**Rapport final — Mithra DJAHANBANI**

**Ecole Centrale de Marseille - Université Aix-Marseille**

Troisième année - Parcours MECA-AISE

Master 2 - Mécanique, Physique et Ingénierie, mention Acoustique

Année Universitaire 2015-2016

# Codeur-décodeur audio expérimental par transformée ERB-MDCT

**- Organisme d'accueil -**

Centre National de la Recherche Scientifique
Laboratoire de Mécanique et d'Acoustique (L.M.A.) (UPR-7051)
Equipe Sons

31 Chemin Joseph Aiguier
13402 Marseille Cedex 20

**Encadrant**
– Olivier DERRIEN — Maître de Conférences, Université de Toulon

**Rapporteurs**
– Serge MENSAH — Maître de Conférences, Ecole Centrale de Marseille
– Olivier MACHEREY — Chargé de Recherche, LMA



# Remerciements

Je tiens tout d'abord à remercier Olivier Derrien, pour m'avoir donné l'opportunité de réaliser ce stage très formateur. Il s'est montré très pédagogue, très disponible, tout en me faisant confiance et en me laissant une grande liberté en terme de prises d'initiatives.

Je remercie également Ichrak Toumi, pour ses conseils, et la grande patience dont elle a fait preuve en début de stage pour m'expliquer certaines notions obscures en traitement du signal...

Enfin, un grand merci à toute l'équipe pour l'accueil, et pour la très bonne ambiance qui y règne !

# Table des matières

# Introduction

L'apparition du CD (Compact Disc) dans les années 80 a été le point de départ en matière de codage audio. Il s'agit d'un système de codage conservant toute l'information contenue dans le signal analogique de départ (codage sans pertes), dans une bande passante de 0 à 22 kHz. S'il garantit une bonne qualité audio, il présente en contrepartie l'inconvénient de stocker un très grand nombre de données. Suite à l'apparition de nouvelles technologies, notamment sans fil, ne pouvant se permettre de traiter des quantités de données aussi importantes, le domaine du codage audio s'est rapidement retrouvé face au défit de réduire considérablement les quantités de données transmises ou stockées, et sont alors apparus les codeurs audio compressifs.

L'interopérabilité nécessaire des différents dispositifs (codeurs et décodeurs) a conduit à la mise en place de la norme ISO/IEC MPEG (Moving Picture Experts Group), sous laquelle différentes versions de codeurs compressifs se sont succédées :
- La série des MPEG-1, qui a été déclinée en trois versions (couches) : le MPEG-1 couche 1, couche 2 puis couche 3 (communément appelé "MP3"), qui ont permis d'atteindre des débits de plus en plus faibles, selon la nécéssité de l'application (broadcast, stockage de données...).
- Le codeur MPEG-2, très similaire au MPEG-1 couche 3, mais épuré des contraintes liées aux autres couches du MPEG-1. La version la plus récente du MPEG-2, dénommée "MPEG AAC" (MPEG Advanced Audio Coding), est aujourd'hui une référence en matière de codage audio.

Si le codec MPEG-AAC présente un bon compromis compression de données/-qualité audio, les recherches actuelles proposent de nouvelles pistes afin de poursuivre cette optimisation, notamment en s'appuyant sur des considérations perceptives relatives à l'audition humaine.

Dans le cadre de ce stage, on se propose, à partir d'un codec AAC, d'implémenter un codeur expérimental qui utilise une transformée temps-fréquence récemment mise au point : la ERB-MDCT, dont la résolution fréquentielle s'adapte à l'échelle perceptive ERB. On attend de ce codeur qu'il soit, à taux de compression identique, meilleur que le codec AAC en terme de qualité sonore.

Ce rapport s'organise de la manière suivante : Dans une première partie, l'état de l'art utile à la compréhension de la problématique et du travail réalisé est présenté. La seconde partie est consacrée au travail d'implémentation qui a été effectué afin de mettre en place le codeur expérimental. La dernière partie se penche enfin sur la validation expérimentale des codeurs et des outils de comparaison, avant d'aborder les études comparatives qui ont été réalisées.

# Chapitre 1

# Etat de l'art du codage audio numérique

Avant de présenter les codeurs audio destructifs classiques (de type MPEG), on présente tout d'abord le codage PCM (Pulse Code Modulation) [1], qu'utilise notamment le CD (codage sans compression de données), et dont les différentes étapes permettent d'introduire toutes les notions nécessaires à la compréhension du fonctionnement des codeurs destructifs MPEG.

## 1.1 Codage sans pertes (PCM)

Les étapes du codage PCM sont schématisées ci-dessous :

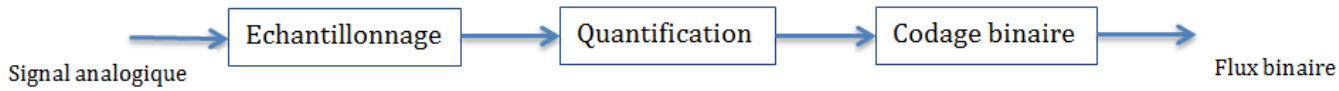



Figure *1.1 – Etapes du codage PCM.*

1 - Le signal analogique est tout d'abord échantillonné : On obtient un signal numérique (une succession d'échantillons), caractérisé par sa fréquence d'échantillonnage.

2 - Ce signal numérique nécessite ensuite d'être discrétisé en amplitude, car on ne peut pas coder une amplitude continue (infinité de valeurs) : c'est l'étape de la quantification, qui consiste à approcher les amplitudes réelles des échantillons par un ensemble discret et fini de valeurs. Ces valeurs sont appelées « niveaux de quantifications ». Le codage PCM réalise une quantification uniforme, i.e. les niveaux de quantifications (lignes en pointillées horizontales) sont espacés d'un pas constant Δ.

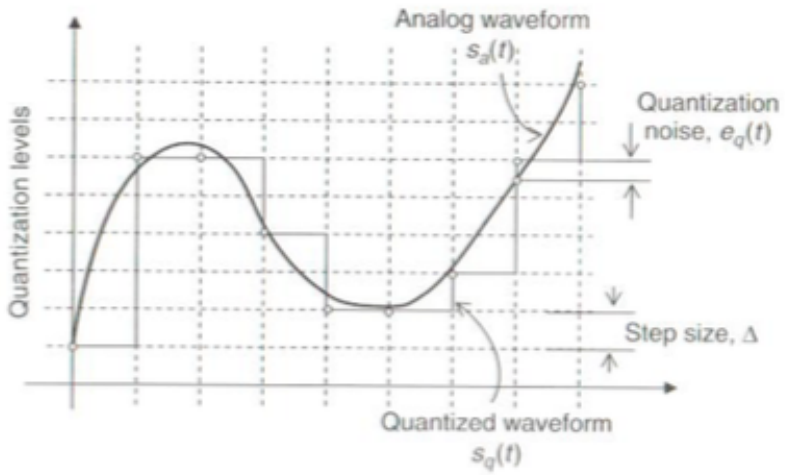



Figure *1.2 – Schéma de la quantification uniforme [1].*

3 - Ces amplitudes quantifiées sont ensuite codées en binaire. Selon le nombre de niveaux souhaités, un certain nombre de bits est donc nécessaire. Un nombre de bits b permet de coder $2^b$ niveaux de quantifications.
On juxtapose alors tous ces « mots » binaires représentant les amplitudes successives les uns après les autres, ce qui forme ainsi un train binaire qui va pouvoir être transmis au décodeur.

Débit

Le débit, exprimé usuellement en kbits/s, est le nombre de bits par échantillon multiplié par la fréquence d'échantillonnage : débit = b x $F_{ech}$.
Le CD, codé sur 16 bits avec une fréquence d'échantillonnage de 44100 Hz, a donc un débit de 705,6 kbits/s (mono), soit le double en stéréo.

Bruit de quantification

Le schéma précédent (figure 1.2) permet d'introduire le bruit de quantification $e_q(t)$, qui est l'erreur entre l'amplitude réelle de l'échantillon et l'amplitude quantifiée. Pour une quantification uniforme, la variance (puissance) du bruit de quantification est reliée au pas de quantification Δ par la formule suivante :

$$\sigma_{e_q}^2 = \frac{\Delta^2}{12} \tag{1.1}$$

Cette puissance du bruit de quantification peut, au delà d'un certain seuil, être perçue, entraînant de la distorsion sonore. A partir de la formule ci-dessus, on peut montrer que, pour une quantification uniforme, l'augmentation d'un bit permet d'augmenter le rapport signal-sur-bruit (SNR) de 6dB [1], et on comprend donc bien l'enjeu du nombre de bits (et donc du débit) sur le bruit de quantification et par conséquent sur la qualité du son restitué. Le CD, avec un débit de plus de 700kbits/s (en mono), est une référence en matière de qualité audio (on parle souvent de qualité CD).

Remarque : Selon la loi de probabilité d'apparition des différents amplitudes, on peut effectuer une quantification non uniforme qui s'adapte à cette loi de probabilité, afin de minimiser la puissance de l'erreur de quantification. Cependant, une telle méthode étant trop coûteuse en ressources de calcul, elle n'est pas utilisée pour le codage PCM.

## 1.2 Structure générale d’un codeur destructif perceptuel

Les nouvelles technologies, que ce soit pour du stockage de données ou du streaming, ne peuvent pas fonctionner à débits aussi élevés. L’objectif initial des normes MPEG 1 a été fixé d’atteindre les 128 kbits/s en stéréo, ce qui nécessite de compresser considérablement l’information. Afin d’atteindre de tels débits tout en conservant une bonne qualité audio, des codeurs destructifs dits "perceptuels" ont vu le jour, dont l’idée repose sur la suppression de l’information qui n’est pas perçue par l’oreille humaine.

Un codeur perceptuel classique se compose des modules présentés ci-dessous (figure 1.3) :

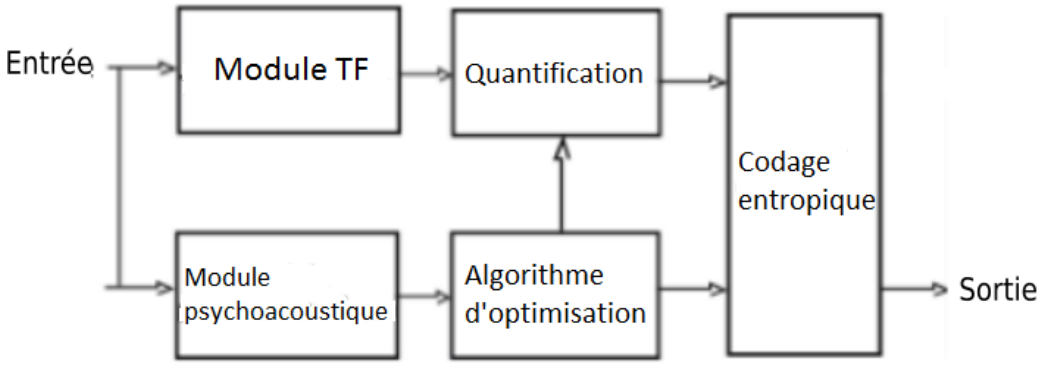



Figure *1.3 – Schéma d’un codeur destructif perceptuel [1].*

1 - La transformée de Fourrier classique ne permettant pas de décrire l’évolution temporelle du spectre d’un signal, le codeur réalise une transformée temps-fréquence, qui permet de représenter l’énergie du signal dans un plan temps-fréquence. Les coefficients spectraux générés par cette transformée vont ensuite être utilisés dans les autres modules.

2 - On utilise ensuite un modèle perceptif (module psychoacoustique) qui va permettre de déterminer l’importance perceptive des coefficients spectraux issus de l’analyse temps-fréquence.

3 - Afin de minimiser la quantité de données transmises tout en conservant une bonne qualité audio, on va alors quantifier de manière optimale (algorithme d’optimisation) les coefficients spectraux, en prenant en considération leur importance perceptive.

4 - On réalise enfin un codage entropique (codage de Huffman [11]) afin de supprimer la redondance statistique résiduelle.

Les paragraphes suivants reviennent sur ces différents modules, d’un point de vue théorique, et leur structure sera détaillée de manière concrète pour les différents codeurs utilisés durant ce stage.

### 1.2.1 Analyse Temps-Fréquence

#### 1.2.1.1 Transformée de Gabor

La transformée de Gabor a été la première transformée temps-fréquence proposée afin de décrire l'évolution temporelle des composantes spectrales d'un signal. L'idée de cette transformée est de fenêtrer temporellement le signal et d'appliquer la transformée de Fourrier à l'intérieur de cette fenêtre, puis de translater cette fenêtre tout au long de la durée du signal.
L'analyse du signal peut alors être vue comme l'application d'un produit scalaire entre le signal et une famille de fonctions appelés "atomes" temps-fréquence, qui correspondent aux exponentielles complexes de l'analyse de Fourrier, fenêtrées, et translatées en temps [10].

Chaque atome de Gabor, caractérisé par un indice temporel n et un indice fréquentiel m, est défini de la manière suivante :

$$g_{m,n}(t) = w(t - \alpha n)e^{2i\pi\beta mt} \tag{1.2}$$

où w(t) est la fenêtre, $\alpha$ le pas temporel de translation de la fenêtre d'analyse, et $\beta$ le pas de modulation fréquentielle de l'exponentielle complexe de Fourrier [8].

Cette famille d'atomes forme un repère de Gabor, qui peut être représenté par une grille sur un plan temps-fréquence (figure 1.4).

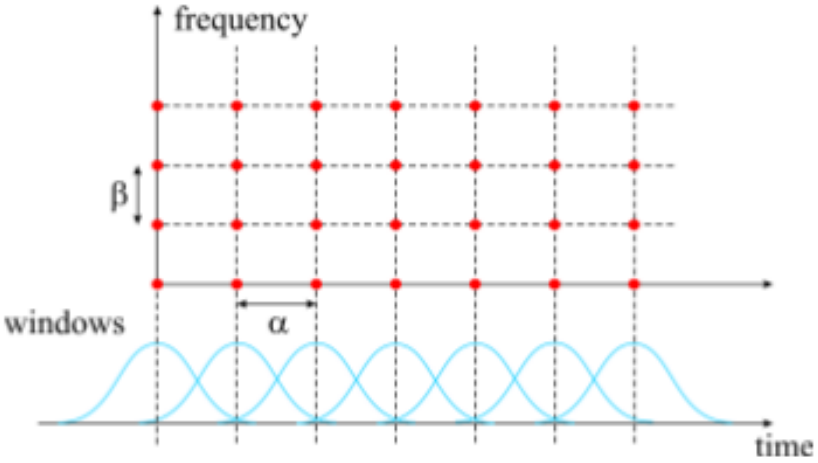



Figure *1.4 – Grille temps-fréquence d'un repère de Gabor [5].*

On définit comme $\rho = \frac{1}{\alpha\beta}$ la redondance du repère de Gabor.
Afin d'obtenir une reconstruction parfaite du signal lors de la transformée inverse, il est nécessaire, pour un repère de Gabor, d'avoir $\rho > 1$ (théorème de Balian-Low) [10].

Dans le cas extrême ($\rho = 1$), un repère de Gabor engendre a minima une redondance de 2 au sens du débit de données, i.e la quantité de coefficients spectraux est double par rapport au nombre d'échantillons dans le signal d'origine. Cette redondance étant considérablement sous-optimale en terme d'efficacité de codage, la transformée de Gabor n'est pas utilisée pour le codage audio.

#### 1.2.1.2 Application au codage audio : la transformée MDCT

Il a été montré que pour avoir une redondance nulle au sens du débit de données, il était nécessaire d'avoir une redondance (au sens de Gabor) de $\rho = \frac{1}{2}$, ce qui est impossible avec un repère de Gabor, soumis au théorème de Balian-Low ($\rho > 1$).

Une nouvelle transformée a été proposée, la MDCT, qui permet d'atteindre cette redondance $\rho = \frac{1}{2}$, et donc d'être une transformée non redondante en terme de codage. Cette transformée est dérivée de la DCT (Discrete Cosine Transform), qui est un type de transformées décomposant le signal, non pas sur une base d'exponentielles complexes, comme pour les repères de Gabor, mais sur une base de cosinus, qui n'est pas soumise au théorème de Balian-Low. Cette DCT est "modifiée" afin d'être adaptée au codage audio : la MDCT (Modified DCT) est sous-échantillonnée en fréquence (nombre de coefficients spectraux divisé par 2) afin d'atteindre cette redondance de $\rho = \frac{1}{2}$.

Les atomes de la MDCT sont définis de la manière suivante [8] :

$$\phi_{m,n}(t) = w(t - an)cos[\frac{\pi}{a}(t - an)(m + \frac{1}{2})] \tag{1.3}$$

Cette famille d'atomes peut former une base, permettant une reconstruction parfaite du signal. Le sous-échantillonnage en fréquence qui permet d'atteindre la redondance de $\rho = \frac{1}{2}$ est en fait compensé par de fortes contraintes sur la fenêtre w, qui sont satisfaites, dans le cas de la MDCT, si on choisit une fenêtre dérivée de la fonction sinus.

En effet, un sous-échantillonnage en fréquence génère a priori un repliement dans le domaine temporel lorsqu'on effectue la transformée inverse. Cependant, les spécificités liées ici à la décomposition sur une famille de cosinus et au choix de la fenêtre, donnent à la MDCT des propriétés de symétrie et d'anti-symétrie qui permettent une suppression de ce repliement.

Ce principe caractéristique de la MDCT est appelé *Time Domain Aliasing Cancelation* (TDAC) et est schématisé figure 1.5.
Si on considère uniquement la première fenêtre d'analyse "Window 1" (graphe (a)), on voit que lors de la reconstruction dans le domaine temporel (graphe (b)), le signal d'origine est superposé avec une version symétrique (temporellement) est inversée de lui-même sur sa première moitié, et une version symétrique sur sa deuxième moitié. Il en est de même pour la "Window 2" (graphe (c)).
Si maintenant on effectue une addition-recouvrement de moitié des deux fenêtres lors de la synthèse dans le domaine temporel (donc une addition des graphes (b) et (c)), on voit que ces artefacts se compensent et on retrouve le signal d'origine sur la portion où les fenêtres se recouvrent (graphe (d)).

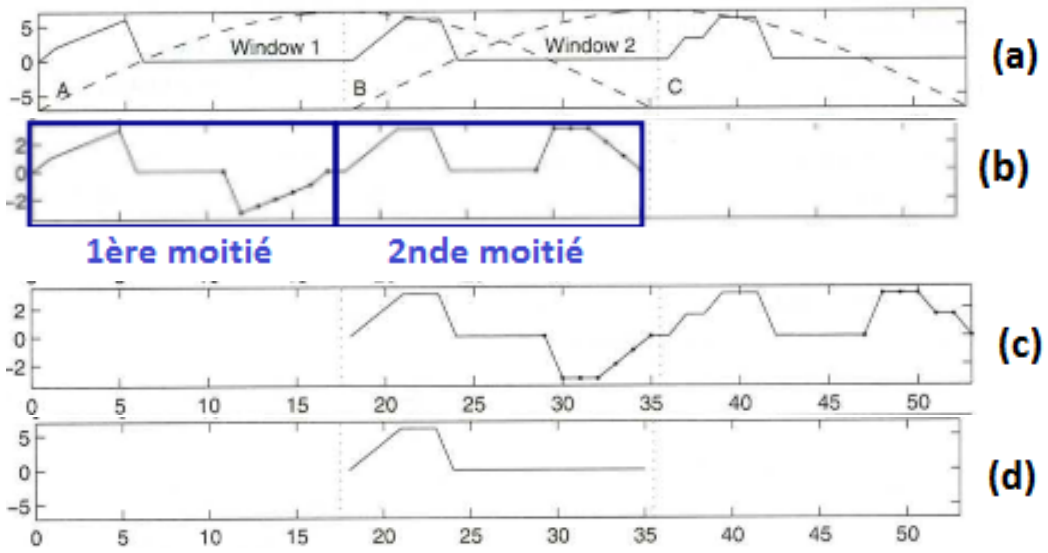



Figure *1.5 – Principe du Time Domain Aliasing Cancelation grâce à une addition-recouvrement de moitié de deux fenêtres de MDCT [9].*

Pour ses avantages en terme de réduction de données, la MDCT est utilisée dans la plupart des codecs utilisant une transformée temps-fréquence.

### 1.2.2 Modèle psychoacoustique

Pour comprendre comment est supprimée l'information inaudible, il est nécessaire de comprendre au préalable le fonctionnement de l'oreille. En première approximation, on peut considérer que la cochlée se comporte comme un banc de filtres passe-bandes se recouvrant. Lorsque l'oreille est excitée par une composante fréquentielle, la membrane basilaire va être excitée sur toute une portion de la cochlée centrée autour d'une position où la résonance est maximale. Cette bande fréquentielle est appelée bande critique. Bien que l'oreille puisse former une bande critique autour de n'importe quelle fréquence (la fonction "bande critique" est donc continue), on considère souvent d'une façon un peu artificielle que l'espace des fréquences est partitionné en 24 bandes aux frontières fixes (figure 1.6).

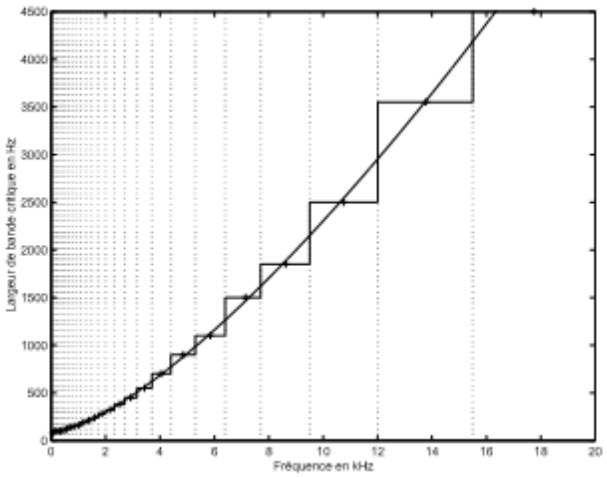



Figure *1.6 – Largeur des bandes critiques en fonction de la fréquence. Partition de l'axe fréquentiel par juxtaposition de 24 bandes critiques [2].*

L’analyse des signaux audio en bandes critiques a une résolution fréquentielle trop faible pour appliquer correctement un modèle d’audition. En pratique, on préfère utiliser d’autres bancs de filtres perceptifs comme celui défini par l’échelle ERB (Equivalent Rectangular Bandwidth), qui compte initialement 43 bandes au lieu de 24. Les fréquences centrales de ces bandes sont agencées de manière à suivre approximativement une progression linéaire jusqu’à 500 Hz, puis une progression logarithmique au dessus de 500 Hz. La largeur de la bande ERB en fonction de la fréquence, ainsi que l’indice de cette bande, sont respectivement calculés par les formules (1.4) et (1.5) [4]. Par généralisation, on peut étendre cette définition aux indices ERB non-entiers (exemple figure 1.7), ce qui permet de définir un nombre quelconque de bandes de fréquence par bande ERB .

$$ERB(F) = 24.7 + \frac{F}{9.265} \tag{1.4}$$

$$ERB_{num}(F) = 9.265 ln(1 + \frac{F}{228.8455}) \tag{1.5}$$

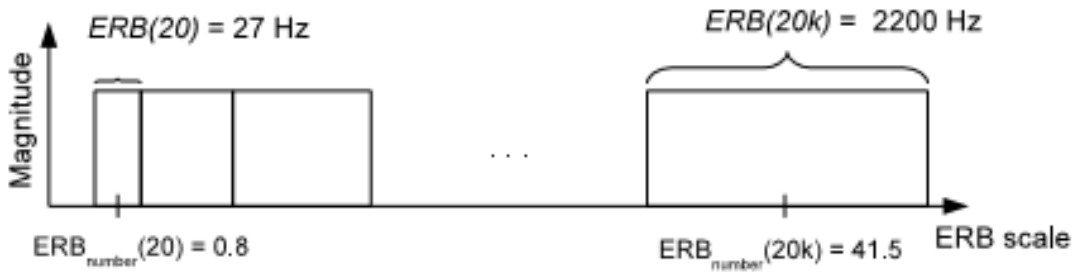



Figure *1.7 – Largeur de bande ERB et indice de bande ERB [7].*

Le fonctionnement de l’oreille peut entraîner un phénomène de masquage d’un son par un autre son lorsqu’ils excitent l’oreille en même temps. En effet, d’un point de vue physique, si une composante tonale ou bruit excite fortement la membrane basilaire, cette dernière ne va potentiellement pas détecter la présence d’une excitation plus faible apparaissant au sein de cette même portion de membrane correspondant à une bande critique [1].
Numériquement, ce phénomène se traduit de la manière suivante : On considère un son masquant donné, de fréquence centrale $f_0$ et et de puissance $\sigma_0^2$, et un son masqué, de fréquence centrale f et de puissance $\sigma^2$. Selon la position fréquentielle f du son masqué, il existe une valeur limite de sa puissance $\sigma^2$ en dessous de laquelle la perception de la somme du son masqué et du son masquant est la même que celle du son masquant [2].
La variation de $\sigma^2$ (puissance minimale que doit avoir le son masqué pour être perçu) en fonction de f (fréquence du masqué) est appelée courbe de masquage (exemple figure 1.8).

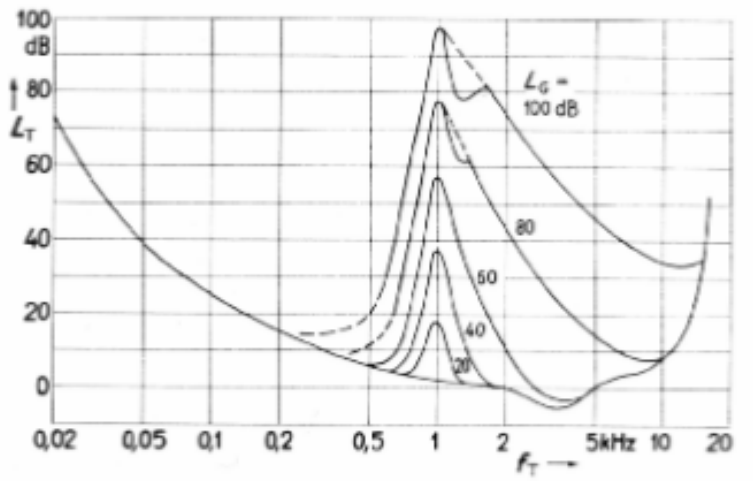



Figure *1.8 – Courbe de masquage (en dB SPL) pour un bruit masquant de fréquence centrale 1kHz [2].*

On peut alors déterminer le seuil de masquage global (figure 1.9, à droite) de la portion de signal analysé. Habituellement, pour calculer ce seuil, on combine, sur chaque sous-bande d'analyse spectrale, les courbes de masquage engendrées par toutes les raies spectrales de la sous-bande, par addition des énergies en décibels, mais il a été prouvé que cette approximation n'est pas tout-à-fait exacte, et que l'additivité du masquage fait intervenir des phénomènes très complexes qui n'ont pas encore été totalement élucidés.

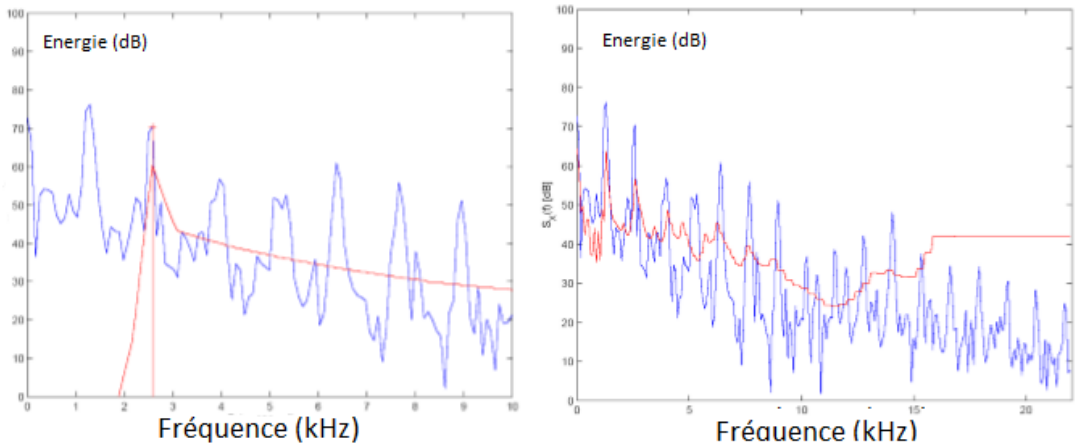



Figure *1.9 – Exemple d'une courbe de masquage générée par un son masquant (à gauche) et du seuil de masquage global combinant toutes les courbes de masquage de toutes les composantes spectrales (à droite). Ces courbes sont superposées au spectre du signal dans les deux cas [6].*

Lorsque le masquant n'est plus présent, le phénomène de masquage décroît progressivement mais ne disparaît pas immédiatement. On parle alors de masquage temporel.

Le modèle psychoacoustique utilisé dans le cadre de ce stage, et dont l'implémentation est décrite dans les chapitres suivants, prend en compte l'aspect temporel du masquage puisqu'il utilise un pattern de masquage temps-fréquence.

### 1.2.3 Quantification et codage

Le processus de quantification doit répondre à deux contraintes [2] :
- Une contrainte de qualité perçue, fixée par le modèle psychoacoustique : dans chaque sous-bande de calcul du seuil de masquage, il est nécessaire de maintenir la puissance du bruit de quantification en dessous de ce seuil.
- une contrainte de débit : Le flux binaire total (nombre de bits de codage utilisés par unité de temps pour coder les données quantifiées mais aussi les informations annexes) doit conserver un débit fixe. C'est le débit brut. Le flux binaire concernant les données quantifiées est appelé débit net, et il est donc sous contrainte imposée par le débit brut.

Généralement, il n'est pas possible de satisfaire les deux contraintes simultanément. Selon le type d'application, le codeur peut fonctionner :
- à débit fixe (contrainte de débit prépondérante, qualité perçue sous-optimale).
- à débit variable et à qualité perçue optimale (contrainte de qualité prépondérante).

Afin de répondre à la contrainte de qualité perçue, tous les coefficients spectraux ne vont donc pas être quantifiés de la même manière. Or, les paramètres qui définissent les quantificateurs doivent également être transmis au décodeur, ce qui augmente le débit binaire. Il est donc souhaitable de limiter le nombre de quantificateurs utilisés. Les coefficients spectraux sont donc regroupés par sous-bandes fréquentielles avant quantification. Cette contrainte supplémentaire liée au codage binaire a eu pour conséquence la définition de sous-bandes pour la quantification et le codage, différentes des sous-bandes du modèle psychoacoustique.

Ce regroupement par sous-bandes de codage des coefficients spectraux se fait par une mise à l'échelle commune des coefficients de la sous-bande. Cette amplification ou diminution de l'amplitude des coefficients spectraux va modifier le bruit de quantification généré au moment de la quantification, afin d'approcher le bruit de quantification de cette sous-bande au plus près du seuil de masquage.

Le module de quantification est donc composé d'un algorithme d'optimisation dont le but est de déterminer les valeurs optimales de ces facteurs d'échelle par sous-bande, en prenant en compte les contraintes de débit et de masquage mentionnées précédemment.

Les valeurs quantifiées des coefficients sont ensuite codées de manière entropique, codage qui consiste à associer aux coefficients quantifiés les plus probables les mots binaires les plus courts (utilisant le plus petit nombre de bits). Ce processus de codage est en général réalisé par algorithme de Huffman [11].

Remarque : Tous les modules ne sont pas standardisés par la norme MPEG : Le choix du modèle psychoacoustique et de l'algorithme d'optimisation de la quantification sont libres d'implémentation.

## 1.3 Le codeur MPEG AAC

### 1.3.1 La MDCT du codeur AAC

La MDCT, telle qu'elle a été présentée précédemment, n'est pas tout à fait optimale en terme de qualité audio.
En effet, l'utilisation d'une fenêtre d'analyse de taille uniforme n'est pas adaptée. Si on utilise une fenêtre trop longue pour analyser un transitoire, il se produit un phénomène de pré-echo : le bruit qu'engendre la quantification dans le domaine transformé se répartit sur toute la longueur de la fenêtre lorsqu'on revient au domaine temporel. Une attaque (augmentation brutale du niveau du signal au milieu de la fenêtre) pourra alors être perçue avec de l'avance.

Ainsi, le codeur AAC a la particularité d'utiliser une MDCT qui commute entre deux tailles différentes de fenêtres d'analyse, selon les caractéristiques du signal :
- Une fenêtre courte, de 256 échantillons, permettant une très bonne résolution temporelle lorsque le signal présente un transitoire.
- Une fenêtre longue, de 2048 échantillons, permettant une bonne résolution fréquentielle pour les parties stationnaires du signal.

On a vu cependant que la MDCT nécéssitait une addition-recouvrement de moitié de ses fenêtres (principe de TDAC présenté précédemment) pour obtenir une reconstruction parfaite lors de la synthèse. Dans le cas du codeur AAC, où on commute entre deux fenêtres de tailles différentes, on insère donc des fenêtres de transition lors du passage d'une fenêtre courte à une fenêtre longue (et inversement), qui doivent respecter certaines propriétés de symétrie au niveau de l'addition-recouvrement [2], comme on peut le voir sur le schéma ci-dessous (figure 1.10). Ces fenêtres de transition sont construites de la manière suivante : elles présentent une arche de sinus correspondant au symétrique de la moitié de la fenêtre avec laquelle il y a recouvrement (fenêtre longue par exemple), une portion constante, puis une deuxième arche de sinus symétrique de la moitié de la deuxième fenêtre qu'elle recouvre (fenêtre courte).

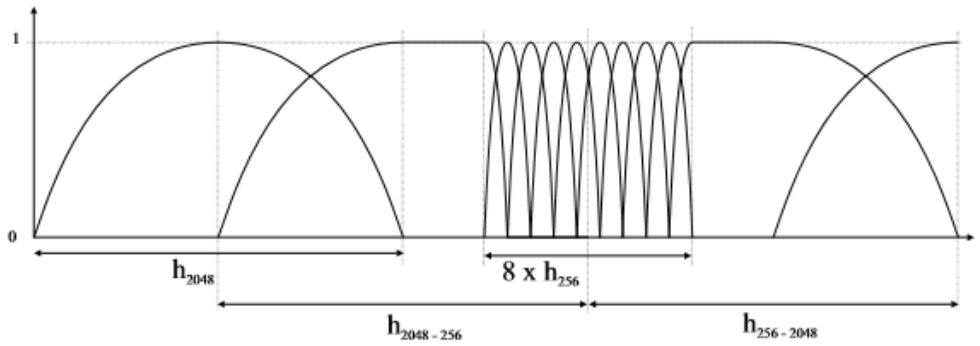



Figure *1.10 – Commutation entre deux tailles de fenêtres de MDCT, avec des fenêtres de transition respectant certaines symétries à l'endroit du recouvrement [2].*

Lorsqu'on veut passer d'une fenêtre longue à une fenêtre courte par exemple, on applique donc une MDCT stationnaire à la fenêtre longue (bloc long), à la fenêtre de transition, puis à la succession de fenêtres courtes (bloc court) et on raccorde ces blocs par addition-recouvrement. Pour raccorder un bloc long et un

bloc court, il est nécessaire que le bloc court ait au moins le même nombre de coefficients spectraux que le bloc long. Du point de vue de l'efficacité de codage, on choisit des blocs courts ayant exactement le même nombre de coefficients que les blocs longs. Dans le cas du codeur AAC, les fenêtres longues fournissant 1024 coefficients, contre 128 pour les fenêtres courtes, il est nécessaire d'avoir 8 fenêtres courtes d'affilé.

Cette MDCT à deux fenêtres du codeur AAC a permis d'obtenir un meilleur compromis au niveau de la résolution temps/fréquence, ainsi que de minimiser les phénomènes de pré-echo.

### 1.3.2 Implémentation

Ce paragraphe expose comment les différentes étapes du codage sont implémentées dans le cas du codeur AAC.

#### 1.3.2.1 Définition des blocs de codage pour la quantification

Sur chaque fenêtre d'analyse, on regroupe les coefficients spectraux, appelés ici coefficients MDCT, en blocs de codage (cf paragraphe 1.2.3). Sur chacun de ces blocs, les résultats du modèle d'audition sont utilisés pour réaliser la quantification optimale.

Concrètement, les coefficients sont regroupés de la manière suivante (figure 1.11) :
- Sur une fenêtre d'analyse longue ou fenêtre de transition, les 1024 coefficients sont répartis en 49 blocs de codage : Le nombre de coefficients par bloc augmente progressivement avec la fréquence, avec un minimum de 4 coefficients dans les bandes basse fréquence.
- Sur une fenêtre courte, les 128 coefficients sont répartis, selon le même principe, en 14 blocs.

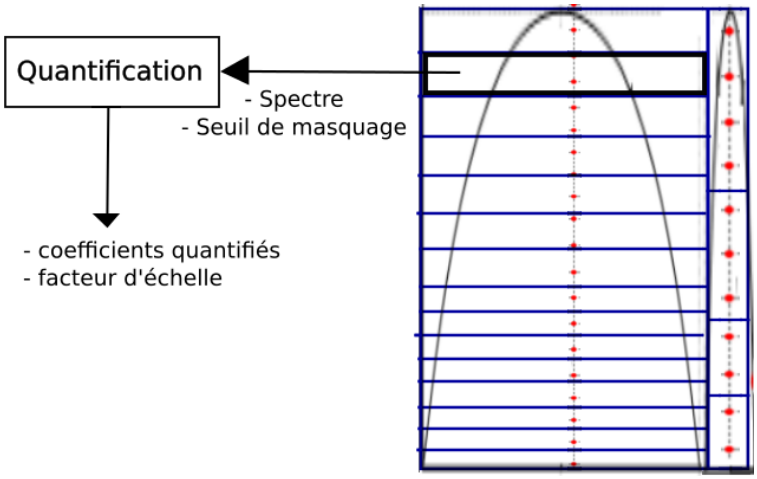



Figure *1.11 – Schéma simplifié de définition des blocs de codage sur fenêtre longue et fenêtre courte (les dimensions des blocs et les nombres de coefficients ne sont pas réalistes). Les points rouges représentent les coefficients MDCT, les rectangles bleus les blocs de codage.*

#### 1.3.2.2 Modèle psychoacoustique

Calcul du Spectre

Sur chaque bloc de codage, on calcule l'énergie des coefficients temps-fréquence :

$$E_x = \Sigma |X_k|^2 \tag{1.6}$$

où les $X_k$ représentent les k coefficients MDCT du bloc de codage.

Le spectre correspond à l'ensemble des énergies sur tous les blocs de codage d'une fenêtre d'analyse.

Calcul du SMR/Seuil de masquage

A partir du signal temporel contenu dans la fenêtre d'analyse et de la délimitation des blocs fréquentiels, le module psychoacoustique calcule un seuil de masquage (une valeur par bloc de codage).

En fait, le modèle psychoacoustique ne calcule pas directement le seuil de masquage, mais fournit l'écart entre le spectre et le seuil de masquage sur chaque bloc (cf écart entre les courbes bleue et rouge figure 1.9), appelé "Rapport Signal à Masque" (Signal-to-Mask Ratio ou SMR).

Par exemple, sur une fenêtre longue, on obtient en sortie de module psychoacoustique :
- un Spectre (vecteur contenant 49 valeurs).
- un SMR (vecteur contenant 49 valeurs).

Le seuil est déduit par soustraction (en dB) ou division (en échelle linéaire), du spectre et du SMR :

$$Seuil = \frac{Spectre}{SMR} \tag{1.7}$$

Remarque : Afin de déterminer le SMR, une analyse temps-fréquence est réalisée au sein du module psychoacoustique. Elle est différente de la transformée du module d'analyse TF. Une variété de modèles psychoacoustiques existent, utilisant différentes analyses fréquentielles.
Le modèle utilisé dans le cadre de ce stage, commun aux codeur AAC et au codeur expérimental, est décrit plus loin dans le rapport.

#### 1.3.2.3 Codage effectif et mesure du débit

On a vu que le débit total est composé de deux types de flux de données :
- Les coefficients MDCT quantifiés, qui portent l'information du signal, appelée **Information MDCT** dans la suite de ce rapport.
- Les données caractérisant les quantificateurs, nécessaires au décodage, appelées **Informations annexes** (en anglais, *Side information*).

Toutes ces données sont codées par codage entropique de Huffman, et c'est à partir de ce codage binaire qu'on peut connaître le débit effectif.

Cependant, cette étape, qui contient des tables de Huffman spécifiques au codeur AAC n'est pas adpatable telle quelle au codeur expérimental que l'on souhaite implémenter. Réaliser un nouveau codage de Huffman demanderait un temps conséquent et présente peu d'intérêt dans le cadre de ce stage de recherche. Ce qui nous intéresse pour cette étude comparative des codeurs, ce n'est pas d'avoir une valeur absolue du débit effectif des codeurs, mais de pouvoir les comparer de manière relative en terme de flux de données.

On a donc utilisé des moyens de substitution pour comparer les flux de données des différents codeurs.

Mesure de l'information annexe

Sur chaque bloc de codage, les coefficients MDCT sont mis à l'échelle avant d'être quantifiés. A un bloc de codage correspond donc un facteur d'échelle. Or, le volume total d'informations annexes à transmettre au décodeur est, en moyenne sur tous les blocs, proportionnel au nombre de facteurs d'échelle, donc au nombre de blocs de codage.
Ainsi, en utilisant un même nombre de blocs de codage pour le codeur AAC et pour le codeur expérimental, on pourra garantir un volume d'informations annexes à peu près équivalent.

Remarque : Cela n'est vrai qu'en moyenne, car, sur chaque bloc, la valeur du facteur d'échelle étant codée par un codage de Huffman, elle n'est donc pas codé avec le même nombre de bits selon sa probabilité d'apparition.

Mesure d'entropie comme estimation du débit MDCT

L'entropie d'un vecteur d'échantillons est la borne inférieure théorique du nombre de bits moyen par échantillon, calculé sur tous les codes binaires possibles.
Dans notre cas, en calculant l'entropie d'un vecteur de coefficients MDCT quantifiés, on obtient la borne inférieure théorique du nombre de bits moyen par coefficient MDCT quantifié.
Or, si mettre en place un codage de Huffman s'avère fastidieux, l'entropie d'un vecteur de nombres se calcule facilement et on dispose d'une fonction déjà implémentée permettant de la calculer.

En théorie, le débit réel est donc supérieur à cette mesure d'entropie. Pour justifier l'utilisation de l'entropie, il est cependant nécessaire de préciser cet écart. En début de troisième chapitre, on revient donc sur ce point, où une loi entre ces deux grandeurs est établie.

## 1.4 Introduction aux transformées non stationnaires

### 1.4.1 Transformée de Gabor non stationnaire : définition de l'ERBLet

Afin de palier le problème de résolution temps-fréquence, une alternative à la commutation de taille de MDCT a été proposée.
L'idée a été d'introduire des transformées dites non stationnaires, qui présentent une grille d'analyse temps-fréquence non uniforme afin de pouvoir adapter localement les résolutions temporelle et fréquentielle de la transformée, en suivant au plus près la sensibilité de l'oreille.

Il est en effet possible de définir une famille d'atomes de Gabor, cette fois-ci non uniforme, de la manière suivante [10] :

$$g_{m,n}(t) = w_n(t)e^{2i\pi\beta_n mt} \tag{1.8}$$

où $w_n(t)$ est centré sur $\alpha_n$ qui ne suit pas une progression régulière selon n.

On constate que la résolution fréquentielle représentée par $\beta$, ainsi que la forme de la fenêtre, dépendent de l'indice temporel n (cf figure 1.12).

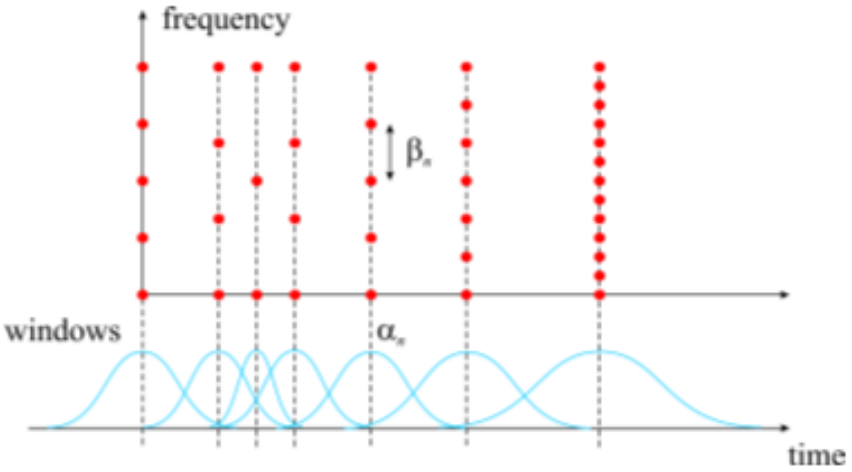



Figure *1.12 – Grille temps-fréquence d'une transformée de Gabor non stationnaire [5]*

L'adaptation continue de la résolution temporelle est en fait une généralisation de la commutation entre deux tailles de fenêtres utilisée dans le codeur AAC [8]. Cela n'empêche pas que la résolution fréquentielle reste localement uniforme en fréquence, ce qui est contraire au fonctionnement de l'oreille. L'idée est donc de reprendre le principe de la transformée non-stationnaire, tout en permutant les rôles des domaines temporels et fréquentiels au moyen d'une transformée de Fourier classique. Pratiquement, on définit une famille non-stationnaire d'atomes de Gabor dans le domaine fréquentiel, puis on revient dans le domaine temporel en appliquant une transformée de Fourier. On obtient donc une transformée non-uniforme en fréquence, dont la résolution temporelle est uniforme en temps, mais variable selon la fréquence [5].

C’est ce qui a été proposé avec la transformée ERBLet, dont la grille d’analyse est non-uniforme en fréquence et suit la progression de l’échelle ERB [4].

Comparaison ERBLet/ MDCT stationnaire

L’ERBLet a été comparée (figure 1.13) à une MDCT stationnaire. On constate qu’une telle transformée apporte bien une meilleure résolution fréquentielle en basses fréquences et une meilleure résolution temporelle en hautes fréquences qu’une transformée stationnaire classique.

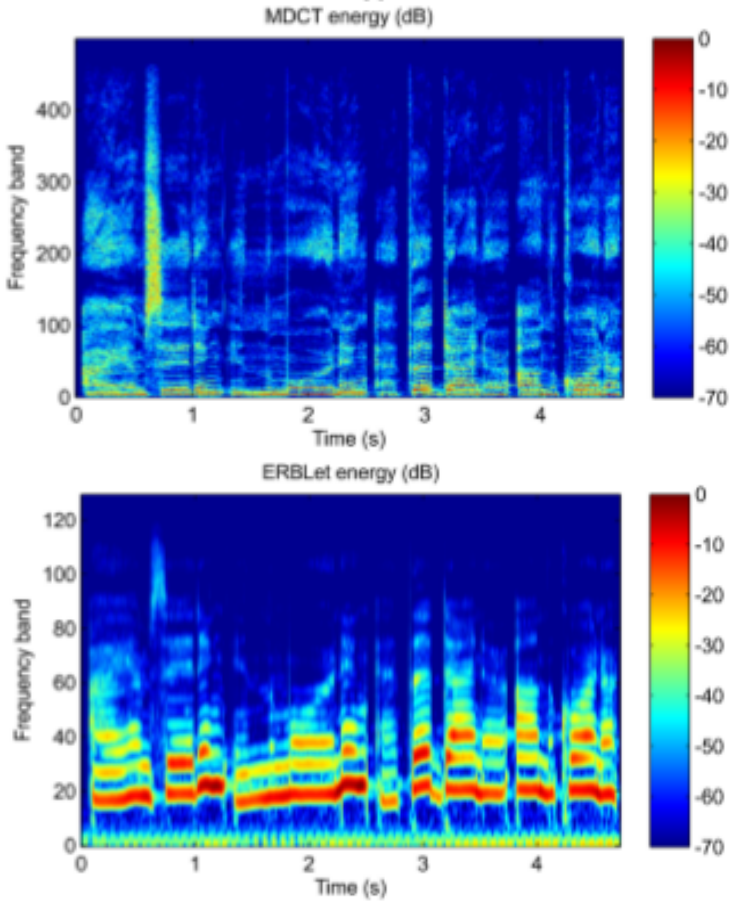



Figure *1.13 – Comparaison des modules (en décibels) d’une MDCT stationnaire (500 bandes fréquentielles d’analyse) et d’une ERBLet (128 bandes) [3].*

Pour les même raisons que dans le cas des transformées stationnaires, l’ERBLet n’est cependant pas envisageable pour le codage audio car elle est issue des transformées de Gabor, et est donc très redondante.

### 1.4.2 Transformée ERB-MDCT : Théorie et implémentation

Afin de pouvoir transposer le principe de l’ERBLet au domaine du codage audio, on a conservé le principe de construction de la ERBLet, mais en partant d’un couple MDCT/DCT pour les transformées de base, au lieu des transformées de Fourrier (exponentielles complexes). Le détail des procédures de définition et

d’implémentation de la ERB-MDCT sont trop complexes pour avoir leur place dans ce rapport. Pour plus de précision, voir [3].

On peut en effet définir une MDCT non-stationnaire (figure 1.14) au même titre qu’on a défini un repère de Gabor non stationnaire.

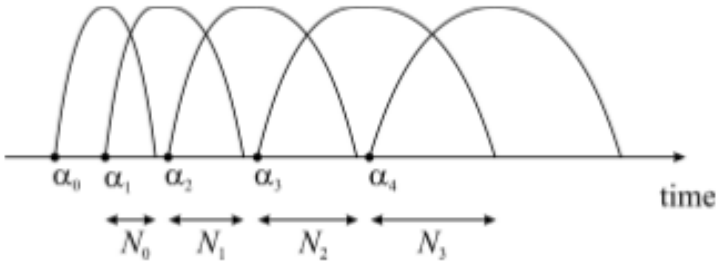



Figure *1.14 – MDCT non-stationnaire en temps suivant une progression non uniforme [5].*

Comme pour l’ERBLet avec la famille d’atomes de Gabor, on adapte ici la MDCT non-stationnaire en temps qui a déjà été théorisée dans la littérature pour définir une grille d’analyse non-uniforme en fréquence (figure 1.15).

L’implémentation de la ERB-MDCT est donc similaire à celle de l’ERBLet. A l’analyse, on applique d’abord une DCT (l’équivalent de Fourrier mais sur une base de cosinus) au signal, puis une MDCT non-stationnaire suivant l’échelle ERB dans le domaine fréquentiel. A la synthèse, on effectue d’abord la MDCT non-stationnaire inverse, puis une DCT inverse.

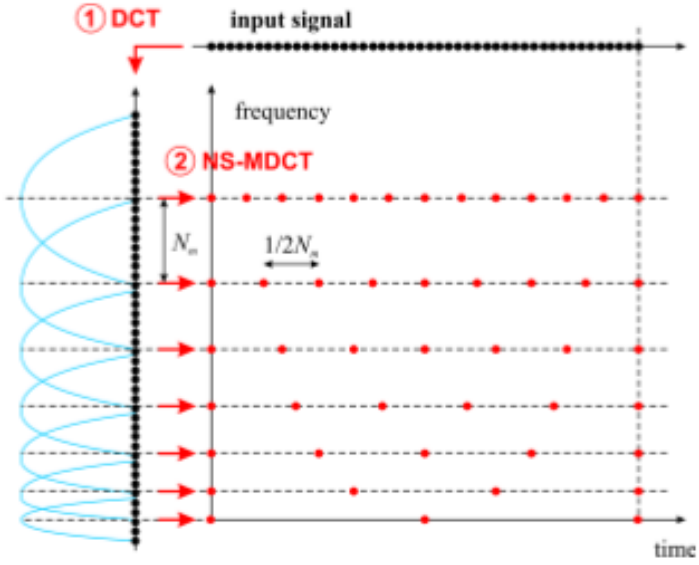



Figure *1.15 – Grille temps-fréquence d’une ERB-MDCT [5].*

On remarque que les coefficients MDCT générés ne sont pas alignés en temps les uns par rapport aux autres, contrairement aux transformées stationnaires ou non stationnaires en temps (comme la figure 1.12). Cela empêche de découper le signal en blocs verticaux très courts et d’appliquer un traitement (quantification et codage binaire) bloc par bloc, comme pour le codeur AAC (figure 1.16).

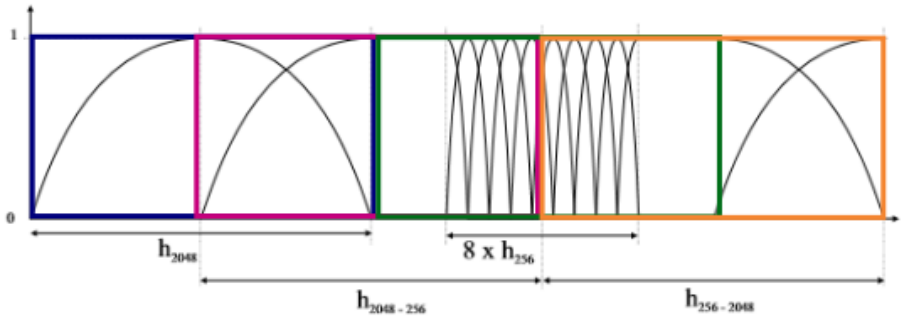



Figure *1.16 – Schéma de l'analyse bloc par bloc du codeur AAC. L'analyse-synthèse est effectuée sur des blocs de 2048 échantillons, puis on réalise une addition-recouvrement des signaux resynthétisés (cf principe du TDAC).*

En fait, le traitement bloc par bloc du codeur AAC répond à des contraintes d'application, où un codage "à la volée" est nécessaire (par exemple pour la diffusion en direct, aussi appelée "streaming"). Si on avait des contraintes d'application pour le codeur ERB-MDCT, nécessitant un codage bloc par bloc temporel assez court, on pourrait découper le signal temporel à analyser en macro-blocs temporels plus courts, mais le raccord entre ces blocs après la resynthèse introduirait un peu de redondance excédentaire au niveau des jonctions. Il est donc préférable de définir des macro-blocs les plus longs possibles, et c'est pourquoi on parle bien de "macro-blocs", car il ne s'agit pas de pouvoir découper le signal en blocs temporels aussi courts que ceux du codeur AAC.

Dans le cadre de ce travail de recherche, cependant, n'ayant pas de contrainte d'application en terme de temps de calcul, et travaillant sur des signaux temporels assez courts (de l'ordre de quelques secondes), ce problème ne se pose pas, et on traitera nos signaux en un seul bloc (dont la taille fait donc toute la durée du signal).

### Comparaison ERB-MDCT/ MDCT stationnaire

La transformée ERB-MDCT est d'ores et déjà implémentée sous Matlab, et elle a été comparée (figure 1.17) à une MDCT non-stationnaire, en terme de répartition de l'énergie dans le plan temps-fréquence.

On voit que, comme l'ERBLet, la ERB-MDCT a une bien meilleure résolution fréquentielle en basses fréquences et temporelle en hautes fréquences, que la MDCT stationnaire. Il est à noter toutefois que, si la ERB-MDCT est peu redondante et donc adaptée au codage audio, elle est cependant moins performante, en terme de résolutions fréquentielle et temporelle, que l'ERBLet.

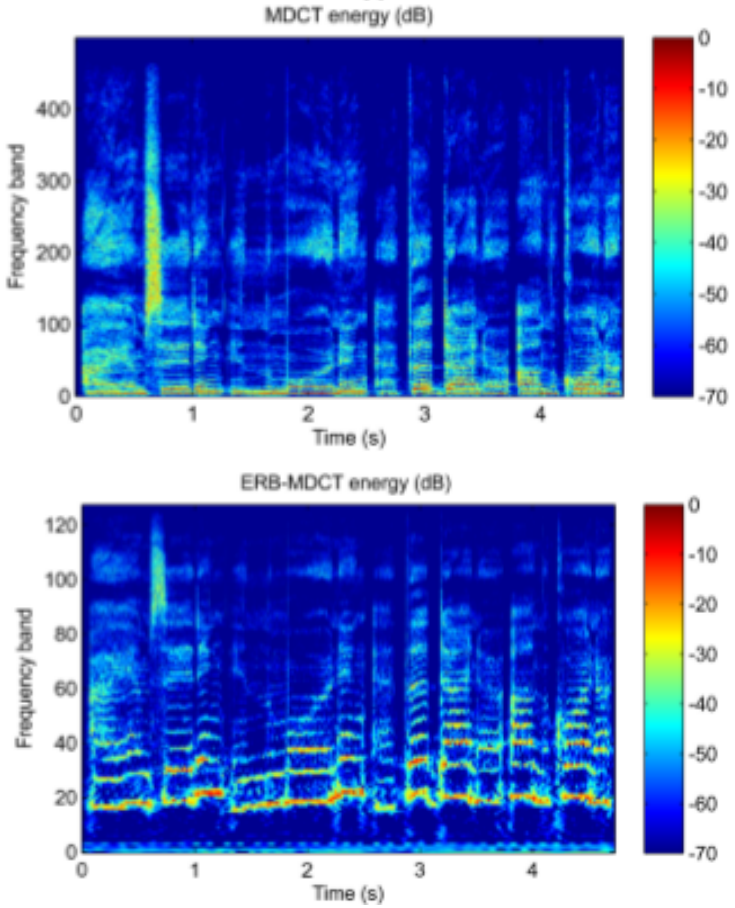



Figure *1.17 – Comparaison des modules (en décibels) d'une MDCT stationnaire (500 bandes fréquentielles d'analyse) et d'une ERB-MDCT (128 bandes) [3].*

Nombre de bandes de la ERB-MDCT

L'échelle ERB standard présente 43 bandes fréquentielles. Cependant, afin d'affiner la précision fréquentielle, on peut définir une transformée dont les bandes sont des sous-multiples des bandes ERB (cf analyse ERB-MDCT de la figure 1.18). Pour définir davantage de bandes, on insère dans les formules (1.4) et (1.5) un paramètre, appelé **v**, qui fixe le nombre de sous-bandes par bande ERB.
N'ayant pas de connaissance a priori quant au nombre minimum de bandes fréquentielles nécessaires pour préserver une bonne qualité perçue, on se propose de comparer différentes configurations. Ainsi, dans la suite de cette étude, sont testées les valeurs v={1, 2, 3}, générant des ERB-MDCT à respectivement {43, 86, 128}) bandes. Ce choix est arbitraire, v n'étant pas nécessairement entier.

# Chapitre 2

# Implémentation du codeur expérimental à ERB-MDCT

## 2.1 Définition des blocs de codage

Comme expliqué au paragraphe 1.3.2.3, il est nécessaire d'utiliser le même nombre de blocs de codage total pour le codeur AAC et le codeur expérimental, afin de s'assurer qu'ils aient à peu près le même débit d'informations annexes.

Calcul du nombre de blocs de codage du codeur AAC

Etant donné que le codeur AAC commute entre deux tailles de fenêtres, cela complexifie le décompte du volume des informations annexes. On fait donc la simplification suivante :
Les fenêtres courtes étant très minoritaires, on peut considérer que le nombre total de facteurs d'échelle est sensiblement le même pour le codeur AAC bloqué en fenêtres longues et pour le codeur AAC qui commute entre deux tailles de fenêtres d'analyse.
Le calcul dans le cas du codeur AAC bloqué en fenêtres longues est alors très simple : il s'agit du nombre de fenêtres d'analyse (dépend de la durée du signal), multiplié par le nombre de blocs de codage en fenêtre longue (soit 49).

Adaptation de l'algorithme de quantification

On remarque que le principe de quantification du nouveau codeur peut être vu comme une version duale de la quantification du codeur AAC : Au lieu de coder séquentiellement des blocs temporels comprenant des sous-bandes de taille variable, on code séquentiellement des blocs fréquentiels comprenant des intervalles temporels de taille variable. Autrement dit, on peut utiliser le même algorithme de quantification que dans AAC, en intervertissant temps et fréquence dans l'indexation des coefficients MDCT.

On a donc une contrainte sur le nombre total de blocs de codage, mais pas sur la répartition dans le plan temps-fréquence. A priori, la répartition en fréquence suit les bandes ERB, mais la répartition temporelle reste à déterminer. On cherche donc la répartition temporelle des blocs qui minimise la distorsion subjective.

Dans le cas du codeur AAC, le découpage en blocs de codage est dicté par les contraintes d'application (codage à la volée par exemple). Dans le nouveau codeur, n'ayant pas ces contraintes, on a a priori une plus grande latitude pour définir ces blocs de codage. Comme il est très difficile d'imaginer des expériences

perceptives concluantes permettant de déterminer le schéma optimal, on a commencé par explorer deux solutions "de bon sens".

**Une logique de répartition des blocs suivant l'échelle ERB**

La première idée a été de répartir les blocs proportionnellement au nombre de coefficients par bande ERB, ce qui revient à avoir un même nombre de coefficients dans tous les blocs de toutes les bandes.

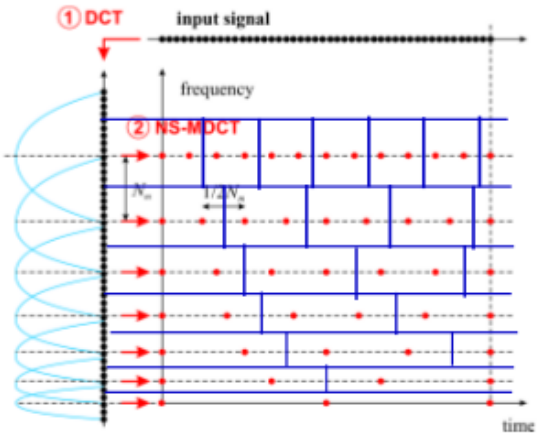



Figure *2.1 – Schéma de répartition des blocs de codage (rectangles bleus) suivant l'échelle ERB.*

**Une logique de répartition temporellement uniforme**

Une seconde idée a été de ne pas tenir compte du nombre de coefficients MDCT sur chaque bande, et donc de répartir les blocs de manière temporellement uniforme.

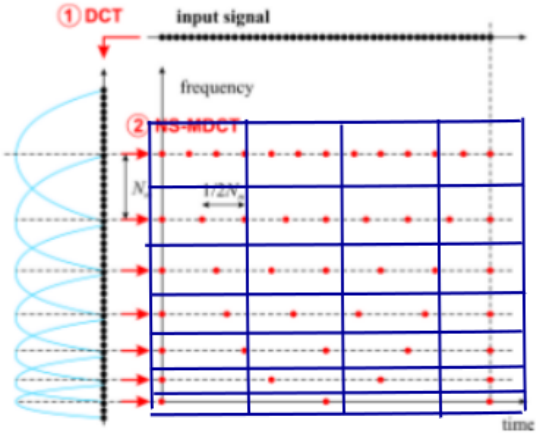



Figure *2.2 – Schéma de répartition temporellement uniforme des blocs de codage.*

Le schéma de la figure 2.2 montre une répartition uniforme théorique. Il n'est cependant pas possible de réaliser une telle configuration en pratique, le nombre de coefficients sur une bande n'étant pas nécessairement un multiple du nombre de blocs souhaités sur cette bande.
En pratique, on fait donc une approximation à un nombre entier de coefficients

par bloc, en ajustant avec le dernier bloc temporel de la bande.

L'algorithme de répartition implémenté (décrit en Annexe B) réalise donc de manière itérative, une répartition "la plus uniforme possible" des blocs de codage, en commençant la répartition sur la bande fréquentielle la plus basse.

Après validation expérimentale, dont la démarche est décrite en début de troisième chapitre, on a opté pour la seconde configuration, à savoir une répartition temporellement uniforme des blocs de codage.

Une fois effectuée la délimitation des blocs de codage, on peut ensuite réaliser l'analyse psychoacoustique et la quantification sur chaque bloc, comme pour le codeur AAC.

## 2.2 Le module psychoacoustique à ERB-MDCT

Lors de la description de l'implémentation du codeur AAC, on a présenté le module psychoacoustique comme une boîte noire permettant d'obtenir, à partir du signal temporel, une valeur de SMR par bloc de codage (cf paragraphe 1.3.2.2).
On a précisé que ce module n'était pas soumis à la norme MPEG et qu'il existait différents modèles, utilisant des transformées temps-fréquence différentes de la transformée utilisée pour le codage.

La principale motivation du codeur expérimental est en fait d'utiliser la même transformée (l'ERB-MDCT) pour le codage et le modèle d'audition, ce qui n'a encore jamais été fait (par faute de transformée adéquate).

Ainsi, dans une partie de l'étude réalisée au cours de ce stage, on utilise un modèle d'audition utilisant la ERB-MDCT, aussi bien pour le codeur expérimental (utilisant déjà l'ERB-MDCT comme transformée de codage), que pour le codeur AAC.
En effet, on cherche à réduire au minimum les différences entre les deux types de codeur, pour garantir la comparaison la plus juste possible.

On présente donc ici l'implémentation de ce modèle d'audition pour les deux codeurs.

**Remarque** : Par souci de clarté, on présente tout d'abord le modèle d'audition du codeur expérimental.
En effet, les coefficients de la transformée ERB-MDCT étant à la fois utilisés pour le codage et pour le modèle psychoacoustique, l'implémentation du modèle est simple et intuitive.
On résume ensuite l'adaptation de ce modèle psychoacoustique au codeur AAC.
Cependant, en réalité, la version du codeur AAC de ce module était déjà implémentée avant ce stage, et le travail réalisé en cours de stage a en fait été, à l'inverse, d'épurer ce module (complexe dans le cas du codeur AAC), et de n'utiliser que les parties utiles au codeur ERB.

### 2.2.1 Cas du codeur expérimental

Le principe de calcul du SMR décrit ici fait appel à deux interpolations des images d'énergie temps-fréquence. Ceci est motivé par le fait qu'on souhaite se ramener à une opération de convolution bidimensionnelle entre l'énergie du signal et un noyau de masquage préalablement défini. Cette convolution implique une grille temps-fréquence rectangulaire, ce qui n'est pas le cas avec la ERB-MDCT.

1 - Les coefficients ERB-MDCT sont donc tout d'abord interpolés, bande par bande, afin d'obtenir une grille temps-fréquence rectangulaire (figure 2.3).

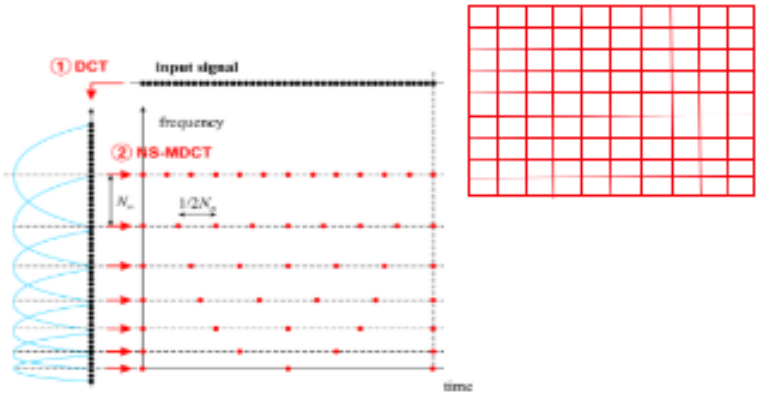



Figure *2.3 – Interpolation des coefficients ERB-MDCT afin d'btenir une grille temps-fréquence rectangulaire.*

2 - On utilise ensuite un programme (développé dans le cadre de la thèse de Thibaut Necciari), qui permet de générer un pattern de masquage temps-fréquence utilisant l'échelle fréquentielle ERB : On spécifie en entrée de ce programme la résolution fréquentielle souhaitée pour le pattern (qu'on va choisir identique à la résolution de la ERB-MDCT utilisée), et on obtient en sortie le pattern sous forme d'une matrice temps-fréquence, dont la taille (résolution temps-fréquence) dépend du nombre de bandes ERB choisi.

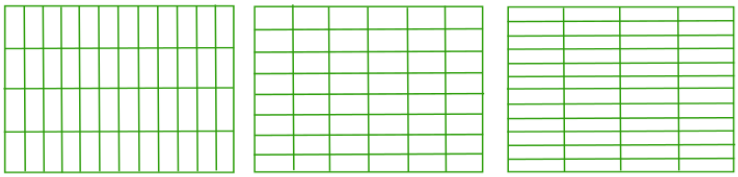


Figure *2.4 – Schéma simplifié représentant des patterns de masquage temps-fréquence suivant l'échelle ERB, à 43 bandes (à gauche), à 86 bandes (au milieu) et à 128 bandes (à droite).*

3 - On calcule ensuite le masque, obtenu par convolution du pattern de masquage avec la grille de coefficients temps-fréquence. En divisant l'énergie du signal par ce masque, on obtient alors une matrice de SMR, décrivant la totalité du plan temps-fréquence.

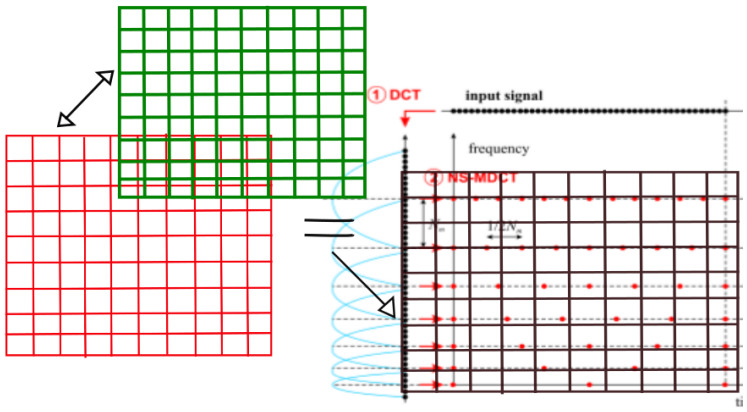



Figure *2.5 – Calcul du masque par convolution des coefficients temps-fréquence (grille rouge) avec le pattern de masquage (grille verte). Le SMR (grille noire) est ensuite obtenu en divisant l'énergie du signal par le masque.*

4 - On réalise enfin une nouvelle interpolation afin d'obtenir une valeur de SMR par bloc de codage (figure 2.6).

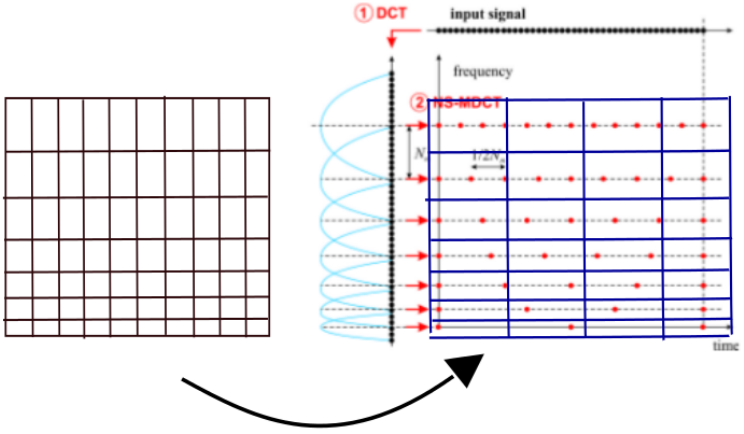



Figure *2.6 – Interpolation des valeurs de SMR sur chaque ligne de la grille afin d'obtenir une valeur de SMR par bloc (rectangles bleus) de codage.*

### 2.2.2 Adaptation au codeur AAC

Comme expliqué en début de paragraphe, l'implémentation de ce module pour le codeur AAC est plus complexe, puisque les transformées de codage (MDCT) et du modèle d'audition (ERB-MDCT) ne sont pas les même.

Sans entrer dans les détails d'implémentation, l'idée de l'adaptation de ce module au codeur AAC, est de réaliser une ERB-MDCT sur un intervalle temporel court centré autour de la fenêtre de MDCT courante.

Toutes les étapes du module qui ont été décrites pour le codeur expérimental, sont donc réalisées, non pas une seule fois, mais sur chaque fenêtre d'analyse, ce qui démultiplie les étapes de calcul.

# Chapitre 3

# Démarche comparative et résultats

## 3.1 Etude préliminaire avec modèle psychoacoustique simplifié

On a tout d'abord comparé les codeurs AAC et ERB-MDCT en terme de compromis débit/distorsion (ou plutôt entropie/distorsion), avec un modèle psychoacoustique simplifié. En effet, il aurait été inutile d'implémenter un modèle psychoacoustique classique, et plus complexe (comme celui présenté précédemment utilisant l'ERB-MDCT), sans s'assurer au préalable d'avoir des résultats cohérents avec un modèle minimal. On a donc tout d'abord défini un SMR constant sur tout le plan temps-fréquence.

Un tel modèle étant cependant trop simpliste pour véritablement évaluer la distorsion perçue, on essaie surtout ici d'évaluer la durée du pré-écho, qui constitue une des manifestations les plus facilement identifiables de la distorsion subjective apportée par un codeur compressif.
Cependant, il est important de préciser que la dégradation perçue n'est pas proportionnelle à la durée du pré-écho : en dessous d'un certain seuil, on n'entend aucune distorsion (c'est le phénomène de masquage temporel "backward"). Les études réalisées dans la littérature sur le masquage "backward" (surtout sur des bruits blancs), montrent que ce seuil semble par ailleurs varier en fonction des composantes fréquentielles contenues dans le signal. Les quelques études trouvées dans la littérature ne permettent cependant pas de tirer davantage de conclusions sur la perception du pré-écho dans notre cas précis.
Des tests perceptifs sont donc menés en fin de chapitre.

Avant de se lancer dans le travail comparatif qui est l'objectif de ce stage, on commence, dans cette section à modèle psychoacoustique simplifié, par vérifier le bon fonctionnement des différents codeurs et valider expérimentalement certaines hypothèses et certains choix qui ont été énoncés dans les chapitres précédents.

### 3.1.1 Définition du SMR constant

Dans cette partie, on fixe un SMR constant : l'écart entre le spectre et le seuil de masquage est constant et identique sur tous les blocs de codage.

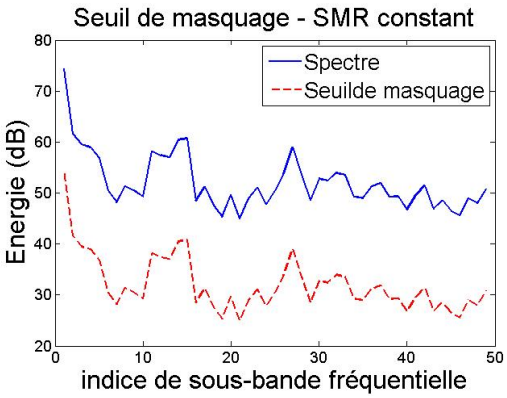



Figure *3.1 – SMR constant à 20dB SPL.*

Cette simplification est un modèle psychoacoustique acceptable. En effet, la grande différence avec un seuil de masquage réel concerne les hautes fréquences, où le seuil de masquage remonte et passe au dessus du spectre, ce qui n'est pas gênant puisque par conséquent la partie haute fréquence du spectre est plus difficilement perceptible.

### 3.1.2 Validation du fonctionnement des codeurs

#### 3.1.2.1 Validation du module de quantification

Dans le cadre de cette étude, on utilise le module de quantification à débit variable, qui assure de ramener la puissance de l'erreur de quantification au plus près du seuil de masquage, par valeur inférieure (cf paragraphe 1.2.3).

Ce fonctionnement à débit variable va permettre de vérifier que le module de quantification est bien utilisé.

En effet, le bruit de quantification étant ramené au plus près du seuil de masquage, l'écart, en sortie de module, entre le spectre et le bruit de quantification, appelé "Rapport Signal à Bruit" (Signal-to-Noise Ratio ou SNR), est supposé être a peu près égal à l'écart entre le spectre et le seuil de masquage (SMR).

On a donc tracé, pour les codeurs AAC et ERB, la valeur du SNR sur chaque bloc de codage, pour une consigne de SMR de 20 dB (figures 3.2 et 3.3).

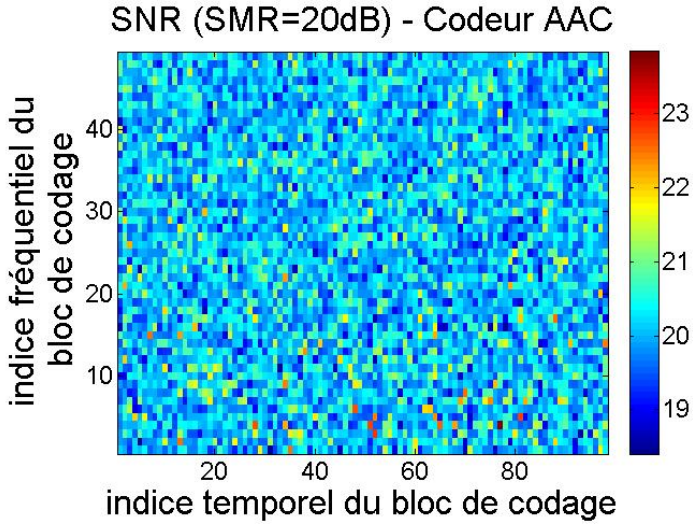



Figure *3.2 – Codeur AAC. Valeur du SNR effectif sur chaque bloc de codage, pour un SMR cible de 20dB.*

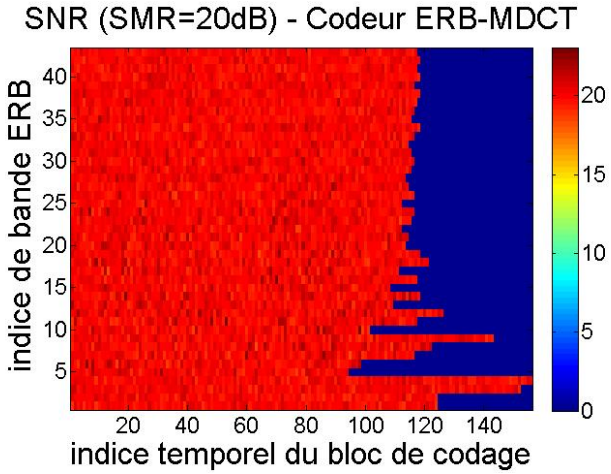



Figure *3.3 – Codeur expérimental. Valeur du SNR effectif sur chaque bloc de codage, pour un SMR cible de 20dB. Remarque : Le nombre de blocs de codage, et par conséquent de valeurs de SNR (en rouge), par bande ERB, n'est en pratique pas tout à fait uniforme (cf explications paragraphe 2.1). On a donc, pour des raisons pratiques de tracé bidimensionnel sous Matlab nécessitant une matrice de valeurs, complété la grille de SNR par des zéros (zone bleue foncée, qui ne correspond donc pas à des valeurs de SNR).*

On constate, qu'à l'exception de quelques blocs de codage en basses fréquences pour le codeur AAC, le SNR effectif correspond au SMR cible à +/- 2dB pour les deux codeurs.

#### 3.1.2.2 La valeur du SMR comme contrôle approximatif du taux de dégradation

Puisqu'on utilise une quantification à débit variable, on ne peut pas réellement imposer un débit rigoureusement identique pour les deux codeurs qu'on veut comparer en terme de débit/distorsion, mais on va néanmoins pouvoir simuler différents taux de dégradation.

En effet, avec ce modèle simplifié, la distorsion subjective est supposée varier dans le sens inverse du SMR.

Afin de mieux cerner cette idée, on décrit les deux cas de figure extrêmes :

Cas d'un SMR très élevé (exemple figure 3.1)
Le seuil de masquage est situé très en dessous du spectre. L'algorithme d'optimisation de la quantification à débit variable ramène le bruit de quantification en dessous du seuil de masquage. Le bruit de quantification, d'un faible niveau, ne devrait alors pas être perçu.

Cas d'un SMR proche de 0
Le seuil de masquage se situe presque au même niveau que le spectre. Le bruit de quantification, ramené au seuil de masquage, a dans ce cas un niveau très élevé, et devrait être fortement perçu.
En réalité, ce n'est pas exactement vrai, puisque sur une sous-bande masquée (en très hautes fréquences par exemple), le bruit n'est pas perçu.

Ainsi, en faisant varier le SMR de 0 à 20dB par exemple, on peut simuler, en première approximation, différents niveaux de dégradation.

#### 3.1.2.3 Validation de la correspondance débit/entropie

On justifie dans ce paragraphe, l'utilisation du compromis entropie/distorsion comme moyen d'évaluation des codeurs, à la place du compromis débit/distorsion. Dans un premier temps, on observe les liens entre débit et entropie pour le codeur AAC, puis, on vérifie qu'une même consigne de SMR pour le codeur AAC et le codeur ERB mènent à des valeurs d'entropie très proches.

Correspondance débit/entropie sur le codeur AAC
On a d'abord étudié, sur le codeur AAC, les correspondances entre débit et entropie, pour différents niveaux de dégradations subjectifs, grâce à différentes consignes de SMR (cf explications précédentes).
On a donc tracé les différents types de débits (débit caractérisant l'information MDCT, débit correspondant à l'information annexe, débit total) et l'entropie, ramenée en kbits/s, en fonction du SMR, pour un SMR variant de 0 à 20dB.

Ce travail a été réalisé sur différents exemples sonores, qui ont tous présenté les même correspondances entre débit et entropie (cf exemple figure 3.4).

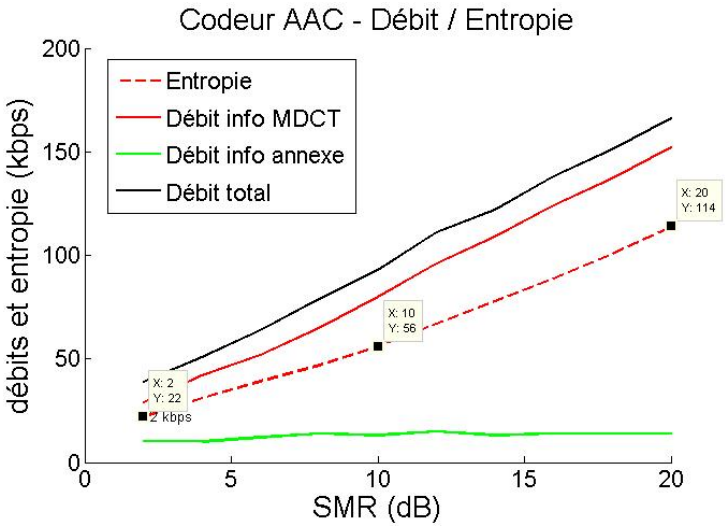



Figure *3.4 – Son de castagnettes. Courbes représentant les différents débits (Informations annexes , information MDCT) et l'entropie, en fonction du SMR.*

On constate qu'entropie et débit MDCT sont approximativement liés par une loi affine.
Par ailleurs, on note que le débit correspondant à l'information annexe est à peu près constant, ce qui confirme que l'information annexe est à peu près proportionnelle au nombre de facteurs d'échelle, imposé par la norme MPEG.

Comparaison des codeurs AAC/ERB en terme de correspondance SMR/entropie

Lorsqu'on calcule l'entropie en fonction du SMR pour le codeur ERB (figure 3.5), on obtient des résultats très proches du codeur AAC, sur tous les exemples sonores testés.
On rappelle cependant que le SMR constant n'est qu'une indication grossière de la distorsion perçue, et que la correspondance entropie/SMR ne permet en aucun cas de comparer réellement les codeurs en terme de compromis entropie/distorsion.

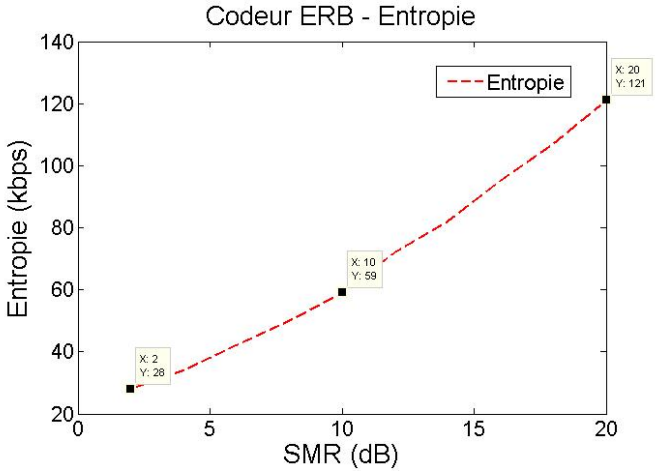



Figure *3.5 – Son de castagnettes. Courbes représentant l'entropie, en fonction du SMR.*

Ces résultats peremttent donc de valider l'utilisation de l'entropie, ramenée en kbits/s (kbps), à la place du débit.

#### 3.1.2.4 Validation du choix de répartition des blocs de codage du codeur expérimental

Maintenant que le modèle à SMR constant a été expliqué, et qu'on a montré que l'entropie est un bon indicateur de débit réel, on revient sur le choix de répartition des blocs de codage du codeur ERB, qui a été discutée paragraphe 2.1, et qui est ici validé expérimentalement avec ce modèle à SMR constant.

On a comparé les deux configurations décrites au paragraphe 2.1, sur différents exemples sonores, en fixant un SMR cible de 10 dB, qui est une consigne de SMR permettant d'obtenir une distorsion suffisante sur les deux codeurs, afin de pouvoir établir une comparaison perceptive significative.
On précise qu'il ne s'agissait que de tests d'écoute informels, par manque de temps.
En terme de qualité globale, le second cas de figure semblait légèrement meilleur que le premier, mais sur certains signaux, il était toutefois difficile de déterminer quelle configuration présentait le plus de distorsion.

N'ayant pas une validation expérimentale rigoureuse d'une des deux configurations, on a mis en place une expérience qui a permis d'orienter notre choix final. On a voulu déterminer, pour une bande fréquentielle donnée, le nombre de facteurs d'échelle à répartir à partir duquel, la qualité perceptive du son codé ne s'améliorait plus.

Pour différentes bandes basses, moyennes et hautes fréquences, on a donc réalisé l'expérience suivante : On ne quantifie que cette bande fréquentielle de coefficients MDCT (figure 3.6), les autres bandes n'étant pas codées (elles n'apportent donc aucune distorsion). Le seul élément apportant de la distorsion au signal resynthétisé est donc la quantification de cette unique bande fréquentielle. On a commencé par quantifier cette bande avec un unique bloc, puis on a augmenté le nombre de blocs au fur et à mesure, jusqu'à ce qu'aucune amélioration de la qualité sonore ne soit perçue. A nouveau, il ne s'agit, par manque de temps, que d'une expérience informelle.

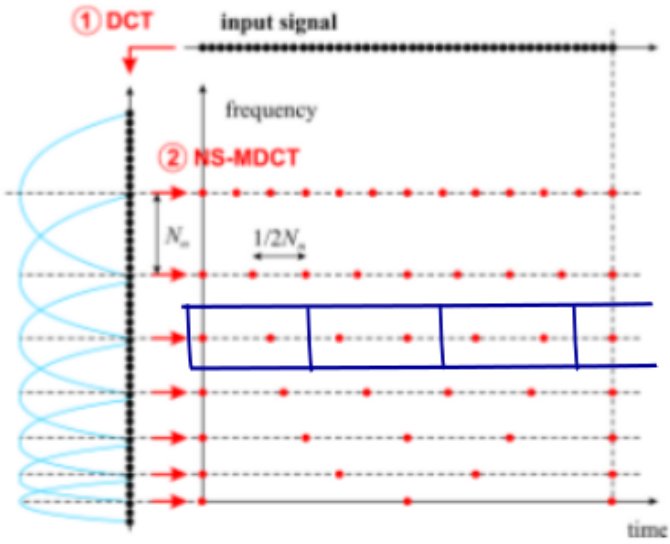



Figure *3.6 – Schéma représentant la quantification d'une seule bande fréquentielle.*

On constate les comportements suivants : Que ce soit en basses, en moyennes ou en hautes fréquences, certaines bandes ne génèrent pas de distorsion, même quantifiées avec un unique facteur d'échelle. A l'inverse, certaines bandes présentent de la distorsion, même avec un grand nombre de facteurs d'échelle.
D'une manière générale, il ressort de la diversité des cas de figures testés (différentes bandes quantifiées sur différents exemples sonores), qu'il est très difficile de mette en évidence de manière conclusive la répartition optimale des blocs de codage selon l'axe temporel.

On a donc opté pour la répartition temporellement uniforme des blocs de codage (second cas de figure) qui nous a semblée meilleure en terme de qualité sonore globale.

### 3.1.3 Etude comparative des codeurs

Comme expliqué en introduction de ce chapitre, on compare, dans cette partie à modèle psychoacoustique simplifié, la durée de pré-écho que présentent les codeurs, à entropies égales (ou très proches).
On rappelle en effet qu'on ne peut pas véritablement fixer des entropies identiques pour nos codeurs, mais qu'en fixant des SMR identiques, on obtient des entropies à peu près égales (cf figures 3.4 et 3.5).

#### 3.1.3.1 Paramètres de l'étude

Nombre de bandes fréquentielles ERB et pré-écho

Dans le cas d'une transformée TF classique (MDCT), la durée du pré-écho est au plus égale à la durée de la fenêtre d'analyse qui caractérise la transformée (taille des blocs temporels).

Dans le cas de la ERB-MDCT, ce n'est pas si simple, puisqu'il n'y a pas de fenêtre d'analyse unique dans le domaine temporel. On peut uniquement affirmer que, plus la résolution fréquentielle est fine, plus la résolution temporelle est mauvaise, et plus la durée de pré-écho devrait augmenter. Ainsi, on suppose que la durée de pré-écho devrait varier inversement avec la fréquence. Par conséquent, plus l'ERB-MDCT a de bandes d'analyse fréquentielle, plus elle devrait générer de pré-écho.

On va donc comparer le codeur AAC qui commute entre deux tailles de fenêtres (qu'on appelle dans toute la suite **Codeur AAC switch**) avec différentes versions du codeur ERB, présentant plus ou moins de bandes fréquentielles ERB :
- Un codeur ERB à 43 bandes (1 bande par ERB), qu'on appelle **Codeur ERB v1**
- Un codeur ERB à 86 bandes (2 bandes par ERB), qu'on appelle **Codeur ERB v2**
- Un codeur ERB à 128 bandes (3 bandes par ERB), qu'on appelle **Codeur ERB v3**

On va également dégrader nos sons avec une version du codeur AAC qui n'utilise que des fenêtres longues (qu'on appelle **Codeur AAC long**), afin d'avoir une référence de codeur générant beaucoup de pré-écho.

L'étude qui suit compare donc ces **cinq** codeurs.

Exemples sonores choisis

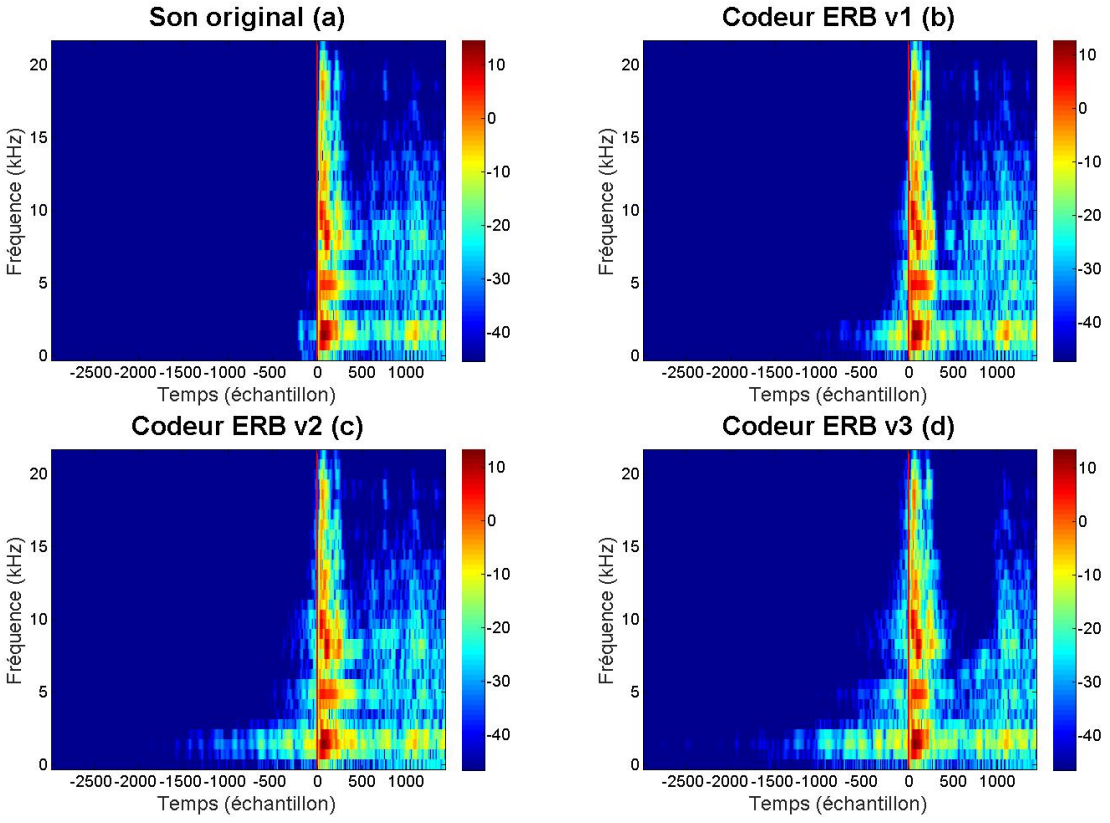
Afin d'analyser le pré-écho généré par les cinq codeurs décrits ci-dessus, on a choisi de dégrader des sons présentant des attaques très marquées (signaux à fort caractère impulsionnel).
Ci-dessous, sont donc décrits les résultats obtenus sur un son de castagnettes, ce son étant particulièrement adapté à l'étude du pré-écho.

Niveaux de dégradations

On a fixé un niveau de SMR de 10dB (seuil de masquage situé 10dB en dessous du spectre, soit relativement élevé), afin que les sons soient suffisamment dégradés pour qu'on puisse différencier les codeurs et établir une comparaison.

### 3.1.3.2 Résultats

On a tout d'abord effectué une analyse temps-fréquence de la durée de pré-écho généré par les différents codeurs, en traçant les spectrogrammes des sons dégradés, et en zoomant sur les transitoires (figure 3.7).



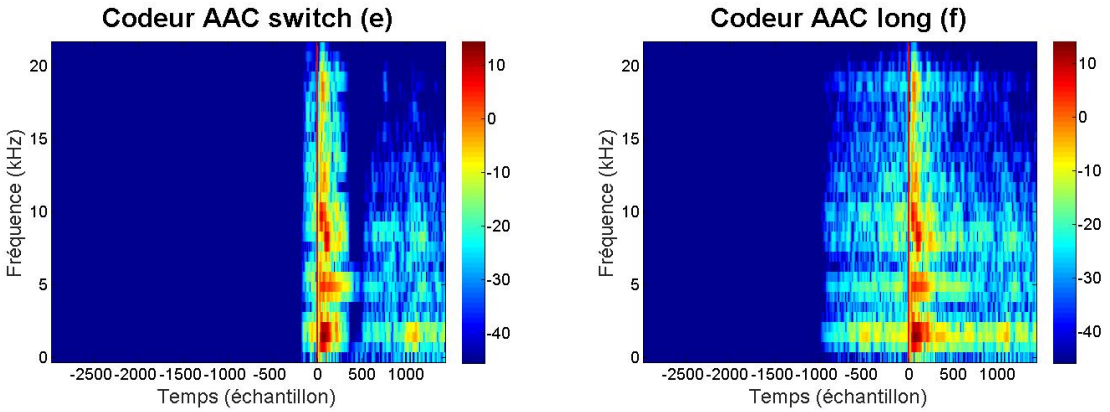



Figure *3.7 – Son de castagnettes. Son original (a), son dégradé par codeur ERB v1 (b), par codeur ERB v2 (c), par codeur ERB v3 (d), par codeur AAC switch (e), et par codeur AAC long (f)*

Ces spectrogrammes montrent bien des résultats très différents obtenus avec les codeurs AAC et ERB :
- Pour le codeur AAC, on remarque une répartition uniforme en fréquence du pré-écho, due à la grille d'analyse uniforme de ce codeur.
On constate que le codeur AAC qui commute entre deux tailles de fenêtre permet effectivement de réduire la durée du pré-echo de manière conséquente par rapport à un codeur AAC fonctionnant uniquement avec des fenêtres d'analyse longues.
- Pour le codeur ERB, les résultats suivent les prédictions : plus la résolution fréquentielle est fine (bandes basses fréquences), plus la durée de pré-écho est grande.
Ainsi, le codeur ERB v1 (résolution fréquentielle la moins bonne), présente bien un pré-écho plus court que le codeur ERB v2, lui même meilleur que le codeur ERB v3.

### Importance de la position de la fenêtre d'analyse pour le codeur AAC

Lors d'études de pré-écho similaires sur différents exemples sonores, on a remarqué que la durée du pré-écho généré par le codeur AAC (switch ou long), était variable.
Cela s'explique par le fait que la durée du pré-écho est majorée par la durée de la fenêtre. Les variations locales dépendent de la position de l'attaque.

Pour vérifier cet aspect, on a tracé de nouveau les spectrogrammes précédents concernant le codeur AAC, avec cette fois-ci différentes positions de la fenêtre d'analyse (figures 3.9 et 3.10) :
- Position 1 : L'attaque se situe au milieu d'une fenêtre (ou à l'extrémité de la fenêtre d'analyse suivante).
- Position 2 : L'attaque se situe à 1/4 (resp. 3/4 par symétrie) d'une fenêtre d'analyse.

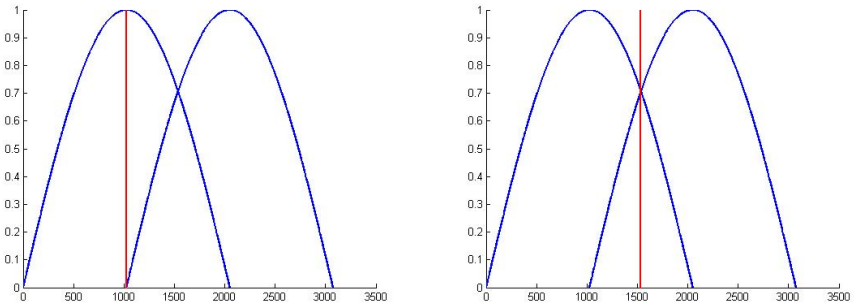


Figure *3.8 – Position des fenêtres d'analyse (en bleu) par rapport à l'attaque (en rouge). L'attaque se situe au milieu ou aux extrémités de la fenêtre (à gauche), l'attaque se situe à 1/4 ou aux 3/4 de la fenêtre (à droite).*

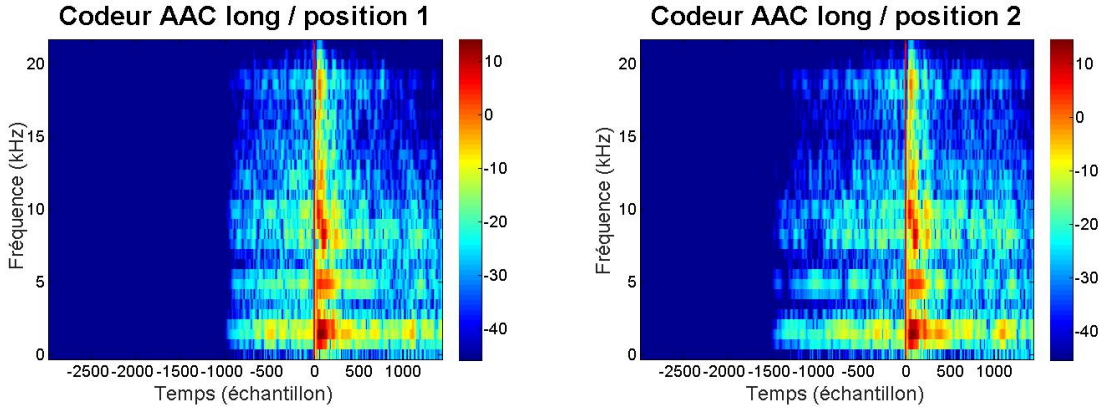



Figure *3.9 – Attaque d'un son de castagnettes dégradé par le codeur AAC long. A gauche la fenêtre d'analyse est en position 1 par rapport à l'attaque, à droite en position 2.*

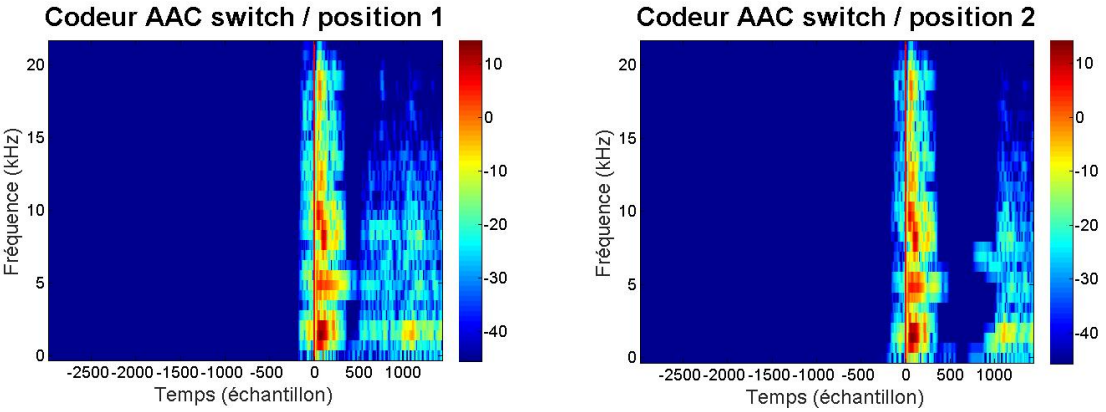



Figure *3.10 – Attaque d'un son de castagnettes dégradé par le codeur AAC switch. A gauche la fenêtre d'analyse est en position 1 par rapport à l'attaque, à droite en position 2.*

On a effectué cette vérification sur différents exemples sonores.
On peut certes observer des différences au niveau des spectrogrammes lorsqu'on se place dans ces deux configurations opposées d'analyse, mais sur tous les exemples testés, ces différences ne modifient pas la hiérarchie des codeurs en terme de durée du pré-écho. La position des fenêtres d'analyse du codeur AAC par rapport au signal n'étant pas significative dans le cadre de cette étude, cet aspect est laissé de côté dans la suite de ce rapport.

Conclusion de la comparaison AAC/ERB

Au vu des spectrogrammes, le codeur ERB v1 semble quantitativement proche du codeur AAC switch en terme de durée de pré-écho, même si, comme on vient de le voir, la répartition fréquentielle du pré-écho n'est pas la même.

Cependant, perceptivement, il semblerait que le codeur ERB v2 soit celui qui s'apparente le plus au codeur AAC switch : Ils semblent très similaires, tant en terme de pré-écho que de qualité globale.

On peut, par contre, écarter définitivement un codeur ERB à 128 bandes (v3), qui présente perceptivement, beaucoup plus de pré-écho que les autres codeurs.

## 3.2 Etude avec modèle psychoacoustique à ERB-MDCT

Dans cette partie, on introduit dans les codeurs AAC et ERB le modèle psychoacoustique utilisant l'ERB-MDCT qui a été décrit au chapitre précédent.

### 3.2.1 Validation du fonctionnement des codeurs.

On a de nouveau vérifié que le module de quantification fonctionnait correctement pour les deux codeurs, en calculant cette fois-ci, la différence entre SMR et SNR sur tous les blocs, le SMR n'étant plus constant. Comme pour l'étude à SMR constant, le SNR correspond au SMR à environ +/- 2dB.

Il est également nécessaire d'étudier de nouveau le comportement de l'entropie des codeurs AAC et ERB avec modèle psychoacoustique, afin de vérifier qu'on peut toujours comparer les codeurs en terme de compromis entropie/distorsion.
Après avoir codé, avec les cinq codeurs précédents, différents exemples sonores, on constate que les entropies des sons dégradées ne sont pas tout à fait égales et présentent des écarts allant de 10 à 20 kbps.
Plus précisément : Les entropies des codeurs ERB v1 et v3 sont les plus éloignées, les valeurs des entropies des codeurs ERB v2, AAC switch et AAC long se situant entre les deux et étant très proches l'une de l'autre.
Cette hiérarchie montre que cette différence d'entropie n'est en tout cas pas due aux éventuelles différences d'implémentation des modules psychoacoustiques du codeur ERB et du codeur AAC.

Il est toutefois nécessaire, si on veut pouvoir comparer les codeurs en terme de compromis entropie/distorsion, de pouvoir générer des sons à entropies très proches.
On rappelle que les codeurs n'étant pas munis de vrai module de codage entropique, on ne peut pas donner directement une consigne de débit. Il est donc nécessaire de pouvoir ajuster autrement l'entropie relative des codeurs.

Pour cela, on translate vers le haut ou le bas, le seuil de masquage de certains codeurs (comme lorsqu'on faisait varier la valeur du SMR lors de l'étude à SMR constant). Autrement dit, on ajoute ou on retranche une certaine constante au SMR (en dB) fourni par le modèle psychoacoustique.

Ainsi, on a choisi le codeur AAC switch comme référence d'entropie, et on a implémenté, dans le codeur ERB, un algorithme ajustant le seuil de masquage jusqu'à ce que l'entropie résultante soit très proche de celle du codeur AAC.

### 3.2.2 Etude comparative des codeurs

#### 3.2.2.1 Paramètres de l'étude

Niveaux de dégradations

Une fois les cinq codeurs ajustés en entropie (paragraphe précédent), on simule un fonctionnement à plus bas débit en relevant les seuils de masquage, d'un même offset, pour les cinq codeurs (figure 3.11).

On a donc relevé progressivement le seuil de masquage (avec un pas de quelques dB), jusqu'à ce que la distorsion perçue après codage des sons soit suffisante pour comparer perceptivement les cinq codeurs en terme de durée de pré-écho.

Par exemple, pour le son de castagnettes qu'on analyse de nouveau dans cette partie, le niveau de distorsion perçue est suffisant à partir d'un seuil de masquage relevé d'environ 8dB.
A partir de 10-12dB, d'autres artefacts apparaissent, qui sont susceptibles de gêner la comparaison perceptive des codeurs.

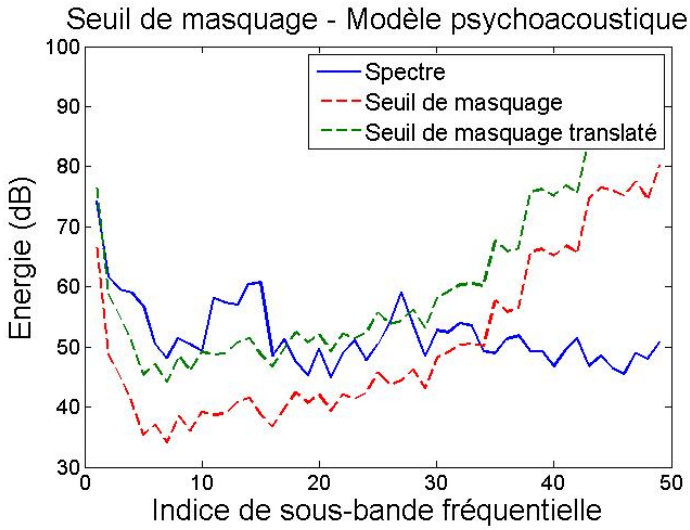



Figure *3.11 – Codeur AAC. Translation du seuil de masquage vers le haut de 8dB sur tous les blocs de codage de la fenêtre d'analyse.*

#### 3.2.2.2 Résultats

Comme pour l'étude à SMR constant, on a effectué une analyse temps-fréquence de la durée du pré-écho, en traçant les spectrogrammes d'un son de castagnettes dégradé par les cinq codeurs (figure 3.12).

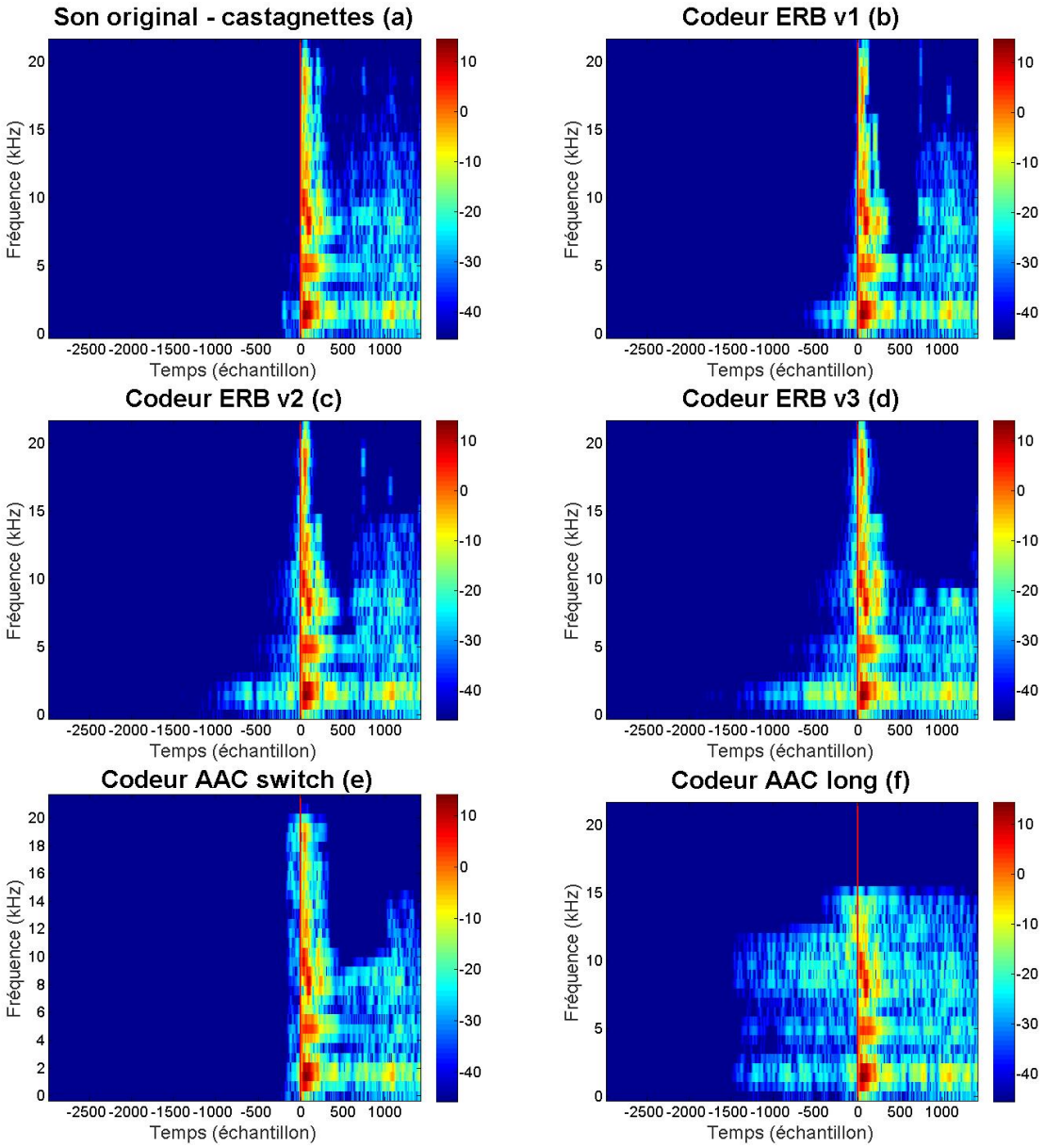



Figure *3.12 – Son de castagnettes. Son original (a), son dégradé par codeur ERB v1 (b), par codeur ERB v2 (c), par codeur ERB v3 (d), par codeur AAC switch (e), et par codeur AAC long (f).*

On constate que l'implémentation, au sein des deux codeurs, du module psychoacoustique utilisant la ERB-MDCT, ne change pas la hiérarchie des cinq codeurs en terme de durée de pré-écho.
En effet, au vu des spectrogrammes, et, après avoir écouté les sons dégradés, on aboutit aux même conclusions que pour l'étude à SMR constant : le codeur ERB semble pouvoir rivaliser avec le codeur AAC en terme de qualité globale et de pré-écho, pour un nombre de sous-bandes par ERB se situant entre 1 et 2.

Remarque : Afin d'être le plus rigoureux possible, on a ajusté, comme expliqué précédemment, les entropies des cinq codeurs. Cependant, même sans ajustement préalable, les spectrogrammes sont très similaires à ceux tracés ci-dessus et la hiérarchie des codeurs reste la même.

Comme mentionné en introduction, l'évaluation de la durée du pré-écho seule ne suffit pas à classer les codeurs perceptivement.

Il a donc été nécessaire de réaliser des tests perceptifs afin d'approfondir ces résultats.

### 3.2.3 Tests perceptifs

Afin de valider les résultats précédents sur la durée de pré-écho et la qualité globale des différents codeurs comparés, deux tests perceptifs ont été menés.

Cette section présente leur mise en place et leur protocole, mais les résultats de ces tests seront présentés lors de la présentation orale, certains tests étant toujours en cours.

#### 3.2.3.1 Evaluation du pré-écho perçu

Un premier test a été élaboré pour comparer les codeurs AAC switch, ERB v1, ERB v2 et ERB v3, en terme de durée perçue de pré-echo.

On a choisi des exemples sonores à caractère très impulsionnel : le son de castagnettes étudié précédemment, ainsi qu'un son de glockenspiel (similaire à un xylophone), et un son de batterie (à caractère légèrement moins impulsionnel, mais présentant des attaques dans différentes plages fréquentielles).
On a dégradé ces sons, à l'aide de ces quatre codeurs, à un niveau de dégradation suffisant pour pouvoir comparer perceptivement le pré-écho. On a donc, pour tous les codeurs, relevé le seuil de masquage de 8 dB par rapport à la sortie du modèle.

Toutefois, à ce niveau de dégradation, d'autres distorsions que le pré-écho apparaissent également. Or, on souhaite que les sujets ne comparent les codeurs qu'en terme de durée de pré-écho perçu (point sur lequel on a insisté auprès des sujets).
Afin de familiariser le sujet avec la perception du pré-écho, spécifiquement sur les sons qu'il est amené à évaluer, on a inclu une phase d'apprentissage.

Pour cette phase d'apprentissage, on fait écouter les sons de référence des trois exemples sonores mentionnés ci-dessus, ainsi que des versions de ces sons où on a ajouté artificiellement et progressivement du pré-écho (figure 3.13).

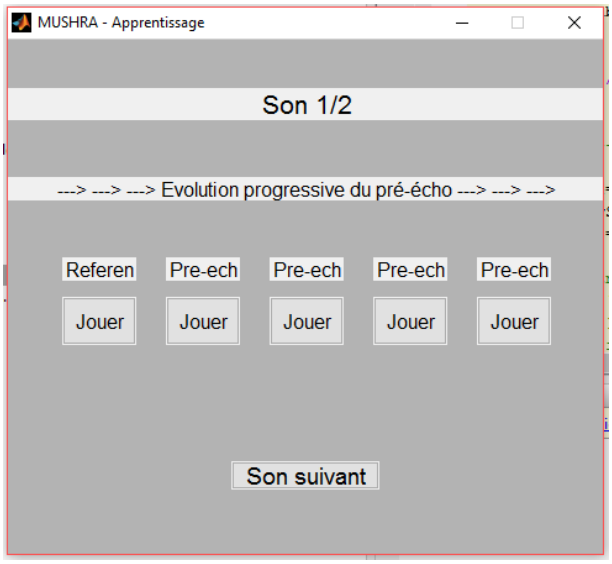



Figure *3.13 – Visuel de l'interface de la phase d'apprentissage. Le sujet peut écouter, autant de fois qu'il le souhaite, le son référence, ainsi que des versions successives de ce son où il va percevoir des durées de pré-écho de plus en plus élevées.*

Méthode d'ajout de différentes durées de pré-écho

On a utilisé un code, déjà implémenté, qui réalise une FFT sur tout le signal sonore. Dès qu'un transitoire est détecté dans la fenêtre d'analyse de la FFT (grâce à une fonction de détection d'attaque), le programme ajoute un bruit de phase sur toute la fenêtre d'analyse, avant de resynthétiser le signal temporel (FFT inverse).
L'ajout de ce bruit de phase a pour conséquence, lors de la resynthèse, de présenter du pré-écho sur la fenêtre temporelle qui chevauche l'attaque.
Ainsi, en choisissant une FFT avec des fenêtres d'analyse plus ou moins grandes, on peut faire varier la durée du pré-écho généré.

Une fois la phase d'apprentissage réalisée sur les trois exemples sonores, le sujet est amené à évaluer les quatre codeurs (nommés A, B, C et D), en terme de durée de pré-écho, sur chaque exemple sonore, en attribuant une note entre 0 et 100 à chaque codeur.
L'ordre des codeurs sur l'interface de test (présentée figure 3.14), est par contre aléatoire pour chaque exemple sonore.

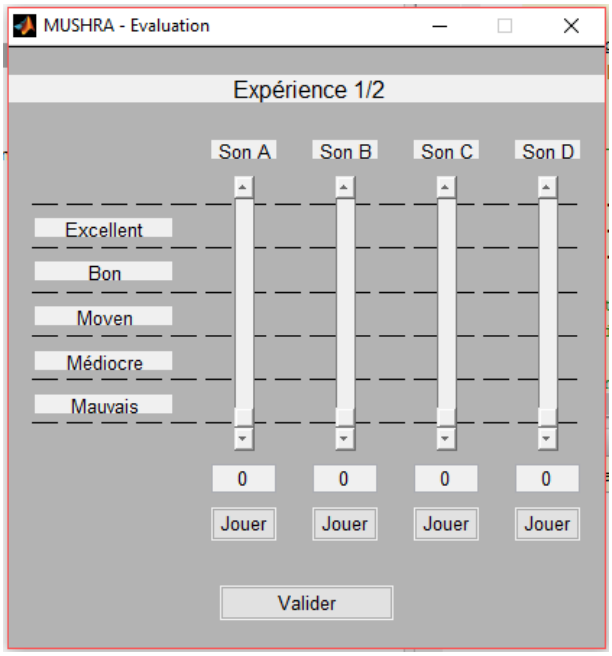



Figure *3.14 – Visuel de l'interface d'évaluation des codeurs.*

#### 3.2.3.2 Evaluation de la qualité globale

Un second test a été mis en place, effectué sur le même panel de sujets que pour le test sur le pré-écho, avec des exemples sonores plus variés (mêlant divers instruments et sons de voix chantée), afin de comparer les codeurs précédents en terme de qualité globale.

L'interface d'évaluation des codeurs est la même que pour le test sur le pré-écho, seule la consigne change : les sujets doivent comparer les sons perçus en terme de qualité globale, tous artefacts confondus. Ce test ne nécessite pas de phase d'apprentissage.

# Conclusion générale et perspectives

Cette première étude, assez générale, d'un codeur utilisant la transformée ERB-MDCT est loin d'être décevante : un tel codeur, moyennant un nombre de bandes d'analyse optimisé, semble présenter des capacités de codage (compromis entropie/distorsion) proches de celles du codeur MPEG AAC.

Les résultats obtenus sont encourageants et des études plus approfondies mériteraient d'être menées, sur plusieurs aspects. Il est tout d'abord nécessaire de calibrer le module psychoacoustique à ERB-MDCT du codeur expérimental, dont certains paramètres ont été repris sans modification depuis le module du codeur AAC.
Il est également primordial d'insérer un véritable module de codage entropique, afin de pouvoir comparer les codeurs précisément en terme de débit/distorsion.
Une fois ces modules optimisés, on pourrait étudier des valeurs intermédiaires du nombre de sous-bandes d'analyses de la ERB-MDCT.
Il serait enfin intéressant de s'appuyer sur davantage de connaissances psychoacoustiques, afin notamment de mettre en place des protocoles de tests perceptifs plus pertinents pour valider les différentes étapes du processus d'amélioration du codeur expérimental proposé ci-dessus.

# Bibliographie


[1] A.Spanias et coll., "Signal processing and audio coding", WILEY-INTERSCIENCE.

[2] O.Derrien et coll., "Le codeur MPEG-2 AAC expliqué aux traiteurs de signaux", 18 Juin 2000.

[3] O.Derrien et coll., "A quasi-orthogonal, invertible, and perceptually relevant time-frequency transform for audio coding", EUSIPCO, 2015.

[4] T.Necciari et coll., "The ERBlet transform : An auditory-based time-frequency representation with perfect reconstruction", ICASSP IEEE, 2013, 498–502.

[5] O.Derrien, "An introduction to non-stationnary time-frequency bases", POTION Kick-off Meeting, Avril 2014.

[6] J-L. Durrieu, "Codage audio et normes", Cours à Télécom ParisTech, Avril 2008.

[7] T.Necciari, "A perfectly invertible and perceptually motivated time-frequency transform for audio representation, analysis and synthesis", ESI12 Workshop, Décembre 2012.

[8] O.Derrien, "Introduction to non-stationnary MDCT", *http ://potion.cnrs-mrs.fr/Documents/Intro_NS_MDCT.pdf* , Janvier 2014.

[9] Y.Wang et coll., "Modified Discrete Cosine Transform - Its implications for audio coding and error cancealment".

[10] P.Balazs et coll., "Theory, implementation and applications of nonstationnary gabor frames".

[11] N.Moreau, "Techniques de compression des signaux", MASSON, 1994.

# Annexes

## Annexe A : Démarche ingénieur

Bien que ce stage constitue un travail de recherche, il s'inscrit dans un domaine à applications variées. En effet, les codecs audio sont utilisés entre autre, pour le stockage de sons, la diffusion en direct (*Streaming* en anglais).

**Transdisciplinarité**

La conception d'un codeur nécessite une approche pluridisciplinaire. En effet, l'amélioration conséquente du compromis débit de données/ qualité sonore qui a été apportée dans les codecs actuels sont le fruit d'études poussées menées, tant en traitement du signal, qu'en psychoacoustique. L'informatique théorique a également son rôle à jouer en terme d'optimisation du flux de données, via le module de codage binaire entropique. Concernant ce stage plus particulièrement, s'il n'a pas été implémenté de module de codage entropique, un travail réfléchi d'implémentation a néanmoins été réalisé : comprendre les logiques d'implémentation des différents modules du codeur AAC, pour les adapter le plus efficacement possible au codeur expérimental.

**Innovations/Limites de fonctionnement**

Ce stage est innovant en terme de recherche, car on a proposé un codeur utilisant une unique transformée temps-fréquence (la ERB-MDCT), à la fois pour le codage et pour le modèle psychoacoustique.
Cependant, dans le cadre de ce travail de recherche, plusieurs simplifications ont été apportées, et de nombreuses hypothèses de travail ont été émises.
L'absence d'un module de codage entropique par exemple, seul véritable outil de calcul du débit réel, ainsi que le traitement du signal sonore en un seul macrobloc, sont des exemples de simplifications et d'hypothèses qui ne permettent pas de placer le codeur ERB-MDCT dans un contexte applicatif pour le moment, et de connaître sa véritable efficacité pour une application donnée.

# Annexe B : Algorithme de définition des blocs de codage du codeur expérimental

Constantes

$F_{AAC}$ = nombre total de facteurs d'échelle à répartir, imposé par le codeur AAC

Nbandes = Nombre de bandes ERB

Variables

$f_{disponibles}$ = nombre de facteurs d'échelle restants, à répartir sur la bande en cours d'itération et sur les bandes restantes.

**Initialisation** $f_{disponibles} = F_{AAC}$

A chaque **Itération n** (pour chaque bande ERB, en démarrant par la bande la plus basse) :

- On divise le nombre de facteurs d'échelle restants par le nombre de bandes ERB restantes, afin d'obtenir une répartition uniforme théorique des facteurs d'échelle sur toutes les bandes restantes.

$f_{/bande_{theorique}} = \frac{f_{disponibles}}{Nbandes-n+1}$

- Ce nombre théorique est arrondi afin de pouvoir distribuer un nombre entier de coefficients par bloc de codage.

$f_{/bande_{rel}}$ = arrondi($f_{/bande_{theorique}}$).

- On soustrait au nombre de facteurs d'échelle restants le nombre de facteurs d'échelle que l'on vient de répartir sur la bande, afin d'obtenir le nombre de facteurs d'échelle restants pour les bandes suivantes.

$f_{disponibles} = F_{AAC} - f_{/bande_{rel}}$

Sur la dernière bande, on distribue tous les facteurs d'échelle restants.