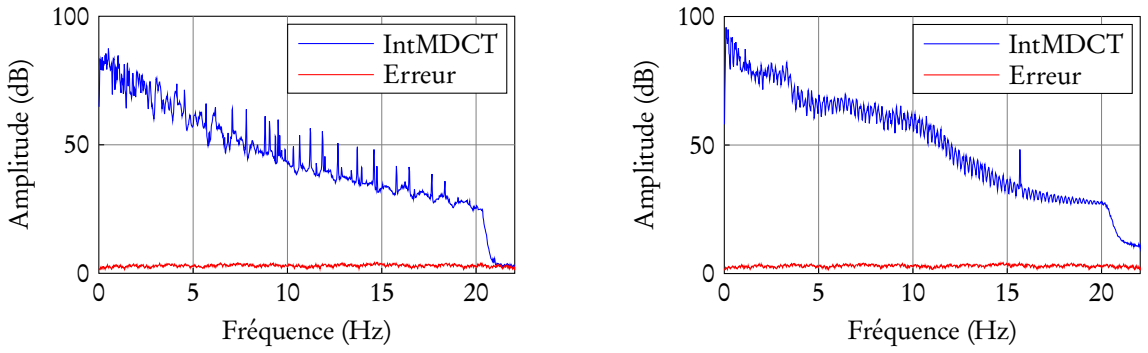


FIGURE 7.3 – Exemple de répartition des charges dans les sous-bandes, pour $N = 2048$. En rouge les bits de poids faible des bandes hautes fréquences, utilisés pour transmettre les charges et les données de synchronisation.

Chapitre 8

Secondes expériences

Sommaire



(a) Classique (b) Folk

FIGURE 8.1 – Spectre moyen de l'erreur entre IntMDCT et MDCT et spectre IntMDCT moyen

8.1 Introduction

Dans ce chapitre, nous allons décrire et discuter les résultats des expériences qui ont été faites sur cette implémentation améliorée du système de tatouage. Les bases de données audio utilisées sont les mêmes que dans la première vague d'expériences, soit BD1 et BD2, présentées à la section 6.2. Nous allons tout d'abord présenter de courtes expériences sur l'IntMDCT, pour montrer que cette transformée est bien similaire à la MDCT. Après cela, nous allons décrire les expériences menées pour tester les méthodes de synchronisation proposées. Ensuite, nous allons présenter des courbes de débit qualité, pour comparer l'implémentation améliorée avec l'implémentation basique et un autre système proposé dans la littérature ayant des caractéristiques similaires. Finalement nous allons faire une validation de l'algorithme PEAQ pour l'IntMDCT, comme nous l'avons fait pour l'implémentation basique.

8.2 Cohérence IntMDCT et MDCT

Les tests confirment bien que l'IntMDCT est une transformée ITI bijective. Cependant il reste à voir si cette transformée est bien similaire à la MDCT, qu'elle cherche à approcher. Cela peut se voir sur la figure 8.1, où l'on a tracé le spectre (le carré des coefficients) de l'IntMDCT moyenné sur tout un morceau de la BD1, et l'erreur par rapport au spectre MDCT, là aussi moyennée sur tout le morceau, avec une longueur de trame de 2048 (les deux transformées ayant exactement la même dynamique étant donné le processus d'approximation ITI, il n'y a pas d'ajustement à faire). On voit sur cette figure que l'erreur due à l'approximation entière de la IntMDCT est répartie également sur toute la bande de fréquence. Cette erreur est très faible comparée au spectre du signal, elle est en effet inférieure de plusieurs dizaines de dB.

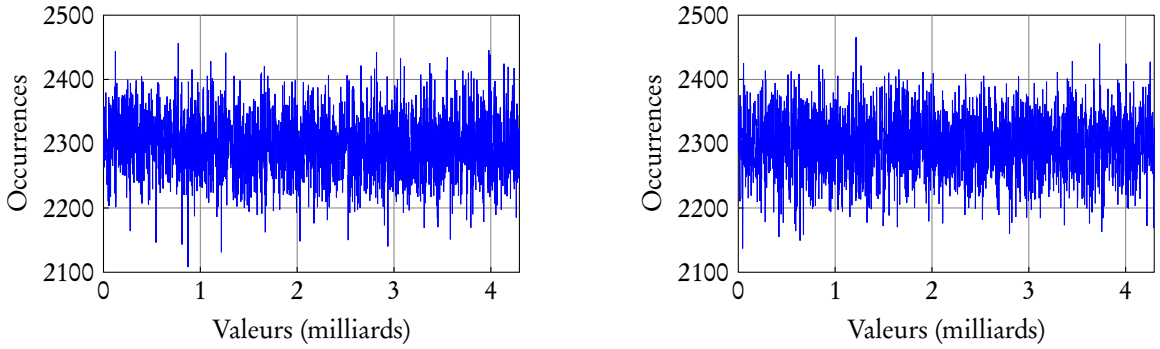
Cette approximation est donc très fidèle à la MDCT classique, tout en étant une transformée ITI bijective.

8.3 Synchronisation

Dans cette section nous allons examiner rapidement des résultats obtenus par les deux techniques de synchronisation mises en place pour notre système de tatouage. Nous allons d'abord parler de la technique dite de synchronisation externe, basée sur des checksums, puis de la technique dite de synchronisation interne, basée sur des marqueurs de la quantité d'information extraite et restante à extraire.

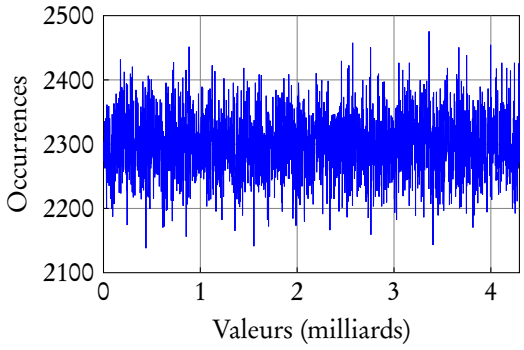
8.3.1 Synchronisation externe

Nous cherchons ici à déterminer l'efficacité de la checksum insérée avec les valeurs des charges dans la zone d'insertion à paramètres fixes. Nous avons vu que cette checksum est sur 32 bits, afin que la quantité d'information insérée par le système de tatouage reste un multiple de 32 bits, qui peut être vu comme un certain nombre de mots de code de 32 bits. L'idée derrière cette technique utilisant des checksums est que si l'on est décalé par rapport à une trame dans laquelle de l'information a été insérée, la checksum sera aléatoire (suivant une loi que l'on espère uniforme) et donc on détectera la quasi-totalité du temps que l'on est décalé. Le problème pour réaliser des expériences dans notre cas est qu'il y a plus de 4 milliards de valeurs possibles pour les checksums. Pour obtenir une estimation statistiquement consistante de l'efficacité de ce système de synchronisation, il faudrait réaliser des quantités astronomiques de tests, ce qui est difficilement réalisable, même avec un ordinateur puissant. Nous proposons alors la démarche suivante. Nous insérons de l'information grâce au système de tatouage dans certains signaux de la base de données BD1, et pour chacun de ces signaux nous calculons les checksums pour toutes les trames possibles (c'est-à-dire presque autant de trames qu'il y a d'échantillons temporels dans le signal). Nous étudions ensuite la distribution empirique des valeurs de checksums obtenues, afin de voir si l'hypothèse de répartition uniforme semble correcte. Les histogrammes des valeurs de checksums calculées pour les trames de quatre éléments de la base de données BD1 sont représentés figure 8.2. Comme on peut le constater, les histogrammes ont une moyenne assez constante, bien qu'ils semblent bruités. Ceci s'explique par le faible échantillonnage statistique : comme nous l'avons dit auparavant, il y a plus de quatre milliards de valeurs de checksums possibles, et un morceau de musique de 5 minutes représente environ vingt-cinq millions de trames si l'on considère les deux canaux. L'hypothèse que les valeurs de checksums sont distribuées uniformément lorsque nous ne sommes pas calés sur une trame d'insertion semble donc correcte. Cela signifie que la probabilité de mauvaise synchronisation est d'environ une sur quatre milliards. Cette probabilité peut de plus être réduite, comme nous l'avons déjà expliqué lors de la présentation de la synchronisation, en testant la checksum sur la ou les trames suivantes.

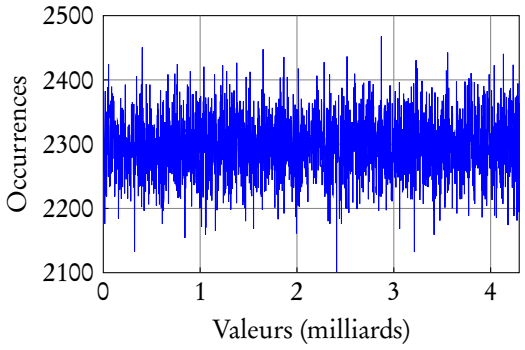


(a) Folk

(b) Latino

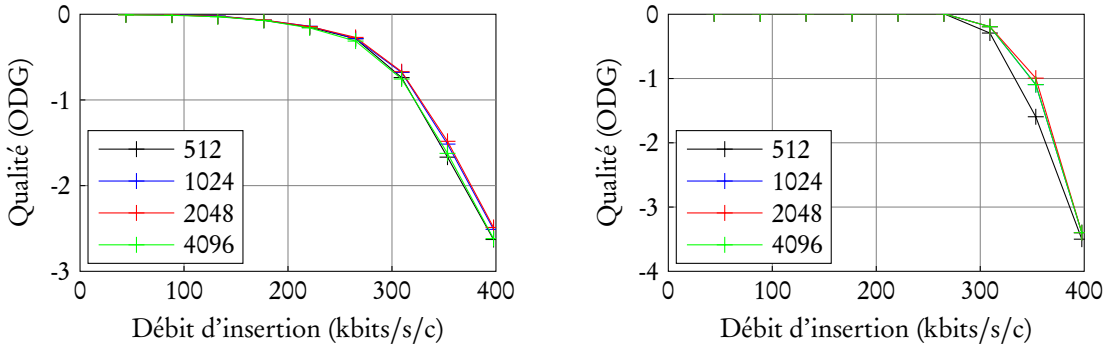


(c) Rap



(d) Rock

FIGURE 8.2 – Exemples d'histogrammes des valeurs de checksum pour quatre éléments de styles musicaux différents de la BD1



(a) Moyenne

(b) Médiane

FIGURE 8.3 – Moyenne et médiane des ODG sur la BD1 en fonction du débit d'insertion, pour les différentes longueurs de trame

8.3.2 Synchronisation interne

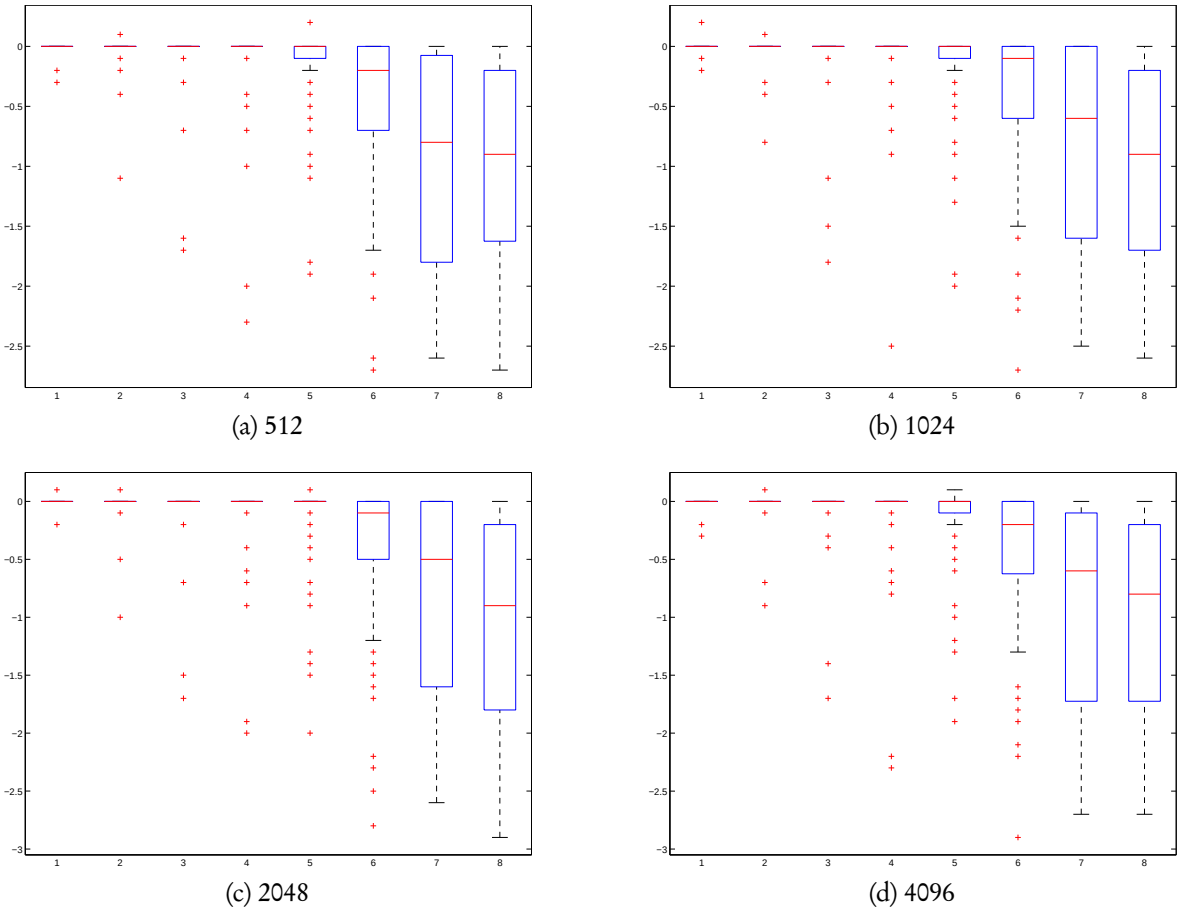
Afin de tester le système de synchronisation interne, nous avons simplement simulé de façon aléatoire des erreurs sur des bits des signaux tatoués avant d'effectuer le décodage. Les résultats sont bien cohérents avec la partie théorique : lorsque le nombre de trames consécutives corrompues n'est pas extrêmement élevé, le système parvient à repérer les données manquantes ou en surplus, et remplace correctement les zones corrompues par des valeurs arbitraires, par exemple des zéros. La limite du nombre de trames consécutives corrompues qui peut être détectée dépend du débit d'insertion puisqu'il faut que la quantité d'information insérée corrompue soit supérieure à 2^{16} (qui est la valeur du modulo pour laquelle les indicateurs m_1 et m_2 sont calculés).

Nos deux systèmes de synchronisation sont donc totalement opérationnels et nous pouvons passer à la suite de l'évaluation de notre système.

8.4 Comparaisons débit qualité

8.4.1 Comparaison entre les longueurs de trame

Nous allons commencer par comparer les performances du système amélioré suivant la longueur de trame utilisée. La base de données utilisée est BD1, et de manière identique à ce qui a été fait dans le système basique, nous représentons d'abord en figure 8.3 les médianes et les moyennes des ODG données par l'algorithme PEAQ suivant le débit. On remarque cette fois aussi une allure très cohérente des courbes, avec une qualité audio décroissante en fonction du débit d'insertion. Afin de vérifier que cette allure est bien similaire pour les différents éléments de la base de données BD1, nous réalisons comme à la section 6.4 une étude statistique des différences d'ODG lorsqu'on augmente le débit. Les résultats



(a) 512 (b) 1024

(c) 2048 (d) 4096

FIGURE 8.4 – Statistique des gains d'ODG suivant l'incrément de débit d'insertion pour l'implémentation améliorée

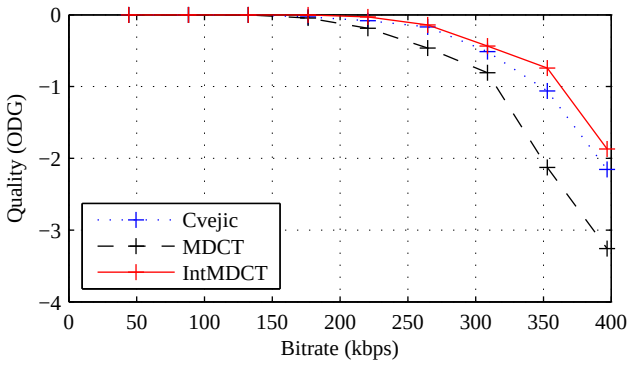
sont donnés sur la figure 8.4, et toujours comme à la section 6.4, on remarque des résultats cohérents, mis à part quelques valeurs aberrantes. Les différentes longueurs de trames ont des performances similaires, surtout du point de vue de la moyenne. Au niveau de la médiane, la longueur de trame $N = 512$ semble donner des résultats moins bons que les autres longueurs. Pour toutes les longueurs de trame, un débit médian de 260 kbits/s/c permet une insertion à ODG nulle, ce qui signifie que pour plus de la moitié des signaux une insertion à ce débit est inaudible. La médiane décroît alors pour atteindre une ODG de -1 à 350 kbits/s/c (légèrement plus tôt pour la longueur de trame $N = 512$). Pour la moyenne, les courbes décroissent lentement pour arriver à une ODG de -0.5 un peu avant 300 kbits/s/c, puis à une ODG de -1 vers 325 kbits/s/c. On constate que la statistique des gains d'ODG est bien négative à l'exception de quelques valeurs aberrantes, ce qui montre bien que la qualité décroît quand le débit d'insertion augmente. La variabilité de la perte de qualité augmente quand le débit augmente, ce qui est normal puisqu'à très bas débit la qualité reste

très bonne quel que soit le signal audio considéré ; mais chaque signal audio ayant ses propres caractéristiques psychoacoustiques, la qualité commence à décroître à des débits qui diffèrent suivant les signaux.

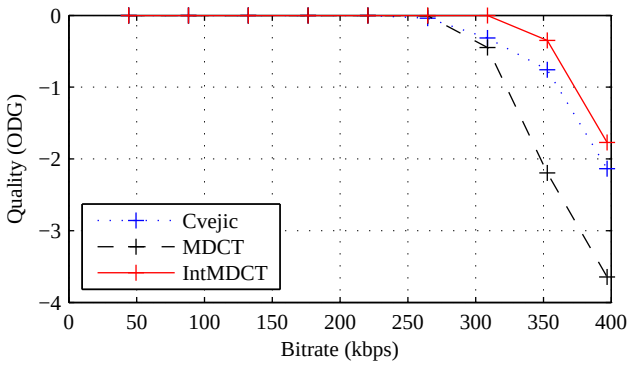
8.4.2 Comparaison inter-systèmes

Dans cette partie nous allons comparer les deux implémentations du système de tatouage, l'implémentation basique de la partie 1 et l'implémentation améliorée que l'on a présentée dans cette partie. Nous allons aussi comparer ces résultats avec un autre système de la littérature décrit dans [CS02b]. Ce système a été choisi pour comparaison car ses contraintes sont similaires, c'est-à-dire un débit très élevé et pas de prise en compte des attaques, sur des données similaires (signaux au format PCM 16 bits). C'est un système basé sur une décomposition en ondelettes, dont le principe est le suivant. Tout d'abord le signal est découpé en trames de 512 échantillons, et chaque trame est transformée en utilisant la transformée en ondelettes de Haar. Les données sont ensuite insérées grâce à une technique d'insertion LSB, le nombre de bits utilisés étant fixé et identique pour chaque coefficient d'ondelettes. 1 LSB par coefficient implique donc un débit fixe de 44.1 kb/s/c, et nous allons tester de 1 à 9 bits pour avoir des débits comparables avec ceux qui ont été testés pour les différentes implémentations de notre système. Le signal est ensuite simplement retransformé du plan temps-échelle vers une représentation temporelle, et reconverti au format PCM. Il est important de noter que ce système ne prévoit pas de contrôle de l'insertion par un modèle psychoacoustique, ce qui va probablement impliquer une qualité variable au sein d'un morceau.

Les résultats pour les deux implémentations de notre système et pour le système utilisant les ondelettes sont présentés figure 8.5. Pour les deux implémentations de notre système, nous avons réglé un taux d'erreur symbole de 10^{-4}, qui est approximativement le taux d'erreur binaire du système en ondelettes, et nous avons choisi une longueur de trames de 2048, qui est celle permettant d'obtenir les meilleurs résultats, les performances obtenues pour les différentes tailles étant toutefois relativement proches. Au niveau de la médiane, pour l'implémentation basique de notre système et pour le système en ondelettes, nous observons un palier à une ODG de 0 jusqu'à un peu plus de 250 kbits/s/c. Pour l'implémentation améliorée de notre système, ce palier se prolonge jusqu'à un peu plus de 300 kbits/s/c. La courbe de notre système basique décroît alors rapidement pour tomber à une ODG de −0.5 aux alentours de 300 kbits/s/c puis −1 vers 325 kbits/s/c. La courbe du système en ondelettes, elle, décroît plus lentement, atteignant une ODG de −0.5 vers 325 kbits/s/c, puis de −1 à un peu plus de 350 kbits/s/c. Notre système amélioré se comporte encore mieux, puisque l'ODG ne tombe à −0.5 qu'après 350 kbits/s/c, et à −1 après 375 kbits/s/c. Au niveau de la moyenne, le système en ondelettes est plus proche de l'implémentation améliorée de notre système, tout en restant ici aussi en dessous. L'implémentation basique du système reste quant à elle assez en dessous des deux autres systèmes, par environ 50 kb/s/c pour une même ODG. Le fait que les courbes de moyennes entre notre système amélioré et le système en ondelettes soient beaucoup plus proches que les courbes de médianes semble



(a) Moyenne



(b) Médiane

FIGURE 8.5 – Comparaison des ODG pour les 3 implémentations, en moyenne et en médiane

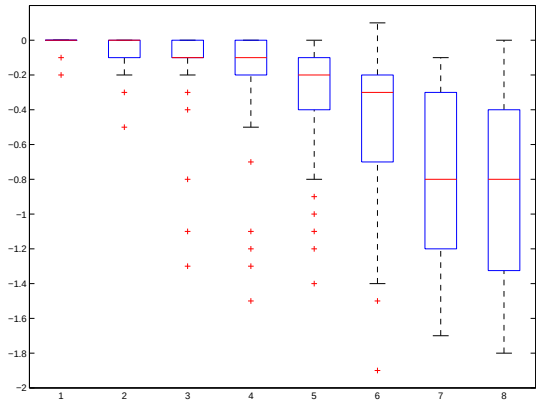


FIGURE 8.6 – Statistique des gains d'ODG suivant l'incrément de débit d'insertion pour le système en ondelettes

indiquer que la qualité du système en ondelettes est encore plus variable suivant le débit que les implémentations de notre système de tatouage. On peut voir ceci plus clairement sur la figure 8.6, qui montre la statistique. Nous pouvons en effet remarquer qu'il y a beaucoup plus de variabilité dans ses boîtes à moustaches que pour celles des deux implémentations de notre système de tatouage. En effet comme il n'y a pas de modèle psychoacoustique intégré au système en ondelettes, pour certains morceaux de la base de donnée ce système se comporte moins bien, voire nettement moins bien, notamment lorsqu'il y a des passages silencieux ou avec très peu de puissance. Avec nos deux systèmes de tatouage, pour atteindre le débit souhaité, peu (voire pas du tout) d'information va être insérée lors des passages silencieux et pour compenser cela plus d'information sera insérée dans les passages plus puissants recommandés par le modèle psychoacoustique. Ceci est évidemment la grande force du modèle psychoacoustique, dont le but est d'essayer d'insérer l'information là où elle sera la moins audible possible.

Une dernière représentation que nous pouvons tracer est la statistique de la différence d'ODG entre l'implémentation améliorée de notre système et le système en ondelettes pour chaque débit en fonction des morceaux de musique composant la base de données. Cette représentation est donnée figure 8.7. Nous voyons qu'à part quelques cas isolés, la différence d'ODG est majoritairement positive, ce qui montre bien que l'implémentation améliorée de notre système de tatouage obtient de meilleurs performances, principalement grâce au modèle psychoacoustique.

8.5 Validation de PEAQ et du MPA pour l'IntMDCT

La validation de l'utilisation de l'algorithme PEAQ pour la IntMDCT a été faite par un test d'écoute, similaire à celui réalisé pour la MDCT est présenté dans les expériences de l'implémentation basique. Les résultats sont présentés sur la figure 8.8. Les résultats sont

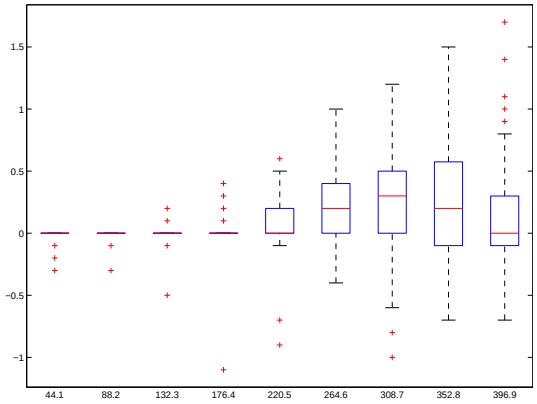
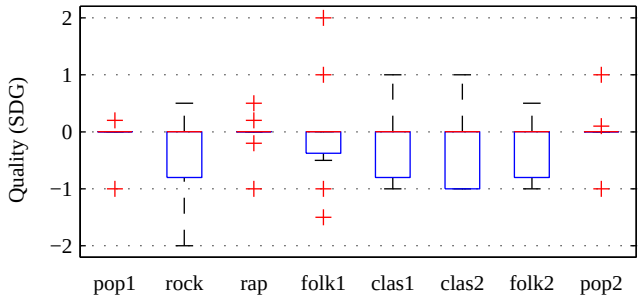
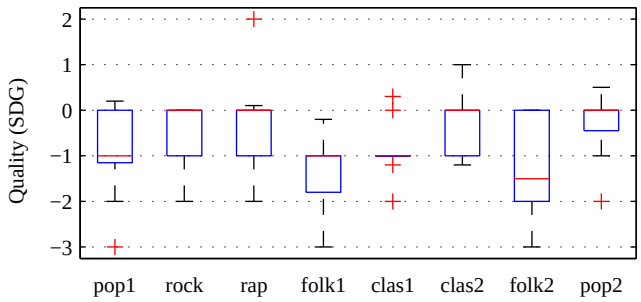


FIGURE 8.7 – Statistiques du gain d'ODG entre l'implémentation améliorée du système de tatouage et le système en ondelettes, suivant les morceaux de la base de données pour chaque débit testé



(a) ODG=0



(b) ODG=−1

FIGURE 8.8 – Résultats du test d'écoute subjectif

très proches de ceux que l'on a obtenu pour la MDCT. Dans le cas d'une ODG cible de 0, tous les morceaux ont été notés avec une ODG médiane de 0, et n'ont qu'en de très rares cas une ODG plus inférieures à −1. On peut donc dire que lorsqu'une ODG de 0 est donnée par l'algorithme PEAQ, on aura dans le pire des cas une modification audible mais non gênante et dans la plupart des cas une modification non audible. Dans le cas d'une ODG cible de −1, les résultats sont légèrement meilleurs que dans le cas de la MDCT, où les médianes sont presque toutes supérieures à −1, et certaines sont même à 0. Au sein des morceaux, les tendances sont les mêmes que dans le cas de la MDCT, et notamment le morceau folk2 a une SDG plus basse que l'ODG. Ceci peut être expliqué par le fait que l'algorithme PEAQ est une fusion de plusieurs algorithmes donnant des marqueurs de qualité, de façon à donner des notes d'ODG égales aux notes de SDG données par des auditeurs. Le signal folk2 étant très particulier, s'il est mal représenté dans la base de données utilisée pour calibrer l'algorithme, celui-ci ne sera pas parfaitement adapté, tout en n'étant pas complètement inutilisable. L'algorithme PEAQ est donc bien utilisable aussi pour le système avec la IntMDCT, surtout dans le cas de très haute qualité (ODG > -1).

8.6 Bilan

Dans cette partie nous avons présenté une version améliorée du premier système d'insertion à haut débit suivant trois axes : la représentation temps-fréquence, la synchronisation et le processus de double insertion / double décodage. L'utilisation de la IntMDCT, approximation ITI bijective de la MDCT, permet tout d'abord une extraction de l'information insérée sans erreur lorsque le signal hôte n'est pas modifié (utilisation normale par un auditeur du fichier son). Elle a aussi permis, avec l'amélioration du processus de double insertion / double décodage basé sur le décodage hiérarchique de la QIM, d'améliorer considérablement le débit d'insertion, avec entre autre une insertion à plus de 300 kbits/s/c de manière inaudible pour plus de la moitié des signaux de la base de données considérée BD1 (constituée de 96 morceaux commerciaux de styles musicaux différents). Ces débits sont bien supérieurs à ceux obtenus pour le système basique (pour lequel le taux d'erreur était de plus non nul), et sont aussi meilleurs que pour le système de la littérature considéré présenté dans [CS02b], celui-ci étant handicapé par l'absence de modèle psychoacoustique. Finalement deux systèmes de synchronisation adaptés aux deux modes d'utilisations principaux du système d'insertion ont été développés et mis en œuvre, le premier permettant de décoder des fragments du message inséré en se synchronisant sur des trames d'insertion et le second assurant l'extraction du bon nombre de bits insérés, même lorsque quelques trames du signal ont été corrompues (par exemple par des erreurs de recopie d'une piste d'un CD-A vers un ordinateur).

Quatrième partie

Système de « parcimonisation » des signaux audionumériques

Chapitre 9

Présentation du système

Sommaire

9.1 Représentations parcimonieuses

Le terme de représentation parcimonieuse d'un signal désigne une représentation du signal (dans un domaine transformé) telle qu'une grande partie de ses coefficients soient nuls (ou presque nuls). Ce type de représentation est très intéressant pour de nombreuses applications, car avoir une représentation parcimonieuse signifie que l'information est concentrée dans un nombre de coefficients significativement inférieur à la dimension de l'espace. En codage audio, l'intérêt est évident puisqu'il y a peu de valeurs non nulles à coder, les valeurs nulles étant soit non codées, soit facilement compressées par des codages de type RLE (*Run-Length Encoding*, comme dans le format JPEG par exemple). De nombreuses autres applications peuvent utiliser des propriétés de parcimonie, par exemple des applications de modifications sonores [MLAS10], de séparation de sources [MBZJ09, Gri02, PG11a], ou de transcription [AP06]. Le principe de parcimonie est de plus très fortement lié au *compressed sensing*, domaine très en vogue depuis plusieurs années.

Un des premiers points à aborder lorsque l'on travaille sur la parcimonie est la définition du terme lui-même. Lorsque l'on considère un élément $\mathbf{x}$ possédant plusieurs composantes (ce peut être par exemple un vecteur ou une matrice), la méthode intuitive pour mesurer la parcimonie est d'utiliser la « norme » ℓ^0 : [1]

$$||\mathbf{x}||_0 = \#\{n, x(n) \neq 0\} \tag{9.1}$$

où # désigne le cardinal d'un ensemble. Autrement dit, on compte simplement le nombre de composantes non nulles de $\mathbf{x}$, et plus ce nombre est faible plus $\mathbf{x}$ est parcimonieux. Cette définition stricte de la parcimonie est cependant problématique pour de nombreux cas pratiques, où les composantes peuvent être très proches de zéro sans être exactement nulles. Ceci peut être dû à la nature du signal, à des perturbations extérieures, à des problèmes de mesure, voire des erreurs de calcul numérique. Afin de pallier ce type de problèmes, différentes mesures ont été proposées pour étudier la parcimonie des signaux, telles que les normes / quasi-normes ℓ^p, la fonction tanh, le kurtosis ou le coefficient de Gini [HR08]. Étant donné que ces méthodes ne sont pas des mesures strictes de la parcimonie, il faut cependant faire très attention au cadre d'utilisation sous peine d'arriver à de fausses conclusions [KA03]. Dans cette partie on distinguera parcimonie au sens strict, c'est-à-dire en utilisant la norme ℓ^0, et au sens large, c'est-à-dire en utilisant un autre indicateur.

Dans le domaine de l'audionumérique, de nombreux travaux ont été réalisés pour étudier des représentations parcimonieuses au sens large de signaux audio [Dau06, PBD+10, NP08, TF05]. En particulier, les signaux audio sont très peu parcimonieux dans le domaine temporel, et ils le sont beaucoup plus dans le domaine temps-fréquence (TFD, MDCT…) ou dans les domaines de transformée en ondelettes.

L'organisation de ce chapitre est la suivante : tout d'abord, nous présenterons le lien entre la « parcimonisation » et le reste des travaux présentés dans ce manuscrit, et ensuite,

1. ℓ^0 peut être définie comme la limite de $(\ell^p)^{1/p}$ quand p tend vers 0, où ℓ^p désigne les normes / quasi-normes usuelles. Bien que ℓ^0 ne soit ni une norme ni une quasi-norme, l'abus de langage « norme ℓ^0 » est cependant couramment utilisé.

nous décrirons le système de « parcimonisation » développé. Dans le chapitre suivant nous présenterons les expériences mises en œuvre pour analyser les performances du système développé, et conclurons sur les contributions apportées.

9.2 « Parcimonisation » et présent travail

Une étude notable [BLED10] s'écarte quant à elle sensiblement des travaux d'étude de la parcimonie des signaux audio présentés précédemment, en utilisant un angle d'approche un peu différent. Ainsi, au lieu de rechercher un domaine dans lequel la représentation des signaux audio est parcimonieuse au sens large, les auteurs considèrent un domaine temps-fréquence où l'on sait que le signal est assez parcimonieux au sens large (en l'occurrence grâce à une transformée de Gabor [2]), et forcent des coefficients à zéro. Ce processus de mise à zéro se base sur un modèle psychoacoustique incorporant le phénomène de masquage fréquentiel, qui délivre un seuil de masquage selon des principes similaires à ceux décrits dans les sections 3.2 et 5.3. Ce seuil est ensuite translaté d'une valeur arbitraire, ce qui est à mettre en parallèle avec notre approche de la section 7.4, et les coefficients sous ce seuil sont considérés comme non-pertinents et mis à zéro (ce processus est appelé *irrelevance filter*).

Ce processus de « parcimonisation » est fortement apparenté au codage audio perceptuel. En effet dans ce dernier cas, et comme on l'a vu en section 3.2, un modèle psychoacoustique est d'abord utilisé pour mesurer la pertinence de chaque coefficient temps-fréquence, puis les coefficients sont quantifiés. Moins un coefficient est pertinent, plus il est quantifié grossièrement, voire supprimé lorsqu'il est trop peu pertinent. Toutefois, ni le modèle psychoacoustique, ni la transformée (Gabor) utilisés dans [BLED10] ne sont une implémentation et une transformée usuellement utilisées en codage perceptuel. Nous proposons donc ici de revisiter ce processus de « parcimonisation », mais en utilisant une transformée temps-fréquence et un modèle psychoacoustique dérivés directement du codage audio perceptuel. De par la quantité de travail qui se trouve derrière les codeurs audio perceptuels, nous considérons que les outils développés dans ce cadre sont particulièrement performants et semblent adaptés à cette problématique de « parcimonisation ». Cette étude de « parcimonisation » est de plus fortement reliée aux systèmes de tatouage présentés précédemment dans ce manuscrit. En effet, le fait de mettre à zéro certains coefficients dans le domaine transformé peut être vu comme une forme de tatouage caractérisant la zone mise à zéro (par exemple pour indiquer son caractère « inutile » dans un processus de séparation de sources, comme on le verra plus loin), dont le décodage est simple puisqu'il suffit de repasser le signal dans le domaine transformé et de regarder quels sont les coefficients nuls.

9.3 Présentation du système

Dans cette section nous présentons le principe général du système de « parcimonisation », inspiré de [BLED10]. Tout comme la majorité des codeurs audio perceptuels ainsi que

2. Cas particulier de STFT avec recouvrement.

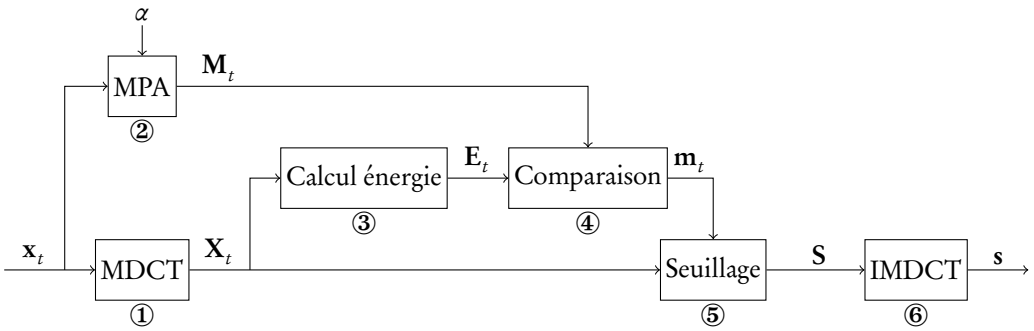


FIGURE 9.1 – Schéma de traitement d'une trame du système de « parcimonisation »

les systèmes de tatouage présentés dans ce manuscrit, le système de « parcimonisation » fonctionne par trame. Le signal est donc tout d'abord découpé en trames de longueur N et chaque trame est traitée de la façon suivante (voir figure 9.1). Tout d'abord, les coefficients MDCT $\mathbf{X}_t$ de la trame $\mathbf{x}_t$ sont calculés (bloc ①, nous reviendrons sur le choix de cette transformée par la suite), et l'énergie $\mathbf{E}_t$ correspondante est calculée (bloc ③). Parallèlement, la trame $\mathbf{x}_t$ est analysée grâce à un modèle psychoacoustique (bloc ②, ce modèle est le même que celui utilisé pour les systèmes de tatouage et inspiré du modèle simple proposé dans la norme AAC [ISO09]), qui fournit un seuil de masquage $\mathbf{M}_t$ (voir sections 3.2 et 5.3). L'énergie de la trame $\mathbf{E}_t$ et le seuil de masquage $\mathbf{M}_t$ sont alors comparés (bloc ④), résultant en un masque binaire $\mathbf{m}_t$ tel que :

$$\forall k \in \left[0, \frac{N}{2} - 1\right],\ m_t(k) = \begin{cases} 1 \text{ si } E_t(k) \geq M_t(k) \\ 0 \text{ sinon} \end{cases} \tag{9.2}$$

Ce masque binaire $\mathbf{m}_t$ sert à distinguer les composantes qui sont considérées pertinentes, et les coefficients MDCT « parcimonisés » $\mathbf{S}_t$ sont simplement obtenus en multipliant terme à terme les coefficients MDCT originaux $\mathbf{X}_t$ et le masque binaire $\mathbf{m}_t$:

$$\mathbf{S}_t = \mathbf{X}_t \odot \mathbf{m}_t \tag{9.3}$$

La trame du signal est ensuite simplement retournée dans le domaine temporel grâce à la transformée inverse, l'IMDCT, afin de reconstruire un signal temporel parcimonieux.

Comme on peut le voir sur le schéma, le modèle psychoacoustique est contrôlé par un paramètre α. De manière similaire à ce que nous avons vu au sujet des modèles psychoacoustiques pour les codeurs audio perceptuels et nos systèmes de tatouage (voir section 5.5), ce facteur α contrôle la translation du seuil de masquage (en dB). Dans ce système, il doit être fixé et sa valeur sera discutée lorsque nous présenterons les expériences qui le concernent dans le chapitre suivant.

9.3.1 Choix de la transformée

Comme nous allons le voir, les conditions liées au choix de la transformée la plus adéquate possible pour le système de « parcimonisation » sont très proches de celles requises pour le système de tatouage décrites en section 5.2.

Bien entendu, on souhaite que la « parcimonisation » soit inaudible. Cette contrainte rejoint celle d'inaudibilité du tatouage, et implique donc deux choses : que la dimension du plan temps-fréquence soit supérieure à la dimension de la représentation temporelle, et que la transformée soit à recouvrement pour éviter les effets de bloc.

Après cela, nous considérons que l'objectif du système de « parcimonisation » est d'être une sorte de pré-traitement. C'est-à-dire qu'une fois que le signal a été rendu parcimonieux et reconverti au format PCM 16 bits, on souhaite retrouver sa structure parcimonieuse lorsque l'on effectue une analyse temps-fréquence avec la transformée utilisée dans le système de « parcimonisation » (ce qui, de façon un peu surprenante, ne semble pas être une contrainte dans l'article [BLED10] ayant inspiré ces travaux). Autrement dit, une fois la représentation temps-fréquence du signal rendue parcimonieuse, il faut que l'enchaînement des opérations de reconstruction temporelle et de re-calcul du plan temps fréquence soit égal à l'identité. Cette contrainte rejoint celle de système de tatouage à faible taux d'erreur, et implique donc que la dimension du plan temps-fréquence soit inférieure à la dimension de la représentation temporelle.

Comme nous l'avons détaillé en section 5.2, la conjonction de ces deux conditions entraîne naturellement le choix de la MDCT comme transformée adéquate. Rappelons qu'à l'inverse, l'utilisation de la décomposition de Gabor / TFD avec recouvrement ne garantit pas cette invariance par aller-retour entre le plan temps-fréquence et le domaine temporel, ce qui implique que le système de [BLED10] n'assure pas une « parcimonisation » au sens strict.

Il reste finalement un point à décider, qui va dépendre des besoins spécifiques des applications faisant appel au système de « parcimonisation ». Nous avons vu que la conversion au format PCM 16 bits induit un bruit gaussien sur les coefficients temps-fréquence lors de l'utilisation de la MDCT (voir section 5.5). Ce bruit étant néanmoins très faible, si l'application faisant appel au système de « parcimonisation » peut se contenter d'une parcimonie au sens large, la MDCT peut très bien être considérée comme suffisante. En revanche si l'application a besoin de conserver une parcimonie au sens strict, l'utilisation de l'IntMDCT (qui permet de s'affranchir du bruit dû à la conversion PCM 16 bits comme on l'a vu en section 7.2) peut être préférable. On peut supposer que cette conversion PCM peut contribuer à expliquer que les auteurs de [BLED10] ne se soucient pas de la non conservation de la parcimonie au sens strict par la décomposition de Gabor, mais cela n'est qu'une supposition. En tout état de cause, comme pour notre système de tatouage, l'utilisation de l'IntMDCT garantit à la fois cette préservation de la parcimonie stricte et la compatibilité directe avec le format PCM.

Nous avons expliqué en section 7.2 que la différence entre les coefficients MDCT et IntMDCT est très faible (voir aussi section 8.2). Au niveau du nombre de coefficients

mis à zéro par le système de « parcimonisation », les deux transformées auront donc des résultats quasiment identiques, ce dont nous nous sommes assurés par des expérimentations préliminaires. Par soucis de simplicité et de rapidité d'exécutions, les expériences que nous allons présenter dans le chapitre suivant ont été réalisées avec la MDCT.

Chapitre 10

Expériences

Sommaire

α (dB)	Taux de coefficients supprimés (%)	Taux d'énergie supprimée (%)
-3,0	82,3	4,8
-4,5	78,6	3,4
-6,0	74,4	2,4
-7,5	69,4	1,7
-9,0	65,4	1,2

TABLE 10.1 – Taux de suppression de coefficients et d'énergie suivant la valeur de α

10.1 Introduction

Dans ce chapitre nous allons présenter deux expériences liées au système de « parcimonisation », séparées en deux catégories. La première concerne les performances du système par rapport au paramètre α, en terme de parcimonie et en terme de qualité audio. La seconde expérience est liée au projet DReaM, et étudie l'intérêt de la « parcimonisation » en tant que pré-traitement pour la séparation de sources informée.

10.2 Première expérience : paramètre α

Nous rappelons que le paramètre α contrôle la translation du seuil de masquage en dB. Plus la valeur de α sera élevée, plus le nombre de coefficients mis à zéro sera grand et plus la qualité audio risquera d'être diminuée. À l'inverse, plus la valeur de α sera faible, moins le nombre de coefficients mis à zéro sera élevé mais plus la qualité sera préservée. La base de données utilisée comporte 10 extraits de 5 secondes de musique commerciale, de styles musicaux variés, extraits de la base de données BD1.

10.2.1 Résultats en terme de parcimonie

Le système de « parcimonisation » a été testé pour différentes valeurs de α, et les résultats en terme de pourcentage de coefficients supprimés et d'énergie du signal supprimée sont résumés dans la table 10.1. Les valeurs de α ont été choisies avec une marge de sécurité importante. Rappelons que le réglage de base du modèle psychoacoustique, dans le cas des systèmes d'insertion, donnait un débit très légèrement supérieur au débit maximal pour un tatouage inaudible (voir section 6.5). -9 dB représente une translation importante du seuil, qui nous assure que le système de « parcimonisation » effectue bien des modifications inaudibles.

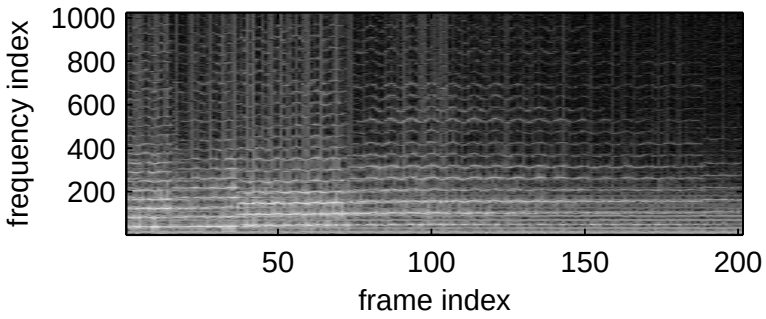
On remarque tout d'abord que le nombre de coefficients mis à zéro par le système de « parcimonisation » est extrêmement élevé, de $82,3\%$ pour $\alpha = -3$ dB à $65,4\%$ pour $\alpha = -9$ dB. Cependant la différence en énergie due à la « parcimonisation » reste très faible : de $4,8\%$ pour $\alpha = -3$ dB à $1,2\%$ pour $\alpha = -9$ dB. On comprend donc bien que, majoritairement,

le système de « parcimonisation » a mis à zéro des coefficients qui avaient déjà des valeurs très faibles. Ceci se voit aussi très clairement sur la figure 10.1, où l'on constate que le masque binaire suit très bien les harmoniques du signal. Il s'agit là d'un exemple particulier d'instrument monophonique utilisé à des fins d'illustration et ne faisant pas partie de la base de données des tests quantitatifs. D'une manière plus générale pour une scène sonore complexe, quand de forts pics fréquentiels sont présents dans une trame, les coefficients de faible valeur aux alentours seront rapidement mis à zéro par le système, et lorsque α est augmenté (parcimonie plus forte), d'autres coefficients seront mis à zéro, en commençant généralement par les hautes fréquences qui sont souvent de moindre importance du point de vue psychoacoustique.

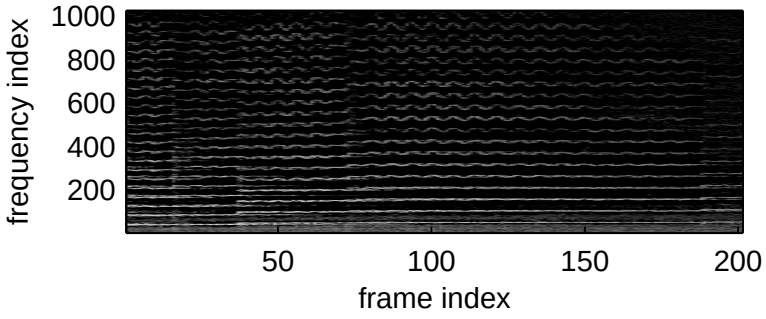
10.2.2 Résultats en terme de qualité

Afin d'évaluer les effets du système de « parcimonisation » en terme de qualité audio subjective sans démultiplier le nombre de tests d'écoute, nous avons tout d'abord effectué quelques tests d'écoute informels pour déterminer les valeurs de α intéressantes. Les premiers sujets n'ayant jamais réussi à détecter de différence entre le signal parcimonieux et le signal original pour des valeurs de α de $-7,5$ dB ou inférieures, nous avons décidé de faire un test d'écoute formel pour les valeurs de α de -3, $-4,5$ et -6 dB. 8 sujets ont alors passé le test, du type AXY. Chaque essai d'une session de test AXY se déroule comme suit : on présente au sujet un signal de référence A (le signal original), puis deux signaux X et Y qui sont le signal original et le signal parcimonieux dans un ordre aléatoire. Le sujet doit donc décider lequel parmi X et Y est identique au signal original. La différence par rapport au test ABC décrit en section 3.5 est qu'ici il n'y a pas notation des deux signaux X et Y mais uniquement un choix X=A ou Y=A. Les pourcentages de bonnes réponses pour α valant -3, $-4,5$ et -6 dB sont respectivement 78,75, 55 et 53,75. Bien que le nombre de sujets et d'essais soit trop faible pour tirer des conclusions de façon sûre, on remarque néanmoins que pour $\alpha = -3$ dB les sujets détectent assez souvent la différence, mais que pour les autres valeurs de α les résultats sont très proches de l'aléatoire. Cela indiquerait donc que pour une valeur de α autour de $-4,5$ dB, les effets du système de « parcimonisation » sont presque inaudibles. Cependant, il faut bien signaler que même si les sujets des tests d'écoute réalisés ont du temps et des signaux pour s'habituer aux types de dégradation à rechercher, ils restent généralement moins fiables que des personnes qui ont beaucoup d'entraînement. L'auteur de ce manuscrit peut par exemple réussir le test avec 100% de réussite, même pour $\alpha = -6$ dB (il ne peut toutefois pas trouver de différences pour $\alpha = -9$ dB).

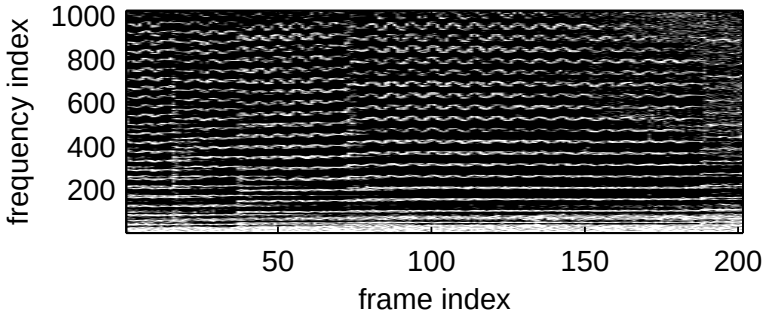
On peut donc dire sans prendre de risque que le système de « parcimonisation » avec un paramètre α réglé à une valeur de $-7,5$ ou même -9 dB permet une « parcimonisation » inaudible, qui correspond à une mise à zéro de 65 à 70% des coefficients en moyenne.



(a) Signal original



(b) Signal parcimonieux



(c) Masque binaire

FIGURE 10.1 – Illustration de la « parcimonisation » d'un signal de violon. Énergies (en dB) avant et après « parcimonisation », et masque binaire associé. Plus un pixel est clair plus la valeur est élevée.

10.3 Deuxième expérience : application à la séparation de sources informée

10.3.1 Séparation de sources informée par inversion locale

Dans cette section, nous nous intéressons à l'application du principe de « parcimonisation » à une technique particulière de séparation de sources informée (SSI) développée au Gipsa-lab [PG11a]. Un schéma résumant les principes du système de SSI est présenté figure 10.2. Ce système de SSI opère dans la configuration typique du projet DReaM : on considère qu'au codeur on dispose des sources séparées, et que le processus de mixage est contrôlé. Rappelons que le principe général du projet DReaM et de la structure codeur / décodeur SSI a été présenté à la section 1. La technique de SSI se base ici sur l'hypothèse que dans chaque point du plan temps-fréquence (la transformée utilisée étant la MDCT), un maximum de deux sources sont présentes. Ces deux sources prédominantes sont déterminées au codeur grâce à un estimateur de type oracle s'appuyant sur la connaissance des sources et de la matrice de mélange. On peut ainsi retrouver les deux sources en effectuant simplement une inversion locale 2×2, la matrice inverse étant l'inverse de la sous-matrice 2×2 composée des coefficients des deux sources pour la voie gauche et la voie droite. L'information additionnelle à transmettre par tatouage est alors l'indice des sources prédominantes pour chaque point du plan temps-fréquence ainsi que la matrice de mixage. Au décodeur, le tatouage qui contient les indices est tout d'abord extrait, puis utilisé pour réaliser la séparation de sources informée par cette inversion locale. Cette technique est appropriée pour séparer $I > 2$ sources à partir d'un mix stéréophonique linéaire instantané. En pratique, elle s'est révélée efficace pour des mélanges de 5 à 8 sources musicales, pour des chansons de type pop, rock et jazz. Le gain typique en terme de rapport de puissance Signal-à-Interférences entre l'entrée et la sortie est de l'ordre de 20 dB et plus, même s'il dépend du type de source sonore. Notons que, bien que l'hypothèse de 2 sources présentes pour un point donné du plan temps-fréquence ne soit pas strictement exacte, elle reste néanmoins pertinente. En effet, pour de nombreux signaux de musique considérés, l'énergie des deux sources prédominantes est prépondérante comparée à l'énergie totale [PG11a], et de fait, l'inversion locale fournit la plupart du temps de bons résultats.

10.3.2 La « parcimonisation » en tant que pré-traitement

L'idée que nous allons maintenant développer est alors d'utiliser le système de « parcimonisation » comme pré-traitement pour améliorer les performances et / ou l'efficacité (en terme de complexité) du système de SSI par inversion locale. Plus précisément, il s'agit premièrement de faire en sorte que l'hypothèse de deux sources maximum par point du plan temps-fréquence soit encore mieux vérifiée pour améliorer la précision de l'inversion, et deuxièmement de limiter le nombre de calculs dans les zones de très faible énergie du mix (et donc des sources qui le composent). Cette mise en tandem des deux procédés, « parcimonisation » et SSI, est bien sûr grandement facilitée par le fait qu'ils partagent tous deux la

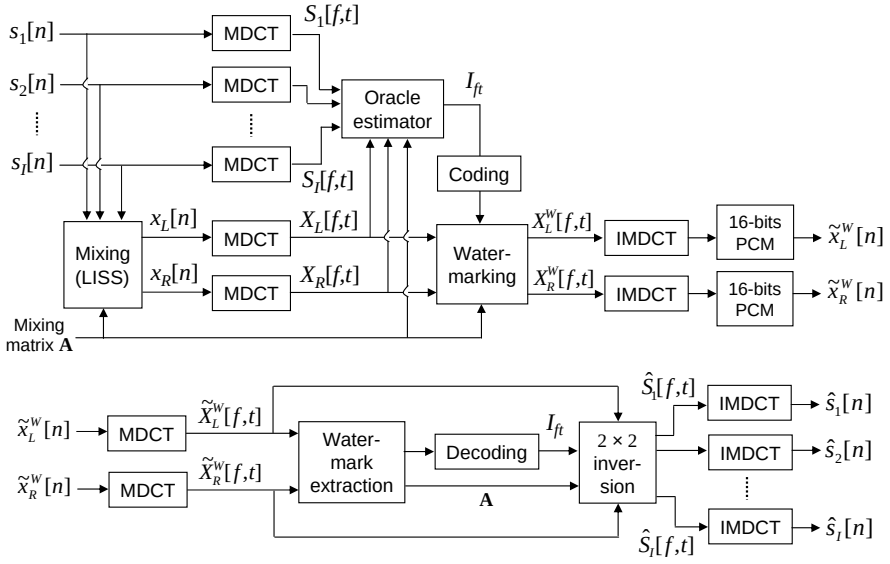


FIGURE 10.2 – Schéma du système de séparation de sources informée

même transformée temps-fréquence, en l'occurrence la MDCT, ce qui assure une forme de « compatibilité directe ».

Nous proposons ici d'utiliser le système de « parcimonisation » indépendamment pour chaque source au codeur, c'est-à-dire avant d'effectuer le mix. Nous choisissons un réglage de $\alpha = -9$ dB qui comme nous l'avons vu permet d'avoir une « parcimonisation » transparente. De plus, si le système de SSI de [PG11a] permet d'atteindre une qualité de séparation très convenable pour des applications de remixage et karaoké généralisé, un léger bruit musical est tout de même présent et masquera très largement les éventuelles perturbations dues à la « parcimonisation » avec un réglage de $\alpha = -9$ dB. Cependant, nous n'allons pas chercher à prendre une valeur de α plus grande afin d'être certain que ces perturbations sont bien négligeables. Une fois la « parcimonisation » effectuée pour chaque source, différents cas de figure sont possibles pour les points du plan temps-fréquence du signal mixé.

Cas 1 : Le coefficient MDCT de chaque source a été mis à zéro par le processus de « parcimonisation ». Dans ce cas le coefficient MDCT du mix pour les deux voies est nul et il n'y a pas de séparation à effectuer. Ainsi le décodeur considère que le coefficient MDCT correspondant pour chacune des sources estimées est nul. [1]

Cas 2 : Le coefficient MDCT d'au moins une des sources n'a pas été mis a zéro par le processus de « parcimonisation ». On distingue alors deux sous-cas :

Cas 2.1 : Le coefficient de une ou deux sources n'a pas été mis à zéro. Dans ce

1. En pratique, les matrices de coefficients MDCT étant initialisées à zéro avant inversion locale, cela signifie qu'il n'y a aucun calcul à faire pour ces coefficients.

Nombre de sources mises à zéro simultanément	Taux de coefficient (%)	Répartition de l'énergie (%)
5	32,64	0
4	25,10	3,30
3	18,50	7,45
2	12,88	18,07
1	7,75	34,73
0	3,13	36,45

TABLE 10.2 – Superpositions des sources dans le mix après « parcimonisation »

cas la séparation est réalisable grâce à l'inversion locale, et de manière exacte, contrairement au cas présenté dans [PG11a], où les sources de très faible énergie peuvent perturber l'inversion.

Cas 2.2 : Le coefficient de strictement plus de deux sources n'a pas été mis à zéro. Dans ce cas la séparation reste imparfaite, au même titre que dans [PG11a]. On considère que le bruit musical dû aux sources interférentes dans l'inversion est moins dommageable que le bruit musical qui proviendrait de la mise à zéro des coefficients non négligeables des sources autres que les deux prédominantes en forçant leur « parcimonisation ». Ce phénomène a été observé dans des tests préliminaires informels (mais n'a pas été testé en profondeur plus formellement).

Nous allons maintenant présenter la répartition de ces différents cas.

10.3.3 Résultats

Ces expériences ont été menées en utilisant 4 morceaux de musique commerciale de différents styles (pop-rock, new-wave, funk et électro-jazz), de durées variant entre 3 et 6 minutes. Ces mélanges sont constitués de 5 sources (généralement guitare, basse, batterie, voix, synthétiseur). La table 10.2 montre les effets de la « parcimonisation » au niveau de la superposition des sources dans le mix. La seconde colonne indique comment sont répartis les coefficients pour chaque configuration de nombre de sources mises à zéro simultanément, et la troisième colonne indique comment est répartie l'énergie du signal mixé parcimonieux dans ces mêmes configurations. Par exemple, pour 25,1% des coefficients 4 sources ont été mises à zéro simultanément, et ces 25,1% de coefficients représentent 3,3% de l'énergie totale du signal parcimonieux.

Nous pouvons tout d'abord remarquer qu'un peu plus de 32% des coefficients du mix sont mis à zéro, ce qui conduit à un gain direct du même ordre pour le temps de calcul puisqu'il n'y a aucune séparation à faire (toutes les sources rendues parcimonieuses dans ces bins sont nulles dans ces bins temps-fréquence et par conséquent toutes les sources estimées au décodeur le seront aussi). Ce taux de coefficients mis à zéro dans le mix est environ deux

Nombre de sources non nulles	Énergie moyenne des deux sources prédominantes (%)
3	97,92
4	96,53
5	95,70

TABLE 10.3 – Proportion moyenne de l'énergie des deux sources prédominantes par rapport à l'énergie totale dans un coefficient

fois plus faible que lorsque l'on a étudié le système de « parcimonisation » seul (c'est-à-dire appliqué directement sur le signal mixé), ce qui montre que les coefficients mis à zéro ne sont pas toujours aux mêmes bins temps-fréquence et varient suivant les sources considérées. Ceci s'explique principalement par le fait que certaines composantes d'une source peuvent être masquées par une autre source ou combinaison de sources dans le mix, mais leur suppression est audible quand on considère la source seule. Ensuite, nous pouvons remarquer qu'un peu plus de 43% des coefficients du mix contiennent uniquement 1 ou 2 sources ($25,1\% + 18,5\%$), ce qui implique que la séparation sera ici parfaite. Pour le reste des coefficients (environ 25%), la séparation n'est donc pas parfaite. Cependant on peut voir table 10.3 que dans les cas où plus de deux sources sont présentes en même temps, la proportion d'énergie provenant des deux sources prédominantes est très élevée. L'hypothèse sur laquelle se base la technique de séparation de sources informée considérée et qui est discutée dans [PG11a] semble donc confortée, et nous pouvons espérer que l'inversion ne sera pas trop perturbée. Toutefois, on peut remarquer que le gain en qualité obtenu par le pré-traitement de « parcimonisation » est assez faible. En effet, on peut voir dans la table 10.2 qu'au niveau de l'énergie, la proportion du signal qui est parfaitement séparée est très faible (un peu moins de 11%). Si elle permet un gain très intéressant en terme de complexité, la « parcimonisation » ne permet en revanche pas une grande amélioration de la qualité de séparation.

10.4 Conclusion

Dans cette partie nous avons vu qu'il est possible de rendre fortement parcimonieux un signal audio dans le domaine fréquentiel, au sens strict avec l'IntMDCT ou au sens large avec la MDCT, avec environ 65% de coefficients mis à zéro sans perte de qualité audio. Ceci est possible grâce à l'utilisation d'outils développés et affinés dans le domaine du codage audio perceptuel, et qui sont peut-être plus appropriés que la transformée de Gabor et le modèle psychoacoustique particulier utilisés dans [BLED10].

La « parcimonisation » a de plus été présentée en tant que pré-traitement pour un système de séparation de sources informée développé dans le cadre du projet DReaM [PG11a]. Les résultats montrent un gain de temps de calcul de 30% environ, et permettent d'améliorer légèrement la qualité en permettant une séparation parfaite sur une large proportion du plan temps-fréquence, même si cette portion n'est pas celle de plus forte énergie.

Conclusion générale

Dans ce manuscrit, nous avons dans les deux premières parties abordé le sujet de l'insertion à très haut débit (plusieurs centaines de kb/s/c) dans des signaux audio au format PCM 16 bits. Dans ce cadre, nous avons développé un système basique mais opérationnel, puis nous avons apporté une série d'améliorations à ce système pour le rendre plus performant. Les deux systèmes sont tous les deux basés sur la quantification d'une représentation temps-fréquence guidée par un modèle psychoacoustique, à l'instar des systèmes de codage audio perceptuels. Plus précisément, les systèmes d'insertion développés reposent sur une technique originale de double insertion / double décodage en cascade dans le domaine temps-fréquence. Les transformées temps-fréquence utilisées, la MDCT ou l'IntMDCT (approximation ITI de la MDCT) sont choisies pour leur caractéristique unique de transformée à recouvrement bijective (et pour l'absence de bruit inhérent au système d'insertion dans le cas de l'IntMDCT). Une première insertion à paramètres fixes permet de transmettre les paramètres d'une deuxième insertion à paramètres variables, optimisés par un modèle psychoacoustique afin de maximiser le débit sous contrainte d'inaudibilité. L'insertion est donc réalisée par trame, et au sein de chaque trame la répartition de l'information à insérer suivant les fréquences est contrôlée par le seuil de masquage du modèle psychoacoustique. Les systèmes d'insertion développés étant dédiés à la transmission de données, il est bien entendu possible de régler le débit d'insertion afin de l'adapter à la quantité d'information à transmettre, et ceci dans deux configurations différentes. Dans la première, chaque trame est tatouée indépendamment, et le seuil de masquage est translaté afin de s'aligner sur la quantité d'information à insérer, ce qui permet entre autres une insertion à débit constant par trame (configuration classique dans les systèmes de tatouage). Dans la seconde configuration, on considère le cas d'un message de grande taille dont l'insertion doit être répartie sur la totalité du morceau. Le débit d'insertion peut alors être adapté de façon globale, c'est-à-dire en translatant le seuil de masquage de la même valeur pour toutes les trames, résultant en des quantités d'informations insérées différentes pour chaque trame avec cependant une insertion plus correcte du point de vue psychoacoustique, car optimisée à la fois fréquentiellement et temporellement. Deux systèmes de synchronisations relatifs aux deux configurations présentées ont été développés, l'un permettant de se synchroniser sur les trames d'insertion dans le cas de la première configuration, et l'autre permettant de s'assurer que la quantité d'information extraite est bien identique à celle insérée. Des tests d'écoute subjectifs ont permis de valider l'utilisation de l'algorithme PEAQ pour mesurer la qualité

audio des signaux tatoués par nos systèmes d'insertion, ce qui a facilité la réalisation de tests intensifs afin de calculer des courbes de débit-qualité sur près d'une centaine d'extraits de 30 secondes de signaux de musique commerciaux stéréophoniques. Le système d'insertion basique permet d'obtenir un débit médian de 250 kbits/s/c environ pour une ODG de 0 qui indique une dégradation due au tatouage inaudible, et de 300 kbits/s/c environ pour une ODG de -0.5 qui indique des modifications audibles mais non gênantes. Le système amélioré quant à lui permet d'atteindre des débits médians de 300 kbits/s/c pour une ODG de 0 et de 350 kbits/s/c pour une ODG de -0.5, ce qui signifie que plus de la moitié des signaux de la base de donnée utilisée peut être tatouée à ces débits avec pas ou quasiment pas d'impact sur la qualité audio (rappelons que le débit du format PCM 16 à 44.1 kHz est d'environ 706 kbits/s/c, ce qui montre l'importance des débits atteints). Bien que ces systèmes aient été développés dans le cadre du projet DReaM et dans le but de répondre à des critères spécifiques, ils n'en restent pas moins utilisables pour des applications génériques de tatouage pour la transmission de données. Ces recherches ont donné lieu à quatre publications, deux lors de conférences internationales [PGBP10, PGB11a], et deux lors de conférences nationales [PGB10b, PGB11b].

Dans la dernière partie, nous avons présenté nos travaux relatifs aux représentations parcimonieuses, qui ont principalement consisté en le développement d'un système destiné à rendre parcimonieux la représentation temps-fréquence d'un signal de musique. Ces travaux sont directement reliés à ceux présentés dans les deux premières parties, car ils reposent en effet sur les mêmes bases : la modification de coefficients temps-fréquence guidée par un modèle psychoacoustique. Ces travaux, inspirés par ceux présentés dans [BLED10], sont aussi basés sur les outils du codage audio perceptuel, à l'instar de nos systèmes d'insertion, car les nombreuses recherches dans ce domaine ont donné lieu au développement d'outils extrêmement performants. Dans notre système de « parcimonisation », le seuil de masquage est translaté pour toutes les trames d'un montant fixe qui a été déterminé par des tests d'écoute subjectifs. Contrairement aux systèmes d'insertion, pour lesquels les coefficients sont plus ou moins quantifiés suivant la valeur du seuil de masquage, les coefficients dont la valeur est inférieure au seuil sont ici mis à zéro, alors que ceux dont la valeur est supérieure ne sont pas modifiés. Les résultats montrent que plus de 65% des coefficients temps-fréquence peuvent être mis à zéro sans impact sur la qualité audio, pour une modification d'énergie mineure du signal (environ 1%, ce qui indique que les coefficients supprimés étaient bien de valeur très faible). Cette thèse s'étant déroulée dans le cadre du projet DReaM, nous avons de plus décrit les travaux qui montrent en quoi cette « parcimonisation » est utile pour le projet DReaM, en particulier en temps que pré-traitement des sources individuelles afin d'améliorer le processus de séparation de sources informée proposé dans la méthode dite « par inversion locale ». Chaque source est rendue parcimonieuse avant le mixage, ce qui permet d'avoir un signal mixé lui aussi parcimonieux (bien qu'il le soit moins que les sources considérées séparément). Ceci se traduit par un gain en temps de calcul direct de plus de 30% lors de la séparation, puisqu'en moyenne 30% des coefficients du signal mixé sont nuls (indiquant que toutes les sources à ces points temps-fréquence sont elles aussi nulles). Bien que le gain en qualité reste probablement négligeable, ce gain en temps de calcul peut être

tout à fait intéressant, surtout dans le cadre éventuel de l'utilisation d'un logiciel DReaM de séparation de sources informée sur des plateformes disposant de ressources limitées (temps de calcul ou énergie), comme un téléphone portable ou une tablette graphique. Ces travaux ont eux aussi fait l'objet d'une publication, lors d'une conférence internationale [PG11b].

Le cadre du projet DReaM a de plus permis de nombreuses collaborations scientifiques entre les différents partenaires. Une collaboration a notamment eu lieu sur l'utilisation d'un des systèmes d'insertion pour une autre méthode de séparation de sources informée, elle aussi développée dans le cadre du projet DReaM en collaboration entre Télécom ParisTech et GIPSA-Lab [LPB+12]. Cette méthode repose sur un filtrage de Wiener. Elle est plus puissante mais plus coûteuse en calculs que la méthode par inversion locale, dont elle peut être vue comme une généralisation. Les filtres de Wiener sont construits à partir des paramètres de mélange (peu coûteux à transmettre au décodeur par tatouage) et à partir de spectrogrammes de puissance des signaux sources. Comme ces spectrogrammes sont assez coûteux à transmettre, ils sont modélisés soit par des techniques de factorisation en matrices non-négatives (NMF pour *Non-negative Matrix Factorization* en anglais), soit encodés comme des images par des techniques de codage avec pertes (de type JPEG par exemple). Le tatouage est alors utilisé pour transmettre soit les paramètres NMF, soit les données de quantification JPEG. La prise en compte du type de données à transmettre a nécessité une adaptation du système de tatouage pour optimiser le temps d'encodage et de décodage. Cette collaboration a ainsi permis de réaliser le développement d'un système complet de SSI opérationnel et performant, basé sur le filtrage de Wiener et le tatouage. D'autres collaborations ont eu lieu sur des sujets plus éloignés des thématiques principales de cette thèse, mais toujours en relation avec le projet DReaM, et ont résulté en deux publications lors de conférences internationales [SLP+12, MBB+12].

Au niveau des perspectives à court terme, des études sont envisagées pour adapter le système d'insertion aux signaux compressés, ceux-ci étant de plus en plus utilisés au détriment du format non compressé PCM 16 bits qui est considéré dans ce manuscrit. Dans le cadre du projet DReaM, l'utilisation de signaux compressés serait en effet un grand atout pour la diffusion de la séparation de sources informée auprès des utilisateurs. Il est bien connu et immédiatement apparent que les systèmes de codage audio perceptuels et de tatouage haut débit entrent en conflit : le premier cherche à supprimer les composantes audio inaudibles alors que le second cherche à rajouter de l'information de manière inaudible. Une solution directe à ce problème consisterait à simplement insérer les informations de séparation dans les zones de méta-données des flux compressés. Cependant dans le cadre du projet DReaM il est intéressant d'étudier s'il serait plus pertinent d'insérer de l'information par tatouage plutôt que de simplement l'insérer dans les méta-données des flux binaires compressés. L'idée est alors de chercher s'il est possible, à qualité audio équivalente et taille de flux binaire total équivalente, de transmettre plus d'information grâce au tatouage que si l'on utilisait les méta-données. En effet, on peut imaginer qu'une quantification des coefficients MDCT peut être contrainte à la fois (c'est-à-dire conjointement) en termes de compression et de tatouage d'un message donné, avec un résultat de débit total inférieur à celui obtenu par une quantification uniquement contrainte par la compression auquel on ajoute le débit

du message (ceci est notamment envisageable dans la compression MPEG-AAC du fait de la non-linéarité du codage entropique qui suit l'allocation de bits et la quantification, et du fait de la liberté permise par la norme sur la gestion de la boucle d'optimisation de ces processus). Un autre intérêt serait de permettre une relative « protection » de l'information additionnelle (plus difficile à supprimer dans les données compressées elles-mêmes plutôt que par un simple parsing du flux binaire entre données et méta-données).

Au niveau des perspectives à long terme, toujours dans le cadre du projet DReaM, il reste à étudier de façon plus approfondie les interactions entre la séparation de sources informée et le tatouage. En effet, le but de la SSI par rapport à la séparation de sources aveugle est d'améliorer la qualité en transmettant de l'information additionnelle. Cependant, si cette information est transmise par tatouage, le signal de mélange est perturbé par l'insertion ce qui peut dégrader les performances de séparation. Pour le système de SSI basé sur l'inversion locale présenté dans [PG11a], dans le cadre d'un mélange à 5 sources, un débit d'environ 64 kbits/s/c est nécessaire. Il est montré que ce débit d'insertion faible ne perturbe quasiment pas la séparation des sources. Cependant une insertion à 250 kbits/s/c (réalisée avec un tatouage aléatoire pour simuler l'insertion d'un plus grand nombre de données) provoque une chute sévère du SDR (*Signal-to-Distortion Ratio*) de 5dB (qui se traduit probablement par des défauts extrêmement audibles). Pour le système de SSI basé sur un filtrage de Wiener présenté dans [LPB+12], la dégradation semble se produire de manière plus lente. On voit par exemple qu'avec une insertion à 200 kbits/s/c, les performances ne subissent presque pas de dégradation, et commencent à se dégrader de plus en plus après ce débit. Ceci s'explique probablement par le fait que le filtrage de Wiener fonctionne de manière moins brutale que l'inversion locale. Il faut donc étudier en détail le compromis entre la quantité d'information à transmettre (plus il y en a, plus la séparation est informée et donc performante), et le débit d'insertion (plus il est élevé, plus le mix est dégradé et donc plus la séparation est dégradée). Finalement, nous pouvons même imaginer étudier de manière conjointe les interactions entre le tatouage, l'information de séparation et la compression. Ces trois paramètres ont en effet un impact sur la qualité de la séparation et sont tous trois quelque peu antagonistes. L'optimisation de ces trois éléments paraît très complexe mais, si elle est réalisable, elle semble être la manière optimale pour améliorer les performances de la SSI.

Annexes

Annexe A

MDCT et IntMDCT

A.1 Notations préliminaires

A.1.1 DCT Type-IV

La définition de la DCT-IV d'un vecteur $\mathbf{u}$ de longueur $N/2$ est la suivante :

$$\forall k \in \left[0, \frac{N}{2} - 1\right], \ \mathrm{DCT}\{\mathbf{u}\}(k) = \frac{2}{\sqrt{N}} \sum_{k=0}^{N/2-1} u(n) \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2}\right)\left(k + \frac{1}{2}\right)\right) \tag{A.1}$$

ou encore sous forme matricielle :

$$\mathrm{DCT}\{\mathbf{u}\} = \mathbf{D}\mathbf{u} \tag{A.2}$$

avec $\mathbf{D}$ la matrice de terme général :

$$d_{k,n} = \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2}\right)\left(k + \frac{1}{2}\right)\right) \tag{A.3}$$

La particularité de la matrice de DCT-IV $\mathbf{D}$ est qu'elle est symétrique (comme l'indique directement sa définition), et orthogonale (c'est-à-dire que son inverse est sa transposée). En combinant ses deux propriétés on a :

$$\mathbf{D}^2 = \mathbf{I} \tag{A.4}$$

où $\mathbf{I}$ désigne la matrice identité (on omettra toujours la dimension des matrices identité, celle-ci se déduisant du contexte) de $\mathscr{M}_{N/2}$.

A.1.2 Matrice « d'inversion »

On note $\mathbf{R}$ la matrice carrée qui permet d'inverser les lignes (respectivement les colonnes) en multipliant à gauche (respectivement à droite) d'une matrice :

$$\mathbf{R} = \begin{pmatrix} 0 & & 1 \\ & \diagup & \\ 1 & & 0 \end{pmatrix} \tag{A.5}$$

Comme pour la matrice identité on omettra la dimension de ces matrices.

A.1.3 Matrices de fenêtrage

On note aussi **Ha** et **Hb** les matrices diagonales de taille N dont les éléments diagonaux sont les échantillons des fenêtres d'analyse et de synthèse :

$$\mathbf{Ha} = \begin{pmatrix} h_a(0) & & 0 \\ & \ddots & \\ 0 & & h_a(N-1) \end{pmatrix} \qquad \mathbf{Hs} = \begin{pmatrix} h_s(0) & & 0 \\ & \ddots & \\ 0 & & h_s(N-1) \end{pmatrix} \tag{A.6}$$

On utilise aussi une écriture par blocs de taille $N/4$ de ces deux matrices :

$$\mathbf{Ha} = \begin{pmatrix} \mathbf{Ha_1} & 0 & 0 & 0 \\ 0 & \mathbf{Ha_2} & 0 & 0 \\ 0 & 0 & \mathbf{Ha_3} & 0 \\ 0 & 0 & 0 & \mathbf{Ha_4} \end{pmatrix} \qquad \mathbf{Hs} = \begin{pmatrix} \mathbf{Hs_1} & 0 & 0 & 0 \\ 0 & \mathbf{Hs_2} & 0 & 0 \\ 0 & 0 & \mathbf{Hs_3} & 0 \\ 0 & 0 & 0 & \mathbf{Hs_4} \end{pmatrix} \tag{A.7}$$

A.2 Décomposition des matrices de MDCT et d'IMDCT

A.2.1 MDCT

Pour simplifier les équations nous démarrons les indices des matrices à 0 (et non à 1 comme il est fait habituellement). On peut décomposer tout d'abord la matrice de MDCT de longueur N $\mathbf{M}_{\text{MDCT}}$ en séparant l'opération de fenêtrage :

$$\mathbf{M}_{\text{MDCT}} = \mathbf{MHa} \tag{A.8}$$

où les coefficients de la matrice **M** sont définis par :

$$\forall (k,n) \in \left[0, \frac{N}{2} - 1\right] \times [0, N-1],\ m_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2} + \frac{N}{4}\right)\left(k + \frac{1}{2}\right)\right) \tag{A.9}$$

On décompose la matrice **M** en blocs de taille $N/2 \times N/4$:

$$\mathbf{M} = \begin{pmatrix} \mathbf{M1} & \mathbf{M2} & \mathbf{M3} & \mathbf{M4} \end{pmatrix} \tag{A.10}$$

Les coefficients de ces sous-matrices sont donc définis par :

$$\forall (k,n) \in \left[0, \frac{N}{2} - 1\right] \times \left[0, \frac{N}{4} - 1\right] \begin{cases} m1_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2} + \frac{N}{4}\right)\left(k + \frac{1}{2}\right)\right) \\ m2_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2} + \frac{N}{2}\right)\left(k + \frac{1}{2}\right)\right) \\ m3_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2} + \frac{3N}{4}\right)\left(k + \frac{1}{2}\right)\right) \\ m4_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n + \frac{1}{2} + N\right)\left(k + \frac{1}{2}\right)\right) \end{cases} \tag{A.11}$$

On rappelle que les coefficients de la matrice de DCT-IV de taille $N/2$ **D** sont définis par :

$$\forall (k,n) \in \left[0, \frac{N}{2}-1\right]^2,\ d_{k,n} = \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right) \tag{A.12}$$

On décompose alors aussi cette matrice **D** en sous-matrices de taille $N/2 \times N/4$:

$$\mathbf{D} = \begin{pmatrix} \mathbf{D1} & \mathbf{D2} \end{pmatrix} \tag{A.13}$$

Les coefficients de ces deux sous-matrices sont donc immédiatement définis par :

$$\forall (k,n) \in \left[1, \frac{N}{2}\right] \times \left[1, \frac{N}{4}\right] \begin{cases} d1_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right) \\ d2_{k,n} = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) \end{cases} \tag{A.14}$$

Nous allons maintenant exprimer les coefficients des matrices **M1**, **M2**, **M3** et **M4** en fonction des coefficients des matrices **D1** et **D2**. On a immédiatement :

$$m1_{k,n} = d2_{k,n} \tag{A.15}$$

Pour les autres coefficients, on utilise les règles de trigonométrie de base :

$$\begin{aligned}
m2_{k,n} &= \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{N}{2}\right)\left(k+\frac{1}{2}\right)\right) && \text{(A.16)} \\
&= \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(-n-\frac{1}{2}-\frac{N}{2}\right)\left(k+\frac{1}{2}\right)\right) && \text{(A.17)} \\
&= -\frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(-n-\frac{1}{2}-\frac{N}{2}\right)\left(k+\frac{1}{2}\right)+\frac{2\pi}{N}N\left(k+\frac{1}{2}\right)\right) && \text{(A.18)} \\
&= -\frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(-n-\frac{1}{2}+\frac{N}{2}\right)\left(k+\frac{1}{2}\right)\right) && \text{(A.19)} \\
&= -\frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\left(\frac{N}{4}-1-n+\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) && \text{(A.20)} \\
&= -d2_{k,N/4-1-n} && \text{(A.21)}
\end{aligned}$$

$$m3_{k,n} = \frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{3N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{A.22}$$
$$= \frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(-n-\frac{1}{2}-\frac{3N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{A.23}$$
$$= -\frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(-n-\frac{1}{2}-\frac{3N}{4}\right)\left(k+\frac{1}{2}\right)+\frac{2\pi}{N}N\left(k+\frac{1}{2}\right)\right) \tag{A.24}$$
$$= -\frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(-n-\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{A.25}$$
$$= -\frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(\frac{N}{4}-1-n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right) \tag{A.26}$$
$$= -d1_{k,N/4-1-n} \tag{A.27}$$

$$m4_{k,n} = \frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+N\right)\left(k+\frac{1}{2}\right)\right) \tag{A.28}$$
$$= -\frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+N\right)\left(k+\frac{1}{2}\right)-\frac{2\pi}{N}N\left(k+\frac{1}{2}\right)\right) \tag{A.29}$$
$$= -\frac{2}{\sqrt{N}}\cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right) \tag{A.30}$$
$$= -d1_{k,n} \tag{A.31}$$

On en déduit donc que :

$$\mathbf{M1} = \mathbf{D2} \tag{A.32}$$
$$\mathbf{M2} = -\mathbf{D2R} \tag{A.33}$$
$$\mathbf{M3} = -\mathbf{D1R} \tag{A.34}$$
$$\mathbf{M4} = -\mathbf{D1} \tag{A.35}$$

On peut donc factoriser la matrice $\mathbf{M}$:

$$\mathbf{M} = \begin{pmatrix}\mathbf{M1} & \mathbf{M2} & \mathbf{M3} & \mathbf{M4}\end{pmatrix} \tag{A.36}$$
$$= \begin{pmatrix}\mathbf{D2} & -\mathbf{D2R} & -\mathbf{D1R} & -\mathbf{D1}\end{pmatrix} \tag{A.37}$$
$$= \begin{pmatrix}\mathbf{D1} & \mathbf{D2}\end{pmatrix}\begin{pmatrix}\mathbf{0} & \mathbf{0} & -\mathbf{R} & -\mathbf{I} \\ \mathbf{I} & -\mathbf{R} & \mathbf{0} & \mathbf{0}\end{pmatrix} \tag{A.38}$$
$$= \mathbf{DF} \tag{A.39}$$

On conclut alors en ajoutant l'opération de fenêtrage :

$$\mathbf{M}_{\mathrm{MDCT}} = \mathbf{DFHa} \tag{A.40}$$

A.2.2 IMDCT

La matrice de l'IMDCT est très similaire à celle de la MDCT. En utilisant les notations précédentes et la définition de l'IMDCT donnée section 3.3.3, on a :

$$\mathbf{M}_{\text{IMDCT}} = \mathbf{HsM}^T \tag{A.41}$$
$$= \mathbf{HsF}^T\mathbf{D} \tag{A.42}$$

puisque rappelons-le, la matrice de DCT-IV $\mathbf{D}$ est symétrique.

A.3 Conditions de Princen-Bradley : version matricielle

Les conditions de Princen-Bradley ont été énoncées dans la section 3.3.3 à l'équation (3.14). Si nous les réécrivons en version matricielle, nous obtenons :

$$\begin{cases} \begin{pmatrix} \mathbf{Hs}_1 & 0 \\ 0 & \mathbf{Hs}_2 \end{pmatrix}\begin{pmatrix} \mathbf{Ha}_1 & 0 \\ 0 & \mathbf{Ha}_2 \end{pmatrix} + \begin{pmatrix} \mathbf{Hs}_3 & 0 \\ 0 & \mathbf{Hs}_4 \end{pmatrix}\begin{pmatrix} \mathbf{Ha}_3 & 0 \\ 0 & \mathbf{Ha}_4 \end{pmatrix} = \mathbf{I} \\ \begin{pmatrix} \mathbf{Hs}_1 & 0 \\ 0 & \mathbf{Hs}_2 \end{pmatrix}\mathbf{R}\begin{pmatrix} \mathbf{Ha}_1 & 0 \\ 0 & \mathbf{Ha}_2 \end{pmatrix} - \begin{pmatrix} \mathbf{Hs}_3 & 0 \\ 0 & \mathbf{Hs}_4 \end{pmatrix}\mathbf{R}\begin{pmatrix} \mathbf{Ha}_3 & 0 \\ 0 & \mathbf{Ha}_4 \end{pmatrix} = 0 \end{cases} \tag{A.43}$$

$$\Leftrightarrow \begin{cases} \begin{pmatrix} \mathbf{Hs}_1\mathbf{Ha}_1 & 0 \\ 0 & \mathbf{Hs}_2\mathbf{Ha}_2 \end{pmatrix} + \begin{pmatrix} \mathbf{Hs}_3\mathbf{Ha}_3 & 0 \\ 0 & \mathbf{Hs}_4\mathbf{Ha}_4 \end{pmatrix} = \mathbf{I} \\ \begin{pmatrix} \mathbf{Hs}_1 & 0 \\ 0 & \mathbf{Hs}_2 \end{pmatrix}\begin{pmatrix} 0 & \mathbf{RHa}_2 \\ \mathbf{RHa}_1 & 0 \end{pmatrix} - \begin{pmatrix} \mathbf{Hs}_3 & 0 \\ 0 & \mathbf{Hs}_4 \end{pmatrix}\begin{pmatrix} 0 & \mathbf{RHa}_4 \\ \mathbf{RHa}_3 & 0 \end{pmatrix} = 0 \end{cases} \tag{A.44}$$

$$\Leftrightarrow \begin{cases} \begin{pmatrix} \mathbf{Hs}_1\mathbf{Ha}_1 + \mathbf{Hs}_3\mathbf{Ha}_3 & 0 \\ 0 & \mathbf{Hs}_2\mathbf{Ha}_2 + \mathbf{Hs}_4\mathbf{Ha}_4 \end{pmatrix} = \mathbf{I} \\ \begin{pmatrix} 0 & \mathbf{Hs}_1\mathbf{RHa}_2 \\ \mathbf{Hs}_2\mathbf{RHa}_1 & 0 \end{pmatrix} - \begin{pmatrix} 0 & \mathbf{Hs}_3\mathbf{RHa}_4 \\ \mathbf{Hs}_4\mathbf{RHa}_3 & 0 \end{pmatrix} = 0 \end{cases} \tag{A.45}$$

$$\Leftrightarrow \begin{cases} \mathbf{Hs}_1\mathbf{Ha}_1 + \mathbf{Hs}_3\mathbf{Ha}_3 = \mathbf{I} \\ \mathbf{Hs}_2\mathbf{Ha}_2 + \mathbf{Hs}_4\mathbf{Ha}_4 = \mathbf{I} \\ \mathbf{Hs}_1\mathbf{RHa}_2 - \mathbf{Hs}_3\mathbf{RHa}_4 = 0 \\ \mathbf{Hs}_2\mathbf{RHa}_1 - \mathbf{Hs}_4\mathbf{RHs}_3 = 0 \end{cases} \tag{A.46}$$

A.4 Reconstruction parfaite de la MDCT

Nous allons montrer ici que la MDCT est bien à reconstruction parfaite si les conditions (3.14) sont vérifiées, ce qui n'est pas forcément évident au premier abord. En effet, dans le cas de la TFD avec recouvrement, on conçoit facilement que la reconstruction est parfaite puisque la TFD est inversible. La seule modification d'une trame vient du fenêtrage, et cet effet est facilement inversible, même si l'on s'arrange parfois pour choisir des fenêtres telles qu'aucune correction ne soit nécessaire. On peut voir immédiatement que ce n'est pas le cas

pour la MDCT car pour une trame de N points temporels il y a seulement $N/2$ coefficients MDCT, et la trame n'est donc pas inversible seule, même sans fenêtrage, contrairement à la TFD. La reconstruction parfaite vient donc forcément du recouvrement et du fenêtrage, comme nous allons le détailler. Il existe de nombreuses méthodes pour montrer la propriété de reconstruction parfaite de la MDCT (qui est montrée entre autres dans l'article fondateur [PB86]), nous en avons choisi une en particulier car elle paraît être la plus simple. Nous allons d'abord montrer que la MDCT et l'IMDCT peuvent s'exprimer en fonction de la DCT de Type IV, puisque la reconstruction est parfaite grâce au recouvrement avec les conditions de Princen-Bradley décrites précédemment.

Avec les notations présentées en annexe A.1 et si l'on note $\mathbf{M}_{\text{MDCT}}$ la matrice de la MDCT (incluant le fenêtrage) de taille N, on a (voir annexe A.2.1) :

$$\mathbf{M}_{\text{MDCT}} = \mathbf{DFHa} \tag{A.47}$$

où $\mathbf{F}$ est une matrice définie par blocs de taille $N/4 \times N/4$ de la façon suivante :

$$\mathbf{F} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{R} & -\mathbf{I} \\ \mathbf{I} & -\mathbf{R} & \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{A.48}$$

Calculer les coefficients MDCT d'un vecteur $\mathbf{x}$ de longueur N revient donc à calculer la DCT-IV du vecteur $\mathbf{FHax}$ de longueur $N/2$, sorte de version fenêtrée et repliée de $\mathbf{x}$.

La matrice de IMDCT $\mathbf{M}_{\text{IMDCT}}$ peut être décomposée de manière similaire (voir annexe A.2.2) :

$$\mathbf{M}_{\text{IMDCT}} = \mathbf{HsF}^T\mathbf{D} \tag{A.49}$$

Dans ce cas, une DCT-IV est d'abord calculée, puis elle est étendue de manière redondante et ensuite fenêtrée.

Maintenant que nous avons exprimé les matrices de MDCT et d'IMDCT, nous pouvons exprimer l'enchaînement de ces deux fonctions en calculant le produit des matrices associées :

$$\mathbf{M}_{\text{IMDCT}}\mathbf{M}_{\text{MDCT}} = \mathbf{HsF}^T\mathbf{DDFHa} \tag{A.50}$$

$$= \mathbf{HsF}^T\mathbf{FHa} \tag{A.51}$$

$$= \mathbf{Hs}\begin{pmatrix} \mathbf{I} & -\mathbf{R} & \mathbf{0} & \mathbf{0} \\ -\mathbf{R} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{R} \\ \mathbf{0} & \mathbf{0} & \mathbf{R} & \mathbf{I} \end{pmatrix}\mathbf{Ha} \tag{A.52}$$

$$= \begin{pmatrix} \mathbf{Hs}_1\mathbf{Ha}_1 & -\mathbf{Hs}_1\mathbf{RHa}_2 & \mathbf{0} & \mathbf{0} \\ -\mathbf{Hs}_2\mathbf{RHa}_1 & \mathbf{Hs}_2\mathbf{Ha}_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Hs}_3\mathbf{Ha}_3 & \mathbf{Hs}_3\mathbf{RHa}_4 \\ \mathbf{0} & \mathbf{0} & \mathbf{Hs}_4\mathbf{RHa}_3 & \mathbf{Hs}_4\mathbf{Ha}_4 \end{pmatrix} \tag{A.53}$$

$$= \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{pmatrix} \tag{A.54}$$

Puisque le résultat ne vaut pas l'identité il va falloir prendre en compte le recouvrement pour avoir la reconstruction parfaite.

On considère alors deux trames temporelles de longueur N consécutives (et donc avec un recouvrement de 50%) $\mathbf{x}_1$ et $\mathbf{x}_2$ que l'on note par sous-vecteurs de longueur $N/2$:

$$\mathbf{x}_1 = \begin{pmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{pmatrix} \qquad \mathbf{x}_2 = \begin{pmatrix} \mathbf{s}_2 \\ \mathbf{s}_3 \end{pmatrix} \tag{A.55}$$

L'enchaînement des opérations de MDCT et d'IMDCT sur ces deux trames donne donc :

$$\mathbf{M}_{\text{IMDCT}}\mathbf{M}_{\text{MDCT}}\mathbf{x}_1 = \begin{pmatrix} \mathbf{As}_1 \\ \mathbf{Bs}_2 \end{pmatrix} \qquad \mathbf{M}_{\text{IMDCT}}\mathbf{M}_{\text{MDCT}}\mathbf{x}_2 = \begin{pmatrix} \mathbf{As}_2 \\ \mathbf{Bs}_3 \end{pmatrix} \tag{A.56}$$

Et le repliement est donc représenté par :

$$\mathbf{Bs}_2 + \mathbf{As}_2 = (\mathbf{A} + \mathbf{B})\,\mathbf{s}_2 \tag{A.57}$$

Il faut donc vérifier que $\mathbf{A} + \mathbf{B} = \mathbf{I}$, c'est-à-dire que :

$$\begin{pmatrix} \mathbf{Hs}_1\mathbf{Ha}_1 & -\mathbf{Hs}_1\mathbf{RHa}_2 \\ -\mathbf{Hs}_2\mathbf{RHa}_1 & \mathbf{Hs}_2\mathbf{Ha}_2 \end{pmatrix} + \begin{pmatrix} \mathbf{Hs}_3\mathbf{Ha}_3 & \mathbf{Hs3RHa}_4 \\ \mathbf{Hs}_4\mathbf{RHa}_3 & \mathbf{Hs}_4\mathbf{Ha}_4 \end{pmatrix} = \mathbf{I} \tag{A.58}$$

et donc que :

$$\begin{cases} \mathbf{Hs}_1\mathbf{Ha}_1 + \mathbf{Hs}_3\mathbf{Ha}_3 = I \\ \mathbf{Hs}_2\mathbf{Ha}_2 + \mathbf{Hs}_4\mathbf{Ha}_4 = I \\ \mathbf{Hs}_1\mathbf{RHa}_2 - \mathbf{Hs}_3\mathbf{RHa}_4 = 0 \\ \mathbf{Hs}_2\mathbf{RHa}_1 - \mathbf{Hs}_4\mathbf{RHa}_3 = 0 \end{cases} \tag{A.59}$$

Or ces quatre équations sont exactement la réécriture matricielle des conditions de Princen-Bradley (3.14) (voir annexe A.3). Nous avons donc bien montré la reconstruction parfaite à condition que les conditions de Princen-Bradley soient respectées.

A.5 Orthogonalité de la matrice O

Nous allons montrer ici que la matrice $\mathbf{O}$ (section 7.2) est bien orthogonale dans le cas où les fenêtres d'analyse et synthèse sont identiques : $\mathbf{Ha} = \mathbf{Hs} = \mathbf{H}$ et symétriques. Nous rappelons la définition de la matrice :

$$\mathbf{O} = \begin{pmatrix} -\mathbf{RH}_3 & -\mathbf{H}_4 \\ \mathbf{H}_1 & -\mathbf{RH}_2 \end{pmatrix} \tag{A.60}$$

Nous avons donc :

$$\mathbf{O}^T\mathbf{O} = \begin{pmatrix} -\mathbf{H}_3\mathbf{R} & \mathbf{H}_1 \\ \mathbf{H}_4 & -\mathbf{H}_2\mathbf{R} \end{pmatrix} \begin{pmatrix} -\mathbf{R}\mathbf{H}_3 & -\mathbf{H}_4 \\ \mathbf{H}_1 & -\mathbf{R}\mathbf{H}_2 \end{pmatrix} \tag{A.61}$$

$$= \begin{pmatrix} \mathbf{H}_3\mathbf{R}^2\mathbf{H}_3 + \mathbf{H}_1{}^2 & \mathbf{H}_3\mathbf{R}\mathbf{H}_4 - \mathbf{H}_1\mathbf{R}\mathbf{H}_2 \\ \mathbf{H}_4\mathbf{R}\mathbf{H}_3 - \mathbf{H}_2\mathbf{R}\mathbf{H}_1 & \mathbf{H}_4{}^2 + \mathbf{H}_2\mathbf{R}^2\mathbf{H}_2 \end{pmatrix} \tag{A.62}$$

$$= \begin{pmatrix} \mathbf{H}_3{}^2 + \mathbf{H}_1{}^2 & \mathbf{H}_3\mathbf{R}\mathbf{H}_4 - \mathbf{H}_1\mathbf{R}\mathbf{H}_2 \\ \mathbf{H}_4\mathbf{R}\mathbf{H}_3 - \mathbf{H}_2\mathbf{R}\mathbf{H}_1 & \mathbf{H}_4{}^2 + \mathbf{H}_2{}^2 \end{pmatrix} \tag{A.63}$$

$$= \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{A.64}$$

en utilisant la version matricielle des conditions de Princen-Bradley (annexe A.3), ce qui conclut la démonstration.

Annexe B

Calculs divers

B.1 Variance du bruit de quantification sur les coefficients MDCT du au PCM 16 bits

On a :

$$\mathbf{B}(k) = \frac{2}{\sqrt{N}} \sum_{n=0}^{N-1} \mathbf{b}(n)\mathbf{h}(n) \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{B.1}$$

On pose alors :

$$\mathbf{b}'(n) = \frac{2}{\sqrt{N}} \mathbf{b}(n)\mathbf{h}(n) \cos\left(\frac{2\pi}{N}\left(n+\frac{1}{2}+\frac{N}{4}\right)\left(k+\frac{1}{2}\right)\right) \tag{B.2}$$

$$= \frac{2}{\sqrt{N}} \mathbf{b}(n)\mathbf{h}(n) c(n,k) \tag{B.3}$$

On a donc :

$$\mathbf{B}(k) = \sum_{n=0}^{N-1} \mathbf{b}'(n) \tag{B.4}$$

En utilisant une version du théorème central limite avec hypothèse faible (conditions de Lyapunov ou Lindeberg, voir [Fel71] ou MathWorld^{TM}en cliquant ici), on peut alors montrer que les coefficients MDCT $\mathbf{B}(k)$ suivent une loi normale de variance $\sigma^2_{\mathbf{B}(k)}$:

$$\mathbf{B}(k) \sim \mathcal{N}\left(0, \sigma^2_{\mathbf{B}(k)}\right) \tag{B.5}$$

avec :

$$\sigma^2_{\mathbf{B}(k)} = \sum_{n=0}^{N-1} \sigma^2_{\mathbf{b}'(n)} \tag{B.6}$$
$$= \sum_{n=0}^{N-1} \frac{4}{N} \frac{\mathbf{h}^2(n)c^2(n,k)}{12} \tag{B.7}$$
$$= \frac{1}{3N} \sum_{n=0}^{N-1} \mathbf{h}^2(n)c^2(n,k) \tag{B.8}$$
$$= \frac{1}{3N} \left[\sum_{n=0}^{N/2-1} \mathbf{h}^2(n)c^2(n,k) + \sum_{n=N/2}^{N-1} \mathbf{h}^2(n)c^2(n,k) \right] \tag{B.9}$$
$$= \frac{1}{3N} \left[\sum_{n=0}^{N/2-1} \mathbf{h}^2(n)c^2(n,k) + \sum_{n=0}^{N/2-1} \mathbf{h}^2(N-1-n)c^2(N-1-n,k) \right] \tag{B.10}$$
$$= \frac{1}{3N} \left[\sum_{n=0}^{N/2-1} \mathbf{h}^2(n)c^2(n,k) + \sum_{n=0}^{N/2-1} \mathbf{h}^2(n)c^2(N-1-n,k) \right] \tag{B.11}$$
$$= \frac{1}{3N} \sum_{n=0}^{N/2-1} \mathbf{h}^2(n) \left[c^2(n,k) + c^2(N-1-n,k) \right] \tag{B.12}$$
$$= \frac{1}{3N} \sum_{n=0}^{N/2-1} \mathbf{h}^2(n) \tag{B.13}$$
$$= \frac{1}{3N} \frac{N}{4} \tag{B.14}$$
$$= \frac{1}{12} \tag{B.15}$$
$$= \sigma^2_{\mathrm{PCM}} \tag{B.16}$$

Cette variance ne dépend donc pas de l'indice k du coefficient MDCT considéré. Le bruit uniforme sur les coefficients temporels de variance σ^2_{PCM} se traduit donc par un bruit gaussien de même variance sur les coefficients MDCT, indépendamment de l'indice du coefficient considéré.

Ce qu'il faut bien noter ici c'est que l'approximation n'est faite que sur la nature du bruit (gaussien), qui vient du fait que l'on suppose que N est assez grand (théorème central limite). Le calcul de la variance lui n'utilise que le fait que les variables sont supposées indépendantes, hypothèse qui semble assez probable. Le fait que nous utilisions ici une hypothèse du théorème central limite implique en fait une convergence plus lente que dans le cas courant. En effet dans le cas usuel, c'est-à-dire où les variables sont indépendantes et identiquement distribuées, le théorème de Berry-Esséen donne une vitesse de convergence en $\frac{1}{\sqrt{N}}$ de la fonction de répartition de la moyenne des variables (voir [Fel71] ou MathWorld$^{\mathrm{TM}}$en cliquant ici). Or dans notre cas avec hypothèses faibles ce théorème n'est plus strictement valable et la convergence est plus lente, ce qui implique que la nature du bruit n'est pas forcément

exactement identique pour tous les canaux fréquentiels, et peut varier significativement dans le cas de valeurs de N faibles.

B.2 Calcul de la probabilité d'erreur cible

Afin de calculer la probabilité d'erreur symbole cible pour un coefficient MDCT donné, on fait l'approximation suivante :

$$p_{es} = \int_{\Delta_Q/2}^{+\infty} \frac{1}{\sigma_{\text{PCM}}\sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma_{\text{PCM}}^2}\right) \mathrm{d}t + \int_{-\infty}^{-\Delta_Q/2} \frac{1}{\sigma_{\text{PCM}}\sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma_{\text{PCM}}^2}\right) \mathrm{d}t \quad \text{(B.17)}$$

$$= 2\int_{\Delta_Q/2}^{+\infty} \frac{1}{\sigma_{\text{PCM}}\sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma_{\text{PCM}}^2}\right) \mathrm{d}t \quad \text{(B.18)}$$

$$= \frac{2}{\sigma_{\text{PCM}}\sqrt{2\pi}} \int_{\Delta_Q/2}^{+\infty} \exp\left(-\frac{t^2}{2\sigma_{\text{PCM}}^2}\right) \mathrm{d}t \quad \text{(B.19)}$$

$$= \frac{2}{\sqrt{\pi}} \int_{\Delta_Q/\left(2\sqrt{2}\sigma_{\text{PCM}}\right)}^{+\infty} \exp\left(-x^2\right) \mathrm{d}x \quad \text{(B.20)}$$

$$= 1 - \operatorname{erf}\left(\frac{\Delta_Q}{2\sqrt{2}\sigma_{\text{PCM}}}\right) \quad \text{(B.21)}$$

Où erf désigne la fonction d'erreur usuelle :

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp\left(-t^2\right) \mathrm{d}t \quad \text{(B.22)}$$

On a donc :

$$\Delta_Q = 2\sqrt{2}\sigma_{\text{PCM}} \operatorname{erf}^{-1}(1 - p_{es}) \quad \text{(B.23)}$$

Pour obtenir une approximation plus fine du taux d'erreur p_{es}, il faudrait prendre en compte le fait qu'une forte modification de la valeur d'un coefficient $\mathbf{Y}(k)$ peut ne pas engendrer d'erreur de décodage (il est en effet possible de retomber sur un autre niveau de quantification du quantificateur utilisé pour insérer l'information). Pour tenir compte de cet effet, il faudrait considérer non pas une loi normale mais une loi normale enroulée (*wrapped gaussian* en anglais), avec des paramètres qui dépendraient de la charge C. Cependant, comme on cherche à avoir un taux d'erreur très faible et un très haut débit (et donc généralement une charge élevée), l'approximation utilisant une loi normale paraît tout à fait acceptable. Ce taux d'erreur p_{es} est à comprendre comme une sorte de taux d'erreur symbole, chaque symbole étant l'information insérée dans un coefficient MDCT. En effet, une erreur résultant du décalage sur un niveau de quantification d'un quantificateur adjacent n'implique pas forcément une erreur de un bit. Il est à noter que la charge étant variable suivant le coefficient MDCT considéré, la taille des symboles est elle aussi variable.

Bibliographie

[AC98] Alan Agresti and B. A. Coull. Approximate is better than "exact" for interval estimation of binomial proportions. *The American Statistician*, 52(2) :119–126, May 1998.

[AP06] S.A. Abdallah and M.D. Plumbley. Unsupervised analysis of polyphonic music by sparse coding. *IEEE Transactions on Neural Networks*, 17(1) :179–196, january 2006.

[ATS10] ATSC. *Document A/52 :2010. Digital Audio Compression Standard (AC-3, E-AC-3).* ATSC, 2010.

[BAB+06] G. Bailly, V. Attina, C. Baras, P. Bas, S. Baudry, D. Beautemps, R. Brun, J.M. Chassery, F. Davoine, F. Elisei, G. Gibert, G. Girin, D. Grison, J.P. Léoni, J. Liénard, N. Moreau, and P. Nguyen. ARTUS : Synthesis and audiovisual watermarking of the movements of a virtual agent interpreting subtitling using cued speech for deaf televiewers. *Modeling, Measurement and Control - C 67SH, supplement : handicap*, 2 :177–187, 2006.

[Bas11] P. Bas. Soft-SCS : Improving the security and robustness of the scalar-costa-scheme by optimal distribution matching. In *13th International Conference on Information Hiding*, pages 208–222, May 2011.

[BCD01] L.D. Brown, T.T. Cai, and A.A. DasGupta. Interval estimation for a binomial proportion. *Statistical Science*, 16(2) :101–133, 2001.

[Ben48] W.R. Bennett. Spectra of quantized signals. *Bell System Technichal Journal*, 27(3) :446–472, 1948.

[BLED10] P. Balazs, B. Laback, G. Eckel, and W.A. Deutsch. Frequency sparsity by removing perceptually irrelevant components using a simple model of simultaneous masking. *IEEE Transactions on Acoustics and Speech, Signal Processing*, 18(1) :34 –49, january 2010.

[Bra87a] K. Brandenburg. Evaluation of quality for audio encoding at low bit rates. In *Audio Engineering Society Convention 82*, 3 1987.

[Bra87b] K. Brandenburg. OCF–a new coding algorithm for high quality sound signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 12, pages 141 – 144, april 1987.

[BS94] J. G. Beerends and J. A. Stemerdink. A perceptual speech-quality measure based on a psychoacoustic sound representation. *Journal of the Audio Engineering Society*, 42(3) :115–123, 1994.

[BVdE92] F.A.M.L. Bruekers and A.W.M. Van den Enden. New networks for perfect inversion and perfect reconstruction. *Selected Areas in Communications, IEEE Journal on*, 10(1) :129–137, 1992.

[CKLS97] I.J. Cox, J. Kilian, F.T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *Image Processing, IEEE Transactions on*, 6(12) :1673 –1687, dec 1997.

[CM03] F. Cayre and B. Macq. Data hiding on 3-d triangle meshes. *IEEE transactions on signal processing*, 51(4) :939–949, April 2003.

[CMB01] Ingemar J. Cox, Matthew L. Miller, and Jeffrey A. Bloom. *Digital Watermarking*. Morgan Kaufmann Publishers, 2001.

[CMM99] Ingemar J. Cox, Matthew L. Miller, and Andrew L. McKellips. Watermarking as communications with side information. *Proceedings of the IEEE*, 87(7) :1127 –1141, jul 1999.

[Col94] C. Colomes. *Étude d'un modèle d'audition et d'une mesure objective de la qualité d'un signal sonore dans le contexte de codage à réduction de débit*. PhD thesis, Université de Rennes 1, Septembre 1994.

[Cos83] Max H. M. Costa. Writing on dirty paper (corresp.). *IEEE Transactions on Information Theory*, 29(3) :439 – 441, may 1983.

[CS02a] N. Cvejic and T. Seppänen. Increasing the capacity of LSB-based audio steganography. In *IEEE Workshop on Multimedia Signal Processing*, pages 336–338, 2002.

[CS02b] N. Cvejic and T. Seppänen. A wavelet domain LSB insertion algorithm for high capacity audio steganography. In *IEEE Digital Signal Processing Workshop*, pages 53–55, 2002.

[CSTS05] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber. Lossless generalized-lsb data embedding. *IEEE Transactions on Image Processing*, 14(2) :253–266, February 2005.

[CT06] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience. John Wiley & Sons, 2nd edition, 2006.

[CW99] B. Chen and G.W. Wornell. An information-theoretic approach to the design of robust digital watermarking systems. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 2061–2064, march 1999.

[CW01] Brian Chen and Gregory W. Wornell. Quantization index modulation : a class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, 47(4) :1423 –1443, may 2001.

[Dau06] L. Daudet. Sparse and structured decompositions of signals with the molecular matching pursuit. *IEEE Transactions on Acoustics and Speech, Signal Processing*, 14(5) :1808 –1816, september 2006.

[DD08] C. Dehon and C. Droesbeke, J.J. et Vermandele. *Eléments de statistique*. Ellipses, 2008.

[DS96] I. Daubechies and W. Sweldens. Factoring wevelet transforms into lifting steps. Technical report, Bell Laboratories, Lucent Technologies, 1996.

[EBTG03] J.J. Eggers, R. Bauml, R. Tzschoppe, and B. Girod. Scalar costa scheme for information embedding. *IEEE Transactions on Signal Processing*, 51(4) :1003 – 1019, april 2003.

[Fel71] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 2. Wiley, 1971.

[FZ06] Hugo Fastl and Eberhard Zwicker. *Psychoacoustics : Facts and Models*, volume 22 of *Springer Series in Information Sciences*. Springer, 2006.

[GHKB02] R. Geiger, A. Herre, J. Koller, and K. Brandenburg. IntMDCT - a link between perceptual and lossless audio coding. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1813–1816, 2002.

[GM13] S. Gorlow and S. Marchand. Informed audio source separation using linearly constrained spatial filters. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(1) :3–13, 2013.

[GP80] S. I. Gel'fand and M. S. Pinsker. Coding for channel with random parameters. *Problems of control and information theory*, 9(1) :19 – 31, 1980.

[Gri02] R. Gribonval. Sparse decomposition of stereo signals with matching pursuit and application to blind separation of more than two sources from a stereo mixture. In *IEEE International Conference on Acoustics and Speech, Signal Processing*, volume 3, april 2002.

[GYS06] R. Geiger, Y. Yokotani, and G. Schuller. Audio data hiding with high data rates based on IntMDCT. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 5, may 2006.

[HK06] R. Huber and B. Kollmeier. PEMO-Q – a new method for objective audio quality assessment using a model of auditory perception. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(6) :1902–1911, 2006.

[Hou77] T. Houtgast. Auditory-filter characteristics derived from direct-masking data and pulsation-threshold data with a rippled-noise masker. *Journal of the Acoustical Society of America*, 62 :409–415, 1977.

[HR08] N. Hurley and S. Rickard. Comparing measures of sparsity. In *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing*, pages 55–60, october 2008.

[IK98] K. Immink and A.S. Kees. The compact disc story. *Journal of the audio engineering society*, 46(5), may 1998.

[IR97] ITU-R, editor. *Recommendation BS.1116-1. Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems*. ITU, 1997.

[IR01] ITU-R. *Recommandation ITU-R BS.1387-1. Method for objective measurements of perceived audio quality*. ITU, 2001.

[IR03] ITU-R. *Recommandation ITU-R BS.1534-1. Method for the subjective assessment of intermediate quality level of coding systems*. ITU, 2003.

[IS04] A. Iliev and M. Scordilis. Multi level high capacity data hiding technique for stereo audio. In *38th Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1793–1797, 2004.

[ISO98] ISO/IEC. *ISO/IEC 13818-3 :1998. Information technology – Generic coding of moving pictures and associated audio information – Part 3 : Audio*. ISO/IEC, 1998.

[ISO03] ISO. *ISO 226 :2003. Acoustics – Normal equal-loudness-level contours*. ISO, 2003.

[ISO09] ISO/IEC. *ISO/IEC 14496-3 :2009. Information technology – Coding of audio-visual objects – Part 3 : Audio*. ISO/IEC, 2009.

[Joh88] J.D. Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on Selected Areas in Communications*, 6(2) :314 –323, february 1988.

[Joh07] J.D. Johnston. Perceptual audio coding - a history and timeline. In *Conference Record of the 41st Asilomar Conference on Signals, Systems and Computers*, pages 2085–2087, 2007.

[KA03] J. Karvanen and Cichocki A. Measuring sparseness of noisy signals. In *International Symposium on Independent Component Analysis and Blind Source Separation*, pages 125–130, Nara, Japan, 2003.

[KMKM00] M. Kesal, M.K. Mihçak, R. Koetter, and P. Moulin. Iteratively decodable codes for watermarking applications. In *Proceedings of the 2nd Symposium on Turbo Codes and Their Applications*, Brest, France, 2000.

[KT88] N. Komatsu and H. Tominaga. Authentication system using concealed images in telematics. In *Memoirs of the school of science and engineering*. Waseda University, 1988.

[Lan02] S. Lang. *Algebra*. Graduate Texts in Mathematics. Springer-Verlag, 3rd edition, 2002.

[LBR11] A. Liutkus, R. Badeau, and G. Richard. Gaussian processes for underdetermined source separation. *IEEE Transactions on Signal Processing*, 59(7) :3155–3167, 2011.

[LGS+12] A. Liutkus, S. Gorlow, N. Sturmel, Shuhua Zhang, L. Girin, R. Badeau, L. Daudet, S. Marchand, and G. Richard. Informed audio source separation : A comparative study. In *Proceedings of the 20th European Signal Processing Conference*, pages 2397–2401, 2012.

[LJS05] S.D. Larbi and M. Jaidane-Saidane. Audio watermarking : a way to stationnarize audio signals. *IEEE Transactions on Signal Processing*, 53(2) :816–823, 2005.

[LPB+12] A. Liutkus, J. Pinel, R. Badeau, L. Girin, and G. Richard. Informed source separation through spectrogram coding and data embedding. *Signal Processing*, 92(8), august 2012.

[MBB+12] S. Marchand, R. Badeau, C. Baras, L. Daudet, D. Fourer, L. Girin, S. Gorlow, A. Liutkus, J. Pinel, G. Richard, N. Sturmel, and S. Zang. Dream : A novel system for joint source separation and multitrack coding. In *Audio Engineering Society Convention 133*, October 2012.

[MBCM10] B. Mathon, P. Bas, F. Cayre, and B. Macq. Considering security and robustness constraints for watermark-based tardos fingerprinting. In 46-51, editor, *IEEE International Workshop on Multimedia Signal Processing*, October 2010.

[MBZJ09] H. Mohimani, M. Babaie-Zadeh, and C. Jutten. A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm. *IEEE Transactions on Signal Processing*, 57(1) :289–301, january 2009.

[MF03] H.S. Malvar and D.A.F. Florencio. Improved spread spectrum : a new modulation technique for robust watermarking. *IEEE Transactions on Signal Processing*, 51(4) :898 – 905, apr 2003.

[MG83] B. C. J. Moore and B. R. Glasberg. Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *Journal of the Acoustical Society of America*, 74(3) :750–753, 1983.

[MK05] P. Moulin and R. Koetter. Data-hiding codes. *Proceedings of the IEEE*, 93(12) :2083 –2126, december 2005.

[MLAS10] M. Moussallam, P. Leveau, and S.M. Aziz Sbai. Sound enhancement using sparse approximation with speclets. In *IEEE International Conference on Acoustics and Speech, Signal Processing*, pages 221–224, march 2010.

[MW98] V.K. Madisetti and D.B. Williams, editors. *The Digital Signal Processing Handbook*, chapter Auditory Psychophysics for Coding Applications. CRC Press, 1998.

[NP08] A. Nesbit and M.D. Plumbley. Oracle estimation of adaptive cosine packet transforms for underdetermined audio source separation. In *IEEE International Conference on Acoustics and Speech, Signal Processing*, pages 41–44, april 2008.

[OS75] A. V. Oppenheim and R. W. Schafer. *Digital Signal Processing*. Prentice Hall, 1st edition, 1975.

[Pat76] R. D. Patterson. Auditory filter shapes derived with noise stimuli. *Journal of the Acoustical Society of America*, 59 :640–654, 1976.

[PB86] J. Princen and A. Bradley. Analysis/synthesis filter bank design based on time domain aliasing cancellation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(5) :1153 – 1161, October 1986.

[PBD+10] M.D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M.E. Davies. Sparse representations in audio and music : From coding to source separation. *Proceedings of the IEEE*, 98(6) :995–1005, 2010.

[PG11a] M. Parvaix and L. Girin. Informed source separation of linear instantaneous under-determined audio mixtures by source index embedding. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6) :1721–1733, 2011.

[PG11b] J. Pinel and L. Girin. Sparsification of audio signals using the mdct / intmdct and a psychoacoustic model – application to informed audio source separation. In *Audio Engineering Society Conference 42*, Ilmenau (Germany), july 2011.

[PGB10a] M. Parvaix, L. Girin, and J. Brossier. A watermarking-based method for informed source separation of audio signals with a single sensor. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6) :1464–1475, 2010.

[PGB10b] J. Pinel, L. Girin, and C. Baras. Une technique de tatouage « haute capacité » pour signaux musicaux au format CD-audio. In *10ème Congrès Français d'Acoustique*, Lyon (France), avril 2010.

[PGB11a] J. Pinel, L. Girin, and C. Baras. A high-rate data hiding technique for audio signals based on IntMDCT quantization. In *14th Conference on Digital Audio Effects*, Paris (France), septembre 2011.

[PGB11b] J. Pinel, L. Girin, and C. Baras. Insertion de données à haut débit dans des signaux de musique basée sur la transformée IntMDCT. In *Colloque GRETSI*, Bordeaux (France), septembre 2011.

[PGBP10] J. Pinel, L. Girin, C. Baras, and M. Parvaix. A high-capacity watermarking technique for audio signals based on MDCT-domain quantization. In *20th International Congress on Acoustics*, Sidney (Australia), august 2010.

[PJB87] J. Princen, A. Johnson, and A. Bradley. Subband/transform coding using filter bank designs based on time domain aliasing cancellation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 12, pages 2161 – 2164, April 1987.

[PMMS92] B. Paillard, P. Mabilleau, S. Morissette, and J. Soumagne. Perceval : Perceptual evaluation of the quality of audio signals. *Journal of the Audio Engineering Society*, 40(1/2) :21–31, 1992.

[PS00] T. Painter and A. Spanias. Perceptual coding of digital audio. *Proceedings of the IEEE*, 88(4) :451 –515, april 2000.

[PSM82] R. Pickholtz, D. Schilling, and L. Milstein. Theory of spread-spectrum communications–a tutorial. *IEEE Transactions on Communications*, 30(5) :855 – 884, may 1982.

[PSM84] R. Pickholtz, D. Schilling, and L. Milstein. Revisions to "theory of spread-spectrum communications - a tutorial". *IEEE Transactions on Communications*, 32(2) :211 –212, february 1984.

[RS78] L. R. Rabiner and R. W. Schafer. *Digital processing of Speech Signals*. Prentice-Hall signal processing series. Prentice Hall, 1st edition, 1978.

[Sam13] I. Samaali. *Tatouage pour le renforcement de la qualité audio des systèmes de communication bas débit*. PhD thesis, École nationale d'ingénieurs de Tunis et Université Paris Descartes, 2013.

[Sch64] L. Schuchman. Dither signals and their effect on quantization noise. *IEEE Transactions on Communication Technology*, 12(4) :162 –165, december 1964.

[Sch82] R. Scholtz. The origins of spread-spectrum communications. *IEEE Transactions on Communications*, 30(5) :822 – 854, may 1982.

[SD13] N. Sturmel and L. Daudet. Informed source separation using iterative reconstruction. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(1) :178–185, 2013.

[SLP+12] N. Sturmel, A. Liutkus, J. Pinel, L. Girin, S. Marchand, G. Richard, R. Badeau, and L. Daudet. Linear mixing models for active listening of music productions in realistic studio conditions. In *Audio Engineering Society Convention 132*, Budapest (Hungary), april 2012. Best Paper Award.

[SMT12] I. Samaali, G. Mahé, and M Turki. Watermark-aided pre-echo reduction in low bit-rate audio coding. *Journal of the Audio Engineering Society*, 60(6) :431–443, 2012.

[Spo97] Thomas Sporer. Objective audio signal evaluation-applied psychoacoustics for modeling the perceived quality of digital audio. In *Audio Engineering Society Convention 103*, 9 1997.

[STHAM09] I. Samaali, M. Turki-Hadj Alouane, and G. Mahé. Temporal envelope correction for attack restoration in low bit-rate audio coding. In *17th European Signal Processing Conference*, pages 929–933, Glasgow (Scotland), 2009.

[STHAM10] I. Samaali, M. Turki-Hadj Alouane, and G. Mahé. Attack restoration in low bit-rate audio coding, using an algebraic detector for attack localization. In *5th International Symposium on I/V Communications and Mobile Network*, pages 1–4, 2010.

[Stu95] J.R. Stuart. A proposal for the high-quality audio application of high-density cd carriers. *Technical Subcommittee Acoustic Renaissance for Audio.[Online] Available* `http://www.meridian.co.uk/ara/ara13.pdf`, 1995.

[Ter79] E. Terhardt. Calculating virtual pitch. *Hearing Research*, 1 :155–182, 1979.

[TF05] Y.F.V. Tan and C. Févotte. A study of the effect of source sparsity for various transforms on blind audio source separation performance. In *International Workshop on Signal Processing with Adaptative Sparse Structured Representations*, Rennes, France, 2005.

[TK96] T. Thiede and E. Kabot. A new perceptual quality measure for bit-rate reduced audio. In *Audio Engineering Society Convention 100*, 5 1996.

[TNM90] K. Tanaka, Y. Nakamura, and K. Matsui. Embedding secret information into a dithered multi-level image. In *IEEE Military Communications Conference*, volume 1, pages 216–220, 1990.

[TRS+93] A.Z. Tirkel, G.A. Rankin, R.M. Van Schyndel, W.J. Ho, N.R.A. Mee, and C.F. Osborne. Electronic watermark. *Digital Image Computing, Technology and Applications*, pages 666–673, 1993.

[Wan84] Zhongde Wang. Fast algorithms for the discrete W transform and for the discrete fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(4) :803–816, aug 1984.

[WLDB08] K. Wang, G. Lavoue, F. Denis, and A. Baskurt. Hierarchical watermarking of semiregular meshes based on wavelet transform. *IEEE Transactions on Information Forensics and Security*, 3(4) :620–634, 2008.

[WPD99] R.B. Wolfgang, C.I. Podilchuk, and E.J. Delp. Perceptual watermarks for digital images and video. *Proceedings of the IEEE*, 87(7) :1108 –1126, jul 1999.

[ZF92] R. Zamir and M. Feder. On universal quantization by randomized uniform/lattice quantizers. *IEEE Transactions on Information Theory*, 38(2) :428 –436, mar 1992.

[ZSE02] R. Zamir, S. Shamai, and U. Erez. Nested linear/lattice codes for structured multiterminal binning. *IEEE Transactions on Information Theory*, 48(6) :1250 –1276, june 2002.

[Zwi61] Eberhard Zwicker. Subdivision of the audible frequency range into critical bands. *The journal of the acoustical society of america*, 33(2) :248, february 1961.

Résumé

Les travaux de recherche présentés dans ce mémoire de thèse de doctorat traitent principalement de tatouage pour la transmission de données pour les signaux audionumériques non compressés (format PCM). Contrairement à ce qui est fait habituellement en tatouage, où les applications sont sécuritaires, les contraintes de robustesse et de sécurité sont ici très faibles et l'on s'intéresse ici principalement à maximiser le débit d'insertion sous contrainte d'inaudibilité. Les recherches se sont déroulées au sein du projet DReaM, dans le cadre de la séparation de sources informée, mais le système développé a pour but d'être utilisable dans n'importe quelle application d'enrichissement de contenu. Le système développé s'inspire du codage audio perceptuel, et repose sur une double insertion QIM appliquée sur les coefficients MDCT (ou IntMDCT) du signal audio, guidée par un modèle psychoacoustique. Une variante du système permettant de rendre parcimonieux la représentation MDCT d'un signal audio est aussi présentée.

Mots-clés : Tatouage, quantification, QIM, temps-fréquence, MDCT, IntMDCT, traitement du signal audionumérique, codage audio perceptuel, psychoacoustique, représentations parcimonieuses.

Abstract

The research work presented in this PhD dissertation deals mainly with watermarking for data transmission in the case of uncompressed digital audio signals (PCM format). Contrary to what is usually done in watermarking where secure applications are considered, robustness and security constraints are very low here and the main interest is to maximize the embedding bitrate under an inaudibility constraint. The research was done within the DReaM project, within the informed source separation framework, however the developed system targets any enriched-content application for audio signals. The developed system is inspired by perceptual audio coding, and is based on a double QIM insertion applied to the MDCT (or IntMDCT) coefficients of the audio signal guided by a psychoacoustic model. An adaptation of the system whose goal is to give a sparse MDCT representation of an audio signal is also presented.

Keywords : Watermarking, quantization, QIM, time-frequency, MDCT, IntMDCT, digital audio signal processing, perceptual audio coding, psychoacoustics, sparse representations.