UNIVERSITÉ DE SHERBROOKE
Faculté de génie
Département de génie électrique et de génie informatique

CODAGE DE PAROLE PAR TRANSFORMÉE POUR LE DÉVELOPPEMENT DE CODEURS PAROLE-AUDIO UNIFIÉS

Thèse de doctorat
Spécialité : génie électrique

Vilayphone VILAYSOUK

Jury : Prof. Roch LEFEBVRE (Directeur de thèse)
Prof. Martin BOUCHARD (Examinateur)
Prof. Eric PLOURDE (Rapporteur)
Prof. Jean ROUAT (Examinateur)

Sherbrooke (Québec) Canada décembre 2015

À mes grand-parents Toum, Angèle et Ké qui me rendent fières de mes racines et qui m'ont permis d'être la personne que je suis aujourd'hui.

"Integrity is doing the right thing, even when no one is watching." - C.S. Lewis

RÉSUMÉ

La compression de tous les types de signaux audio (parole et audio) constitue un vaste domaine de recherche, car il tente de répondre à de nombreuses et différentes demandes provenant de l'industrie. Actuellement, l'industrie de la téléphonie mobile possède de nombreuses requêtes au niveau de la compression de signaux audio à faible débit (sous les 32 kbit/s). Dans cette plage de débit, deux modèles sont nécessaires pour compresser tous les types de signaux audio : les codecs temporels s'utilisent pour la compression des signaux de parole et les codecs fréquentiels (par transformée) plus généraux s'utilisent pour la compression des signaux audio tels que la musique. Les téléphones intelligents et les tablettes numériques représentent des exemples d'appareils qui doivent intégrer deux codecs différents. Idéalement, ces appareils devraient intégrer un codec unique qui compresse tous les types de signaux audio.

Cependant, l'unique moyen actuel d'obtenir un «codec universel» consiste en un «codec hybride universel». Les codecs hybrides universels intègrent au moins deux modèles de codage et un classificateur, qui sélectionne le modèle à exécuter selon le signal à traiter. Ces codecs ne représentent donc pas véritablement des codecs unifiés. De plus, avec l'utilisation d'un classificateur, les codecs hybrides introduisent également la possibilité d'erreurs de classification durant l'analyse. Ces codecs hybrides ont également tendance à être plus complexes puisqu'ils doivent gérer les différents modèles de codage. Après plus de trente ans de recherche, il existe toujours une distinction entre les approches utilisées pour la compression des signaux de parole et celles utilisées pour les signaux audio. Les codecs temporels se basent sur un modèle de production de la parole tandis que les codecs fréquentiels utilisent un modèle de perception auditive pour les signaux audio.

Cette thèse propose des contributions dans l'élaboration d'un modèle de codage audio universel et véritablement unifié. Ces contributions se présentent dans cette thèse par un modèle d'analyse-synthèse de type harmonique-plus-bruit pour les signaux de parole qui fonctionne entièrement dans le domaine fréquentiel. Cette thèse démontre qu'il est possible d'obtenir un signal de parole de qualité perceptuelle transparente sans nécessairement suivre l'évolution de la forme d'onde du signal original.

De plus, cette thèse propose également une version quantifiée du modèle d'analyse-synthèse et démontre qu'il est possible d'obtenir un signal de synthèse de bonne qualité pour des débits autour de 24 kbit/s et de 30 kbit/s. Lors des tests subjectifs MOS, le modèle se situe dans la même catégorie de qualité que la norme G.722.2 (AMR-WB) de l'institut UIT pour un débit autour de 24 kbit/s. Le modèle possède l'avantage de fonctionner entièrement dans le domaine fréquentiel et démontre ainsi les possibilités d'un codec réellement universel puisque traditionnellement le domaine des fréquences était réservé aux signaux audio autres que les signaux de parole.

Mots-clés : Codage de parole, codage audio universel, codage fréquentiel, codage temporel, harmonique-plus-bruit

REMERCIEMENTS

Les travaux de cette thèse ont été effectués au sein du Groupe de Recherche sur la Parole et l'Audio (GRPA) de l'Université de Sherbrooke dirigé par le Professeur Roch Lefebvre. Je tiens à le remercier d'avoir été mon directeur de thèse. Je tiens également à remercier Claude Laflamme de m'avoir proposé un sujet de recherche avec autant de défis scientifiques et d'avoir supervisé mes travaux durant le développement du projet.

Les fonds québécois de la recherche sur la nature et les technologies (FQRNT) et le conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) reçoivent toute l'expression de ma reconnaissance pour avoir permis de mener à bien ce projet.

Je remercie les membres de mon jury de thèse en les personnes du Professeur Martin Bouchard, du Professeur Eric Plourde et du Professeur Jean Rouat pour s'être intéressés à mon travail, pour l'évaluation et la correction de cette thèse.

Merci à tous mes collègues et amis de l'Université de Sherbrooke qui ont contribué de près ou de loin à ce projet. Je leurs adresse toute ma gratitude et leurs souhaite beaucoup de succès dans leur carrière. Un remerciement particulier à Danielle Poirier et Philippe Gournay qui ont grandement contribué à mener à bien ce projet. Je vous remercie pour tous vos judicieux conseils. Un remerciement particulier également à Shoucri et Marc pour m'avoir toujours soutenue et encouragée (merci les amis !).

Je tiens également à remercier de tout cœur ma famille et en particulier mes parents, Bounsoum et Jean pour leur soutien inconditionnel. Je vous remercie d'avoir toujours été présent (malgré la distance) et de tous vos encouragements. Sans votre soutien, je n'aurais pas pu mener à bien ce projet de thèse.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

LEXIQUE

LEXIQUE

signal audio	Comprends tous les signaux audio à l'exception de la parole
pitch	Terme utilisé pour la fréquence fondamentale d'un signal de parole
partie bruit	Partie non-harmonique d'un spectre
partie harmonique	Partie harmonique d'un spectre
spectre mixte	Spectre qui contient une partie harmonique et une partie bruit

LISTE DES ACRONYMES

Acronyme	Définition
3GPP	**3**rd **G**eneration **P**artnership **P**roject
ACR	**A**bsolute **C**ategory **R**ating
AAC-LC	**A**dvanced **A**udio **C**oding - **L**ow **C**omplexity
ACELP	**A**lgebraic **C**ode **E**xcited **L**inear **P**rediction
ADPCM	**A**daptive **D**ifferential **P**ulse **C**ode **M**odulation
AMR-WB	**A**daptive **M**ulti **R**ate - **W**ide**B**and
AMR-WB+	Extended **A**daptive **M**ulti **R**ate - **W**ide**B**and
ANOVA	**AN**alysis **O**f **VA**riance
CELP	**C**ode **E**xcited **L**inear **P**rediction
CM	**C**arré **M**oyen
CODEC	**CO**deur et **DEC**odeur
DCT	**D**iscrete **C**osine **T**ransform
dB	**d**éci**B**el
dl	**d**egré de **l**iberté
EVS	**E**nhanced **V**oice **S**ervices
FAC	**F**orward **A**liasing **C**ancellation
FD	**F**requency **D**omain
FIFO	**F**irst **I**n **F**irst **O**ut
FFT	**F**ast **F**ourier **T**ransform
IEC	**I**nternational **E**lectrotechnical **C**ommission
IETF	**I**nternet **E**ngineering **T**ask **F**orce
IFFT	**I**nverse **F**ast **F**ourier **T**ransform
ISO	**I**nternational **O**rganization for **S**tandardization
HE-AAC	MPEG-4 **H**igh **E**fficiency - **A**dvanced **A**udio **C**oding
HE-AAC(v2)	MPEG-4 **H**igh **E**fficiency - **A**dvanced **A**udio **C**oding **v2**
HSD	**H**onestly **S**ignificant **D**ifference
INMARSAT	**IN**ternational **MAR**itime **SAT**ellite organization
LP	**L**inear **P**rediction
LPC	**L**inear **P**redictive **C**oding
LPD	**L**inear **P**redictive **D**omain
LSD	**L**east **S**ignificant **D**ifference
LTP	**L**ong **T**erm **P**rediction
MBE	**M**ulti**B**and **E**xcitation
MDCT	**M**odified **D**iscrete **C**osine **T**ransform
MNRU	**M**odulated **N**oise **R**eference **U**nit
MOS	**M**ean **O**pinion **S**core
MP3	MPEG-1 layer 3
MPE	**M**ulti-**P**ulse **E**xcited

Acronyme	Définition
MPEG	**M**oving **P**icture **E**xperts **G**roup
MPEG-2 AAC	MPEG-2 **A**dvanced **A**udio **C**oding
MPEG-2 BC	MPEG-2 **B**ackward **C**ompatible
MPEG-2 NBC	MPEG-2 **N**on-**B**ackward **C**ompatible
MPEG-D USAC	MPEG-D **U**nified **S**peech and **A**udio **C**oding
MUSHRA	**MU**ltiple **S**timuli with **H**idden **R**eference and **A**nchor
NMR	**N**oise to **M**ask **R**atio
PCM	**P**ulse **C**ode **M**odulation
PEAQ	**P**erceptual **E**valuation of **A**udio **Q**uality
PESQ	**P**erceptual **E**valuation of **S**peech **Q**uality
PNS	**P**erceptual **N**oise **S**ubstitution
PS	**P**arametric **S**tereo
QV	**Q**uantification **V**ectorielle
RM0	**R**eference **M**odel **0**
RSB	**R**apport **S**ignal sur **B**ruit
SBR	**S**pectral **B**and **R**eplication
SC	**S**omme des **C**arrés
SFM	**S**pectral **F**latness **M**easure
SMR	**S**ignal to **M**ask **R**atio
SNK	**S**tudent-**N**ewman-**K**euls
SNR	**S**ignal to **N**oise **R**atio
STC	**S**inusoidal **T**ransform **C**oding
TNS	**T**emporal **N**oise **S**haping
UIT	**U**nion **I**nternationale des **T**élécommunications
USAC	**U**nified **S**peech and **A**udio **C**oding

CHAPITRE 1

Introduction

La compression de tous les types de signaux audio (parole et audio) constitue un vaste domaine de recherche, car il tente de répondre à de nombreux et différents besoins de l'industrie. Malgré plus de trente années de recherche, il existe toujours une distinction entre les approches utilisées pour la compression des signaux de parole et celles utilisées pour les signaux audio. Actuellement, aucune des approches ne réussit à compresser tous les signaux audio avec une qualité uniforme à de faibles débits (sous les 32 kbit/s).

Dans ces situations de faibles débits, les signaux de parole et les signaux audio se compressent avec deux modèles de codage distincts ainsi que deux domaines de traitement différents. Les téléphones intelligents et les tablettes numériques représentent des exemples d'appareils qui intègrent deux codecs afin d'utiliser des applications multimédias (radio, musique, etc.) en mode continu (*streaming*). Idéalement, ces appareils devraient intégrer un codec unique avec un seul modèle de codage afin de traiter tous les types de signaux audio.

Cependant, l'unique moyen actuel d'obtenir un «codec universel» consiste en un «codec hybride universel». Le codec hybride universel intègre au moins deux modèles de codage et un classificateur qui sélectionne le modèle à exécuter selon le signal à traiter. L'ajout d'un classificateur augmente la complexité d'un codec pour la gestion de la classification et implique également l'introduction d'erreurs de classification. De plus, l'utilisation d'un codec hybride nécessite une gestion des recouvrements des signaux de sortie provenant des différents modèles, car ceux-ci ne possèdent pas nécessairement les mêmes longueurs de trames ou les mêmes types de fenêtrage. Malgré l'augmentation de la complexité et de la gestion des différents modèles, ces codecs hybrides réussissent à obtenir une qualité audio uniforme sur tous les types de signaux audio.

L'existence de ces codecs hybrides universels démontre également qu'il subsiste toujours des modèles de codage et des approches différentes pour la compression des signaux de parole et des signaux audio. Les modèles de codage temporel à forme d'onde obtiennent de bons résultats avec les signaux de parole en utilisant une approche qui reproduit le modèle source-filtre de la production de la parole [Fant, 1960]. Cette approche source-filtre obtient de bons résultats, mais demeure très spécifique à la compression des signaux de parole.

La compression des signaux audio utilise des modèles perceptuels par transformée qui possède une approche plus générique que les modèles à forme d'onde. Ces modèles analysent le signal d'entrée par sous-bande et lui attribue un nombre de bits selon des critères perceptuels de l'oreille. Ces modèles distribuent plus de bits aux sous-bandes où l'oreille possède de plus grandes sensibilités lors de l'écoute. Cette approche fonctionne bien pour tous les types de signaux, mais possède une lacune à faible débit pour les signaux de parole. Lors de situations de faibles débits, les modèles perceptuels par transformée possèdent une performance inférieure avec les signaux de parole, comparativement aux modèles temporels à forme d'onde. Cette lacune avec les signaux de parole à faible débit empêche le modèle par transformée d'être un modèle de codage universel.

Cette brève description de ces deux approches démontre qu'il existe toujours un clivage important entre les approches utilisées pour la compression des signaux de parole et celles utilisées pour la compression des signaux audio lors de débits restreints. Cette thèse tente de réduire ce clivage et de tendre vers une compression de tous les types de signaux audio dans le domaine de la transformée, afin d'aspirer un jour à un modèle de codage universel véritablement unifié. Cette thèse étudie la possibilité de développer un modèle de codage entièrement dans le domaine de la transformée pour les signaux de parole. Le modèle développé tentera d'obtenir une qualité comparable aux modèles de parole existants (norme G.722.2 (AMR-WB) [UIT-T-G.722.2, 2003]) et qui représentent la référence scientifique et industrielle actuelle.

Pour ce faire, cette thèse propose un modèle d'analyse-synthèse de type harmonique-plus-bruit qui fonctionne entièrement dans le domaine de la transformée de Fourier. Le modèle développé permet un suivi précis de l'évolution du pitch afin d'obtenir un signal de synthèse de qualité. Les résultats du test subjectif MUSHRA (***MU****ltiple* ***S****timuli with* ***H****idden* ***R****eference and* ***A****nchor*) démontrent que le modèle d'analyse-synthèse obtient une qualité perceptuelle transparente sans nécessairement suivre la forme d'onde du signal original.

Cette thèse présente également une version quantifiée du modèle d'analyse-synthèse qui fonctionne entièrement dans le domaine de la transformée de Fourier. Le modèle quantifié réussit à obtenir une bonne qualité audio autour de 24 kbit/s et de 30 kbit/s. Les résultats du test subjectif MOS (Mean Opinion Score) situent le modèle quantifié dans la même catégorie de qualité que la norme G.722.2 (AMR-WB) [UIT-T-G.722.2, 2003] pour un débit autour de 24 kbit/s. Le modèle quantifié possède l'avantage de fonctionner entièrement dans le domaine de la transformée et démontre ainsi les possibilités d'un codec réellement

universel puisque traditionnellement le domaine des fréquences était réservé aux signaux audio autres que les signaux de parole.

1.1 Brève description du modèle développé

Le modèle proposé dans cette thèse ne représente pas le seul modèle de codage de parole à utiliser le domaine de la transformée. La section 2.2.1 du chapitre 2 décrit les modèles paramétriques qui utilisent également le domaine fréquentiel afin d'extraire les paramètres d'un signal de parole.

La figure 1.1 montre le schéma général de fonctionnement qui est commun aux modèles de codage par transformée. Le modèle développé apporte quelques modifications au schéma général de la figure 1.1 et présente ces modifications dans la figure 1.2.

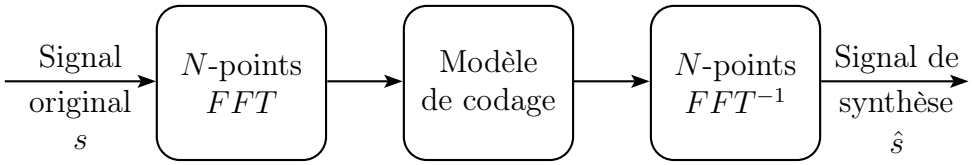


Figure 1.1 Principe général d'un modèle de codage par transformée

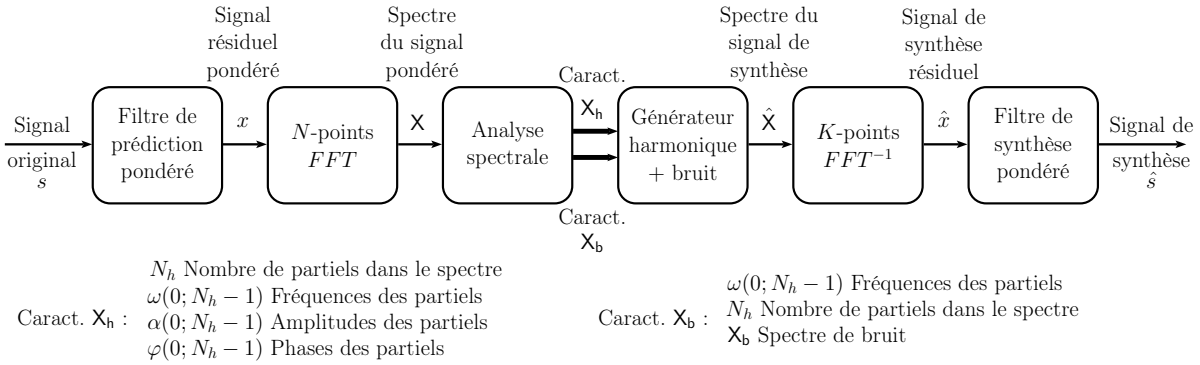


Figure 1.2 Principe général de fonctionnement du modèle proposé

La figure 1.2 montre que le modèle développé utilise un nombre de points différents ($N > K$) pour les transformées de Fourier et de Fourier inverse. Pour obtenir une grande précision durant l'analyse, le modèle utilise un nombre de points plus élevé pour la transformée de Fourier. Durant la synthèse, le modèle utilise un nombre de points plus faible pour la création du spectre afin d'obtenir un bon suivi du pitch. Les détails complets du modèle se trouvent dans les chapitres 3 et 4 de cette thèse. La prochaine section énumère les originalités apportées par le modèle d'analyse-synthèse et sa version quantifiée.

1.1.1 Énumération des originalités apportées par le modèle et son codec

Le modèle d'analyse-synthèse et sa version quantifiée contiennent les originalités suivantes :

- La possibilité d'atteindre la qualité (ou de s'approcher) de la transparence pour le signal de synthèse sans nécessairement suivre la forme d'onde du signal original ;

- La démonstration d'un codage efficace du signal de parole dans le domaine des fréquences, et démontre les possibilités d'un codec réellement unifié dans le domaine fréquentiel, qui est traditionnellement réservé aux signaux audio ;

- La possibilité de représenter un signal audio dans le domaine fréquentiel sans les contraintes de la reconstruction parfaite qui caractérise les approches fréquentielles existantes. En particulier, le modèle démontre avec l'analyse (l'encodeur) et la synthèse (décodeur) une relative indépendance l'une à l'autre avec les différentes longueurs lors de l'utilisation des transformées. Dans l'approche présentée dans le chapitre 3, l'analyse et la synthèse s'effectuent dans le domaine fréquentiel, mais avec des résolutions très différentes ;

- Finalement, la démonstration d'un codage audio efficace dans le domaine fréquentiel sans avoir à respecter la contrainte d'échantillonnage critique qui caractérise les approches fréquentielles MDCT (***M**odified **D**iscrete **C**osine **T**ransform*). L'utilisation d'une transformée de Fourier pour la synthèse facilite certaines opérations, comme l'usage de la prédiction dans le domaine fréquentiel. Cette démonstration de la prédiction possible dans le domaine fréquentiel est présentée dans le chapitre 4 avec la prédiction des phases.

1.2 Organisation du document

Cette thèse se divise en 6 chapitres. Le chapitre 2 décrit l'état actuel des recherches dans le domaine de la compression des signaux audio à faible débit. Ce chapitre de revue littéraire cible les modèles de codage existants ayant de faibles débits afin de mieux positionner le modèle de codage développé. Le chapitre 3 décrit les détails du modèle d'analyse-synthèse proposé par cette thèse. Le chapitre 4 donne les détails du modèle quantifié développé à partir du modèle d'analyse-synthèse proposé du chapitre 3. Le chapitre 5 explique les tests réalisés et les résultats obtenus sur les différentes parties du modèle ainsi que les analyses

des résultats effectuées. Finalement, le chapitre 6 contient les conclusions de cette thèse et les travaux futurs envisageables qui en découlent.

CHAPITRE 2

REVUE DES MODÈLES DE CODAGE AUDIO

Afin de positionner le modèle développé avec les modèles existants, ce chapitre présente la situation actuelle des recherches dans le domaine de la compression de tous les types de signaux audio. Puisque la compression des signaux audio constitue un vaste domaine de recherche, ce chapitre se concentre particulièrement sur les modèles de codage fonctionnant à faible débit. C'est particulièrement dans cette plage de débit qu'il existe un clivage entre les approches utilisées pour la compression des signaux de parole et de l'audio.

La figure 2.1 montre qu'il n'existe aucun modèle de codage qui réussit à compresser tous les types de signaux audio avec une qualité uniforme à faible débit. L'unique moyen de compresser tous les types de signaux audio avec une qualité uniforme s'effectue en combinant deux modèles de codage de la figure 2.1 afin d'obtenir un codec hybride universel. La section 2.4 de ce chapitre décrit un codec hybride universel appelé USAC (***U****nified* ***S****peech and* ***A****udio* ***C****oding*) qui intègre un modèle de codage source à forme d'onde ainsi qu'un modèle perceptuel par transformée.

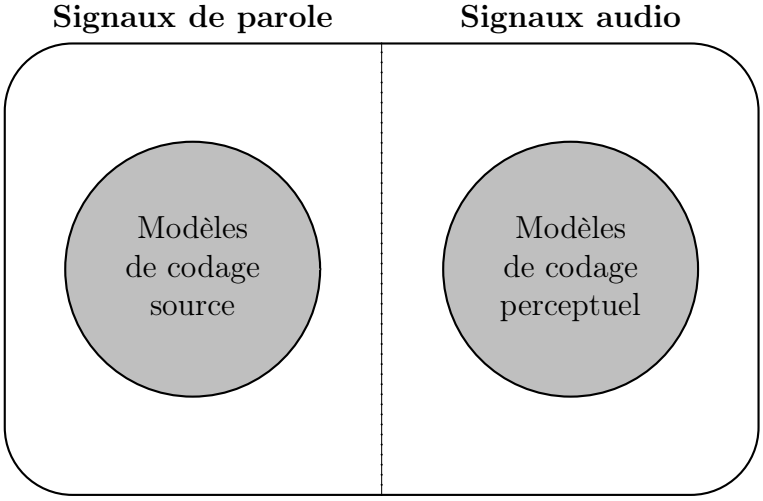


Figure 2.1 Modèles de codage à faible débit

Déroulement du chapitre

Ce chapitre se divise en 5 grandes sections. Les 2 premières sections décrivent deux modèles de codage de type source : les modèles de codage temporel à forme d'onde et les modèles de codage paramétrique. La section 3 décrit les modèles de codage perceptuel par transformée. Chacune de ces sections décrit en ordre chronologique les modèles importants. La section 4 décrit un codec hybride universel appelé USAC. Finalement, la dernière section est la conclusion de ce chapitre.

2.1 Modèles de codage temporel à forme d'onde

Les modèles de codage temporel à forme d'onde représentent un groupe important pour la compression des signaux de parole. Ces modèles tentent de représenter le plus fidèlement possible la forme d'onde du signal original avec une approche qui reproduit le modèle source-filtre de la production de la parole [Fant, 1960]. Cette approche spécifique possède de bons résultats avec des signaux de parole, mais ne réussit pas à obtenir un même niveau de qualité avec les signaux audio.

Les grandes familles des modèles de type CELP (***C**ode **E**xcited **L**inear **P**rediction*) utilisent cette approche de codage à forme d'onde. Afin de reproduire le plus fidèlement possible la forme d'onde du signal original, les modèles CELP utilisent les techniques suivantes : une boucle d'analyse-par-synthèse, un filtre de prédiction à long-terme et un filtre perceptuel. Le modèle ACELP (***A**lgebraic **C**ode **E**xcited **L**inear **P**rediction*) qui est contenu dans plusieurs standards appartient à la famille CELP.

Les prochaines sections décrivent les modèles importants qui ont contribué à l'évolution de la famille des modèles CELP. En premier lieu, cette section décrira le fonctionnement du modèle MPE (***M**ulti-**P**ulse **E**xcited*) qui n'appartient pas directement à la famille CELP, mais c'est avec ce modèle que les auteurs [Singhal et Atal, 1984] proposent une boucle d'analyse-par-synthèse et un filtre perceptuel, des techniques importantes que les modèles CELP utilisent par la suite.

Cette section décrit ensuite le modèle CELP qui propose une alternative au générateur d'excitations du modèle MPE avec un dictionnaire de type stochastique. Finalement, cette section décrit le modèle ACELP qui remplace le dictionnaire stochastique du modèle CELP par un dictionnaire algébrique. De plus, le modèle ACELP propose quelques modifications au schéma de fonctionnement du modèle CELP afin de réduire la complexité de la recherche

dans le dictionnaire. Cette réduction de complexité et le changement de dictionnaire permettent un fonctionnement en temps réel au modèle ACELP.

2.1.1 MPE (Multi-Pulse Excited) (1980)

C'est au début des années 80, que [Atal, 1986] propose une alternative au modèle simple de codage LPC-10 (***L**inear **P**redictive **C**oding*) [Tremain, 1982] avec le modèle de codage MPE (***M**ulti-**P**ulse **E**xcited*).

Le modèle LPC-10 représente la modélisation la plus simple du signal de parole. Le modèle analyse le signal de parole par trame et classifie celle-ci en voisée ou non-voisée. La figure 2.2 montre que le modèle crée un train d'impulsions périodique à la valeur du pitch du signal pour une trame déclarée voisée, tandis que pour une trame non-voisée, le modèle crée du bruit.

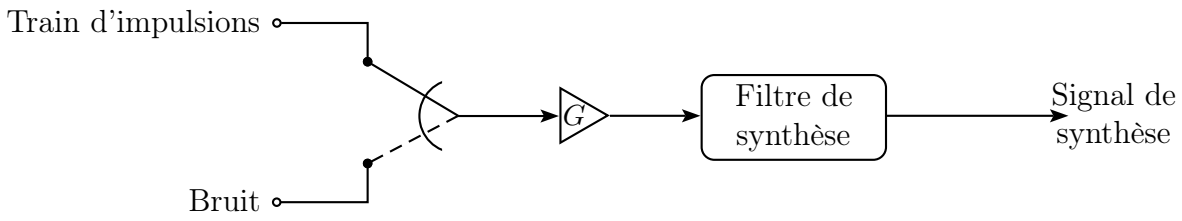


Figure 2.2 Création du signal de synthèse de parole par le modèle LPC-10

Le modèle LPC-10 fonctionne à faible débit (≈ 4 kbit/s) [Atal, 1986], mais certains auteurs [Makhoul *et al.*, 1978] mentionnent que le signal de synthèse possède un bourdonnement (*buzzy*) et un manque de richesse comparativement au signal original.

Le modèle LPC-10 utilise une modélisation rudimentaire du signal de parole et qui ne permet pas d'obtenir une qualité perceptuelle transparente semblable au signal original avec un débit infini.

Hypothèses sur le manque de la qualité perceptuelle du modèle LPC-10

Les auteurs [Makhoul *et al.*, 1978] posent l'hypothèse que le bourdonnement et le manque de richesse du signal de synthèse du modèle LPC-10 proviendraient du choix limité des excitations possibles : un train d'impulsions pour les segments voisés et du bruit pour les segments non-voisés. D'ailleurs, la figure 2.3 montre qu'il existe une autre excitation possible appelée mixte qui contient à la fois une partie voisée et une partie non-voisée dans un même segment.

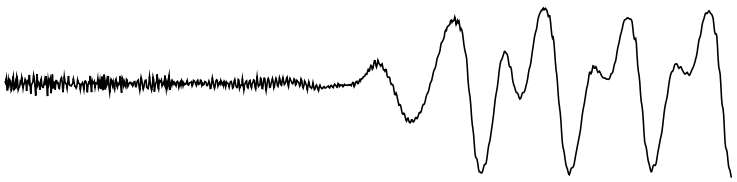
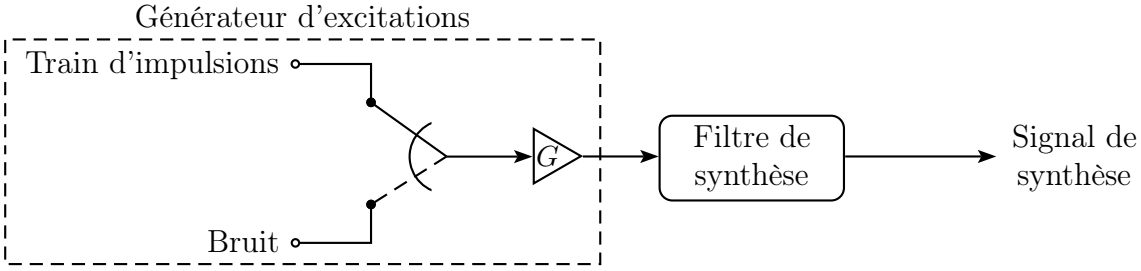


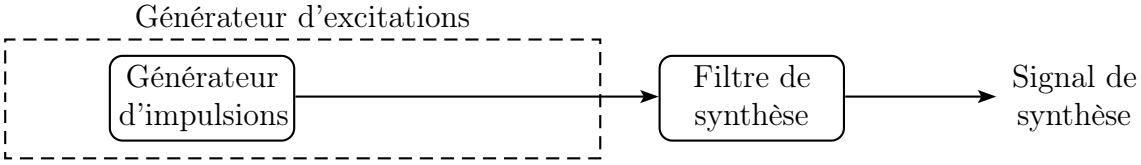
Figure 2.3 Segment d'un signal de parole de type mixte

Avec ces différentes excitations possibles à traiter, les auteurs [Atal et Remde, 1982] proposent un générateur d'excitations avec un nombre fixe d'impulsions pour recréer tous les types de signaux. Les auteurs [Atal et Remde, 1982] proposent ce générateur d'excitations dans le modèle de codage MPE (***M**ulti-**P**ulse **E**xcited*).

La figure 2.4 compare les modélisations d'un signal de parole entre les modèles LPC-10 et MPE. La différence majeure entre les deux modèles se situe au niveau du type de générateur d'excitations utilisé pour créer les signaux de synthèse. Le modèle LPC-10 possède un générateur avec deux types d'excitations tandis que le modèle MPE utilise un générateur d'impulsions. De plus, l'utilisation d'un générateur d'impulsions par le modèle MPE enlève le classificateur et élimine par le fait même les erreurs de classification possibles.



Modélisation de la parole avec le modèle LPC-10



Modélisation de la parole avec le modèle MPE

Figure 2.4 Comparaison des générateurs d'excitations des modèles LPC-10 et MPE

Dans le modèle MPE, le générateur crée une excitation $v(n)$ à l'aide d'impulsions possédant des positions m_k et des amplitudes β_k précises (cf. équation 2.1).

$$v(n) = \sum_{k=0}^{M-1} \beta_k \delta(n - m_k) \qquad \begin{array}{ll} n = & 0, ..., M-1 \\ M = & \text{nombre d'impulsions} \end{array} \tag{2.1}$$

Selon l'auteur [Atal, 1986], le générateur utilise que quelques impulsions afin de créer tous les types de signaux de parole avec peu de distorsion audible. Pour des trames de 10 ms (80 échantillons) et une fréquence d'échantillonnage de 8 kHz, l'auteur [Atal, 1986] utilise entre huit et seize impulsions pour créer une excitation. Les prochaines sections décrivent la procédure utilisée afin de sélectionner la meilleure position et la meilleure amplitude de chaque impulsion.

Recherche de la position et de l'amplitude optimales des impulsions

Afin d'obtenir le meilleur signal d'excitation, le modèle MPE effectue une recherche de la position et de l'amplitude optimales pour chaque impulsion δ. Une recherche simultanée des positions et des amplitudes possibles de toutes les impulsions représente une grande complexité de calcul dû au nombre élevé de combinaisons. Par exemple, pour une longueur de trame de 40 échantillons contenant quatre impulsions il existe 91320 combinaisons possibles pour les positions uniquement [Salami, 1995].

Pour diminuer la complexité des calculs, le modèle MPE [Atal, 1986] utilise un processus itératif afin d'obtenir la position et l'amplitude de chaque impulsion δ. Le modèle recherche la meilleure position et calcule ensuite la valeur de l'amplitude de cette impulsion. Le processus itératif reste sous-optimal, mais possède l'avantage de diminuer la complexité des calculs au niveau du nombre d'opérations à exécuter.

L'itération s'effectue dans une boucle d'analyse-par-synthèse au niveau de l'encodeur. Cette boucle reste fermée tant que l'erreur pondérée entre le signal original et le signal de synthèse ne se situe pas sous un seuil prédéfini (cf. figure 2.5 [Atal et Remde, 1982]).

La figure 2.5 montre que la boucle d'analyse-par-synthèse utilise un filtre perceptuel $W(z)$ qui modifie l'enveloppe spectrale de l'erreur. Ce filtre $W(z)$ applique les concepts de la perception de l'oreille humaine sur le signal d'erreur $e(n)$.

L'équation 2.2 [Salami, 1995] montre comment $W(z)$ modifie l'enveloppe spectrale du signal à l'aide du filtre de synthèse $1/A(z)$.

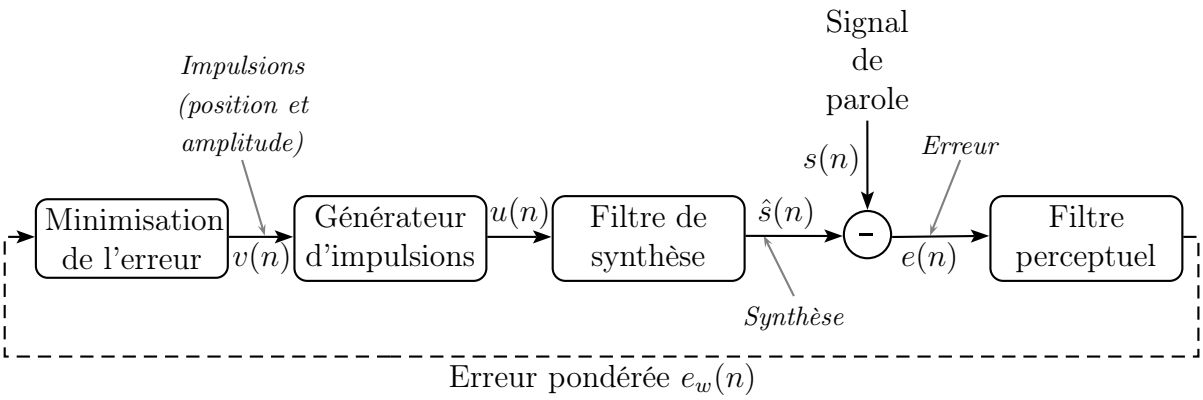


Figure 2.5 Boucle d'analyse-par-synthèse du modèle MPE à l'encodeur

$$W(z) = \frac{A(z)}{A(z/\gamma)} = \frac{1 - \sum_{k=0}^{P-1} a_k z^{-k}}{1 - \sum_{k=0}^{P-1} a_k \gamma^k z^{-k}} \qquad P = \text{ordre du filtre } A(z) \tag{2.2}$$

La valeur de γ de l'équation 2.2 contrôle le degré d'affaissement et varie habituellement entre 0 et 1. Avec une valeur de $\gamma = 0$, l'erreur pondérée $e_w(n)$ se situe dans le domaine de l'excitation du filtre LPC de synthèse ($W(z) = 1/A(z)$). Et à l'opposé, lorsque $\gamma = 1$, l'erreur pondérée se situe dans le domaine du signal original. La figure 2.6 [Salami, 1995] montre qu'une diminution de la valeur de γ entraîne une augmentation des largeurs des résonances introduites par les pôles dans la réponse en fréquence de $W(z)$.

Ainsi, le filtre perceptuel $W(z)$ diminue l'énergie des formants afin de mieux répartir le signal d'erreur dans le domaine spectral. Le filtre perceptuel considère les propriétés de masquage du bruit de quantification en pondérant plus fortement l'erreur de quantification dans les zones de faibles amplitudes et plus faiblement dans les zones de formants du spectre.

Les modèles temporels à forme d'onde utilisent $W(z)$ pour modifier le bruit de quantification afin que son spectre suive l'enveloppe du signal original. Le filtre perceptuel $W(z)$ effectue une mise en forme du bruit tout en s'assurant que le spectre du bruit de quantification soit masqué par le spectre du signal de parole. Le résultat de ce filtrage donne un signal d'erreur pondéré $e_w(n)$ qui représente le nouveau critère de comparaison avec le seuil prédéfini.

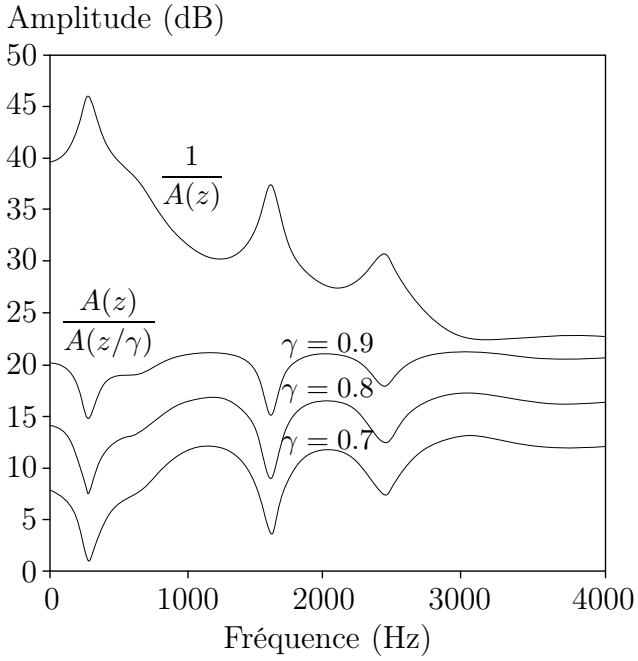


Figure 2.6 Spectre d'amplitudes de $\frac{1}{A(z)}$ et $\frac{A(z)}{A(z/\gamma)}$ avec différentes valeurs de γ

Exemple d'ajout d'impulsions dans une excitation

La figure 2.7 [Atal et Remde, 1982] montre un exemple de la création d'un signal d'excitation. Au début, le signal d'excitation ne possède aucune impulsion (cf. figure 2.7a). Le signal de synthèse à la figure 2.7a provient de l'effet de la mémoire du filtre de l'ancienne excitation créée.

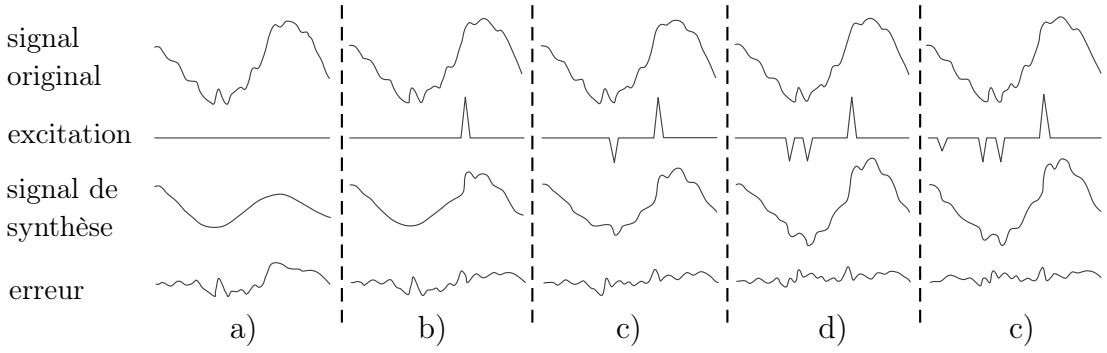


Figure 2.7 Exemple de l'évolution du signal de synthèse par l'ajout d'impulsions

La soustraction du signal de parole original avec la contribution de la mémoire du filtre et l'ajout d'une impulsion représente la nouvelle excitation (cf. figure 2.7b). Le modèle calcule une nouvelle erreur et détermine la contribution de l'impulsion ajoutée. Le processus d'ajout d'impulsions s'exécute tant que l'erreur n'atteint pas un seuil minimum préétabli.

Dans la figure 2.7, l'excitation contient uniquement quatre impulsions, mais le processus ajoute autant d'impulsions que nécessaire. Les auteurs [Atal et Remde, 1982] mentionnent qu'après huit impulsions la diminution de l'erreur pondérée devient peu significative.

Les résultats présentés dans l'article [Atal et Remde, 1982] démontrent que le signal de synthèse suit l'évolution du signal original et qu'il s'adapte rapidement durant les transitions. Avec quelques impulsions, le modèle parvient à créer tous les types de signaux : voisé, non-voisé et mixte. Durant les tests d'écoute, les auteurs [Atal et Remde, 1982] mentionnent que le signal de synthèse ressemble au signal original et que les sons *buzzy* produits par le modèle LPC-10 n'y sont plus. Le modèle de codage MPE obtient de bons résultats, mais possède une grande complexité de calcul dû à son processus itératif pour la création de l'excitation. La prochaine section décrit le modèle CELP (***C**ode **E**xcited **L**inear **P**rediction*) qui diminue cette complexité de calcul en remplaçant le générateur d'impulsions par un dictionnaire stochastique pour la création de l'excitation.

De plus, au même moment l'industrie demandait un codec ayant un débit sous les 8 kbit/s, mais ce n'est qu'à partir de 9.6 kbit/s [Salami, 1995] que le modèle MPE obtient un bon signal de synthèse. Le modèle MPE nécessite un débit minimum afin d'encoder son excitation. La qualité du signal de synthèse se détériore rapidement lorsque le modèle limite le débit attribué aux impulsions ou que le nombre d'impulsions diminue. Ainsi, pour augmenter la qualité du signal de synthèse tout en réduisant le débit, les auteurs [Schroeder et Atal, 1985] du CELP proposent de changer le générateur d'impulsions du modèle MPE pour un dictionnaire stochastique.

2.1.2 CELP (Code Excited Linear Prediction) (1985)

Le modèle de codage CELP (***C**ode **E**xcited **L**inear **P**rediction*) offre une solution au problème de complexité de calcul du modèle MPE de la section 2.1.1. La différence majeure entre le modèle MPE de 1980 et le modèle CELP de 1985 se situe sur le type de générateur d'excitations utilisé. Le modèle CELP remplace le générateur d'impulsions par un dictionnaire (*codebook*) stochastique pour la création de l'excitation. Le dictionnaire possède 1024 excitations de type gaussien [Schroeder et Atal, 1985], aussi appelées des innovations. La recherche effectuée avec la boucle d'analyse-par-synthèse reste similaire au modèle MPE à l'exception que ce n'est pas une impulsion à la fois, mais une excitation à la fois.

Excitation stochastique pour générer un signal de synthèse

La figure 2.8 montre comment le modèle CELP crée un signal de synthèse avec un bruit gaussien. Le modèle utilise un filtre de prédiction à long-terme qui s'ajoute au prédicteur

court-terme afin de créer le signal de synthèse [Schroeder et Atal, 1985]. La prédiction long-terme reconstitue la corrélation du signal entre les échantillons plus éloignés et qui correspond à la période (pitch) du signal.

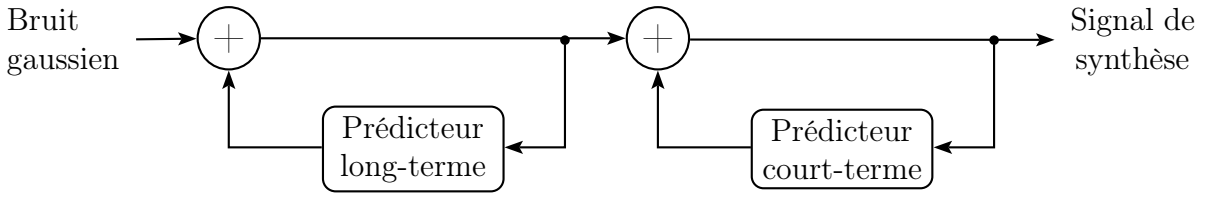


Figure 2.8 Génération du signal de synthèse avec les deux prédicteurs

La figure 2.9 [Adoul *et al.*, 1987] montre l'utilisation des prédicteurs de la figure 2.8 dans l'encodeur CELP afin de créer un signal de synthèse $\hat{s}(n)$. L'encodeur recherche la meilleure innovation k filtrée du dictionnaire afin d'obtenir le signal de synthèse qui ressemblera le plus possible au signal d'entrée $s(n)$.

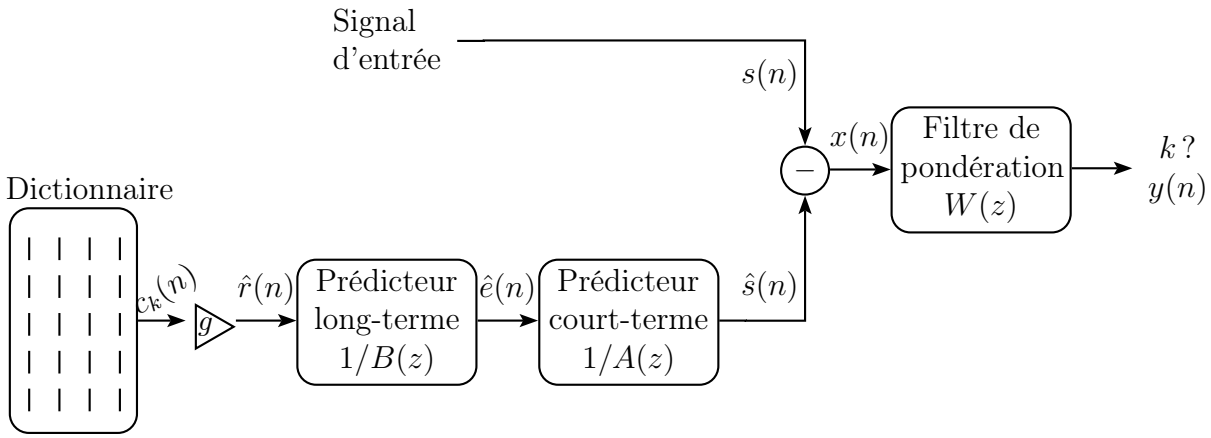


Figure 2.9 Schéma de fonctionnement à l'encodeur du modèle CELP

Recherche dans le dictionnaire

Pour trouver l'innovation optimale, le modèle CELP compare les innovations filtrées avec le signal de référence, aussi appelé la cible $\mathbf{x}$ (cf. équation 2.3 [Laflamme *et al.*, 1991]). La soustraction de la mémoire du filtre de synthèse pondéré $1/A(z/\gamma)$ au signal original donne la cible $\mathbf{x}$. La variable g représente le facteur de gain et $\mathbf{H}$ est une matrice construite à partir de réponses impulsionnelles du filtre de synthèse pondéré. Le symbole mathématique $\|...\|^2$ indique la somme au carré des vecteurs et les lettres en caractères gras représentent des vecteurs.

$$E_k \quad = \quad \|\mathbf{x} - g\mathbf{H}\mathbf{c}_k\|^2 \tag{2.3}$$

Pour trouver la valeur du gain optimal g, le modèle applique une dérivée partielle en fonction de $\partial E_k/\partial g = 0$ (cf. équation 2.4 [Laflamme *et al.*, 1991]).

$$\frac{\partial E_k}{\partial g} = 0 \quad \longrightarrow \quad g = \frac{\mathbf{x^T H c_k}}{\|\mathbf{H c_k}\|^\mathbf{2}} \tag{2.4}$$

Le développement de l'équation 2.3 avec l'équation 2.4 donne le résultat de l'équation 2.5 [Laflamme *et al.*, 1991].

$$E_k = \|\mathbf{x}\|^\mathbf{2} - \frac{\mathbf{x^T H c_k}}{\|\mathbf{H c_k}\|^\mathbf{2}} \tag{2.5}$$

L'encodeur CELP effectue le calcul de l'équation 2.5 pour chaque innovation du dictionnaire. L'équation 2.5 implique un temps de calcul élevé, car l'encodeur possède un dictionnaire de 1024 innovations [Schroeder et Atal, 1985]. Les auteurs [Schroeder et Atal, 1985] mentionnent que cela prenait 125 secondes de temps de calcul pour une seconde de signal de parole avec un superordinateur CRAY-1.

Le temps de calcul élevé provient principalement des recherches effectuées dans le dictionnaire pour trouver la meilleure innovation. Malgré cette grande complexité de calcul, le modèle CELP obtient un signal de qualité à un débit autour de 4.8 kbit/s [Salami, 1995].

La prochaine section 2.1.3 présente une technique qui diminue ce temps de calcul élevé avec l'utilisation d'un dictionnaire algébrique au lieu d'un dictionnaire stochastique, ainsi que l'ajout d'une méthode de recherche plus rapide. Ces techniques d'amélioration s'intègrent dans le modèle de codage ACELP (**A**lgebraic **C**ode **E**xcited **L**inear **P**rediction) qui possède un signal de parole de qualité à des débits allant de 4.8 kbit/s à 16 kbit/s [Laflamme *et al.*, 1990] avec un fonctionnement en temps réel.

2.1.3 ACELP (Algebraic Code Excited Linear Prediction) (1990)

Comme mentionné précédemment dans la section 2.1.2, le modèle CELP obtient un signal de synthèse de qualité autour de 4.8 kbit/s [Salami, 1995], mais il requiert un temps de calcul élevé. Le temps élevé provient principalement de la recherche dans le dictionnaire afin de trouver la meilleure innovation. Cette section décrit le modèle ACELP (***A****lgebraic* ***C****ode* ***E****xcited* ***L****inear* ***P****rediction*) qui utilise un dictionnaire algébrique au lieu du dictionnaire stochastique [Laflamme *et al.*, 1990] et qui propose également une modification du schéma de fonctionnement à l'encodeur du modèle CELP afin de réduire la complexité de la recherche.

Plusieurs standards de l'UIT (**U**nion **I**nternationale des **T**élécommunications) contiennent une version du modèle ACELP [UIT-T-G729, 2007][UIT-T-G.722.2, 2003]. La figure 2.10 (inspirée de [VoiceAge, 2011]) montre les standards internationaux qui intègrent une version du modèle ACELP.

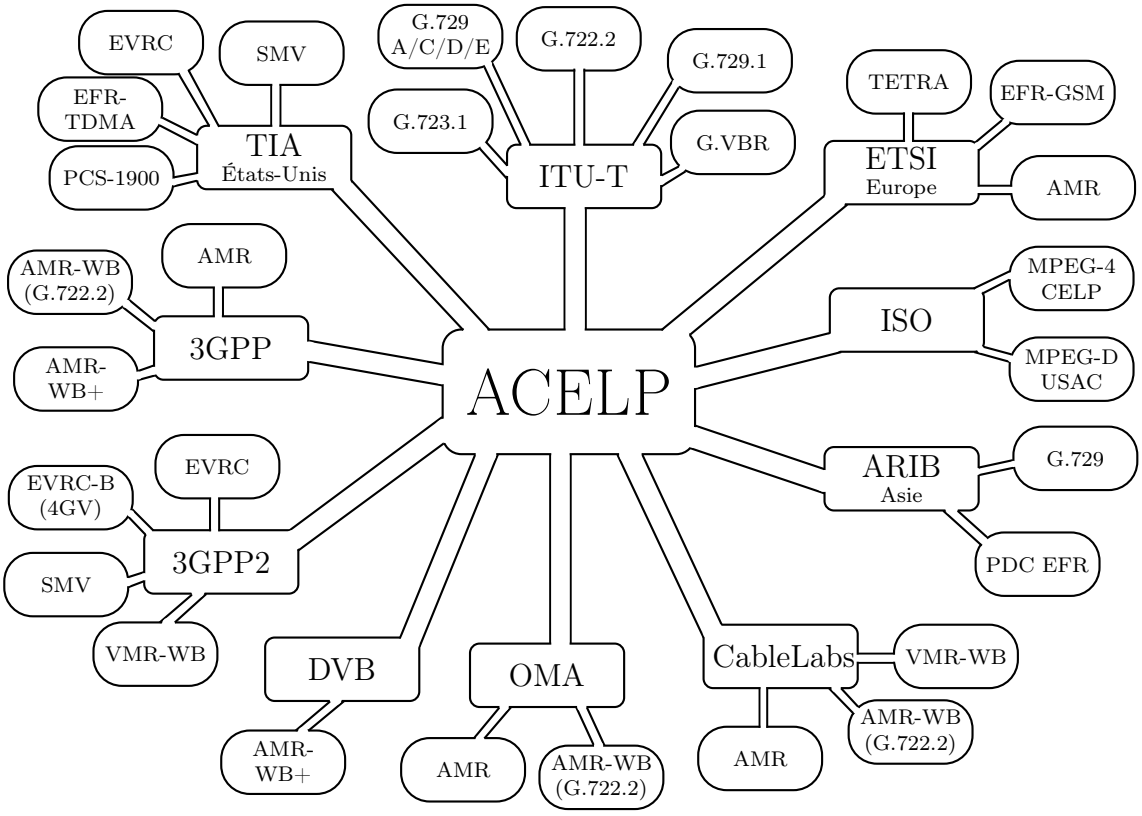


Figure 2.10 Standards internationaux qui intègrent une version du modèle ACELP

Les prochaines sections décrivent le dictionnaire algébrique ainsi que la méthode de recherche utilisée par le modèle de codage ACELP.

Utilisation d'un dictionnaire algébrique

Le modèle ACELP remplace le dictionnaire stochastique du modèle CELP par un dictionnaire algébrique. Dans un dictionnaire algébrique, les excitations possèdent peu d'impulsions et leur valeur d'amplitude vaut $+1$ ou -1. Ces deux caractéristiques d'un dictionnaire algébrique, peu d'impulsions et un nombre restreint de valeurs possibles pour l'amplitude permettent de réduire la complexité de la recherche dans le dictionnaire.

Cependant, le changement de dictionnaire ne représente pas le seul facteur qui permet d'obtenir un modèle de codage qui fonctionne en temps réel. Un second facteur qui consiste en une modification du schéma original de fonctionnement à l'encodeur du modèle CELP a permis de réduire la complexité et le temps de la recherche.

Modifications au schéma de l'encodeur du modèle CELP

Afin de réduire la complexité de la recherche dans le dictionnaire, le modèle ACELP propose des modifications au schéma de l'encodeur du modèle CELP de la figure 2.9. La figure 2.11 montre les modifications apportées par le modèle ACELP. Dans le schéma de la figue 2.11, le modèle change l'emplacement du filtre perceptuel $W(z)$ afin que le filtrage s'effectue avant le calcul de l'erreur quadratique entre le signal original et le signal de synthèse. Ce changement de position du filtre perceptuel $W(z)$ modifie l'équation 2.3 du calcul de l'erreur de la section 2.1.2.

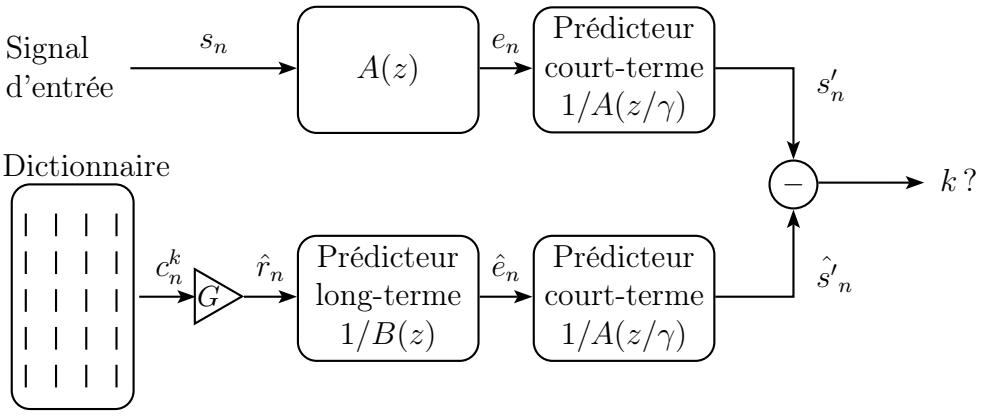


Figure 2.11 Schéma de fonctionnement à l'encodeur du modèle ACELP

Comme mentionné dans la section 2.1.2, l'erreur E s'exprime comme suit [Laflamme *et al.*, 1990]

$\mathbf{S'}$	Signal d'entrée après l'utilisation du filtre perceptuel
$W(z) = A(z)/A(z/\gamma)$	Filtre perceptuel
$\mathbf{P}$	Prédiction du pitch avec le *ringing* de l'excitation passée
$\mathbf{X} = \mathbf{S'} - \mathbf{P}$	Signal cible (la référence à atteindre)
$\hat{\mathbf{X}} = \mathbf{gCH^T}$	Innovation avec le filtre perceptuel
$\mathbf{C} = (\mathbf{c_0}, \mathbf{c_1}, ..., \mathbf{c}_{M-\mathbf{1}})$	Dictionnaire contenant les innovations de longueur L

Tableau 2.1 Définition des variables de calcul pour le modèle ACELP

$$E = \|\mathbf{X} - \mathbf{gCH^T}\|^{\mathbf{2}} = \|\mathbf{X}\|^{\mathbf{2}} - \frac{(\mathbf{X}(\mathbf{CH^T})^{\mathsf{T}})^{\mathbf{2}}}{\|\mathbf{CH^T}\|} \tag{2.6}$$

L'équation 2.7 montre que la minimisation de $\mathbf{g}$ dans l'équation 2.6 s'effectue en maximisant la valeur absolue. Cette minimisation de l'erreur ainsi que le filtrage $\mathbf{H}$ s'effectue pour chaque innovation, et ce qui a pour conséquence une complexité de calcul élevé pour le modèle CELP.

$$\max_{\mathbf{k}} \left| \frac{\mathbf{X}(\mathbf{C_k H^T})^{\mathsf{T}}}{\sqrt{\|\mathbf{C_k H^T}\|^{\mathbf{2}}}} \right| = \max_{\mathbf{k}} \left| \frac{(\mathbf{XH})\mathbf{C_k^T}}{\sqrt{\|\mathbf{C_k H^T}\|^{\mathbf{2}}}} \right| \tag{2.7}$$

Pour diminuer cette complexité de calcul lors de la recherche, le modèle ACELP modifie la structure du dictionnaire d'innovations de l'équation 2.7. Le modèle ACELP propose d'intégrer le filtre perceptuel $\mathbf{F}$ dans le dictionnaire algébrique $\mathbf{C}$. La figure 2.12 [Laflamme *et al.*, 1990] montre la solution proposée pour le dictionnaire par le modèle ACELP.

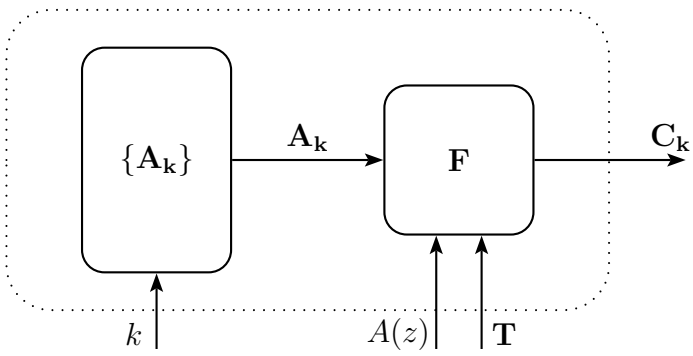


Figure 2.12 Description du module dictionnaire dans le modèle ACELP

Dans le schéma de la figure 2.12, le dictionnaire d'innovations devient la combinaison d'un dictionnaire algébrique et d'un filtre perceptuel ($\mathbf{C} = \mathbf{AF^T}$). Cette modification change certaines variables dans l'équation 2.6. L'équation 2.8 [Laflamme *et al.*, 1990] montre les

modifications au niveau du numérateur de l'équation 2.7 pour la recherche de la meilleure innovation.

$$\underset{\mathbf{k}}{\mathrm{Max}} \left| \frac{(\mathbf{XH})\mathbf{C}_\mathbf{k}^\mathsf{T}}{\sqrt{\|\mathbf{C_k H}^\mathsf{T}\|^\mathbf{2}}} \right| = \underset{\mathbf{k}}{\mathrm{Max}} \left| \frac{(\mathbf{XHF})\mathbf{A}_\mathbf{k}^\mathsf{T}}{\sqrt{\|\mathbf{C_k H}^\mathsf{T}\|^\mathbf{2}}} \right| \tag{2.8}$$

L'équation 2.9 [Laflamme *et al.*, 1990] montre les modifications au dénominateur de l'équation 2.7).

$$\begin{aligned} \sqrt{\|\mathbf{C_k H}^\mathsf{T}\|^\mathbf{2}} &= (\sqrt{\|\mathbf{C_k H}^\mathsf{T}\|^\mathbf{2}})^\mathbf{2} \\ &= \|\mathbf{C_k H}^\mathsf{T}\|^\mathbf{2} \\ &= \mathbf{A_k F}^\mathsf{T}\mathbf{H}^\mathsf{T}\mathbf{HFA}_\mathbf{k}^\mathsf{T} \\ &= \mathbf{A_k T}^\mathsf{T}\mathbf{TA}_\mathbf{k}^\mathsf{T} \\ &= \mathbf{A_k UA}_\mathbf{k}^\mathsf{T} \end{aligned} \tag{2.9}$$

Le résultat de l'équation 2.9 montre qu'il n'est plus nécessaire de filtrer les innovations à chaque analyse. Le calcul de la matrice **U** s'exécute une seule fois et devient un paramètre constant pour toutes les innovations. De plus, comme les innovations possèdent peu de valeurs non nulles cela entraîne également une diminution des calculs. Ainsi, tous ces ajouts et modifications donnent un modèle ACELP qui fonctionne en temps réel tout en ayant une bonne qualité audio autour de 4.8 kbit/s [Adoul *et al.*, 1987].

2.1.4 Conclusion sur le codage temporel à forme d'onde

Les sections précédentes décrivaient les modèles de codage temporel à forme d'onde qui par leur approche tente de reproduire l'enveloppe du signal de parole. Ces modèles représentent le meilleur compromis afin d'obtenir un signal de synthèse de qualité avec le plus bas débit possible. Les prochaines sections décrivent les modèles de codage paramétrique qui tentent d'extraire toutes les caractéristiques du signal de parole afin de les transmettre au décodeur. Le codage paramétrique représente une autre possibilité pour la compression des signaux de parole à faible débit.

2.2 Modèles de codage paramétrique

Les modèles paramétriques extraient les paramètres importants du signal de parole afin de créer le signal de synthèse. Ces modèles fonctionnent à des débits plus faibles que les modèles à forme d'onde, mais ne réussissent pas à conserver un signal de synthèse de qualité. Les modèles paramétriques s'utilisent lors de conditions de débits restreints pour l'encodage du signal de parole. Par exemple, pour des systèmes de communication par satellite comme Inmarsat M et Iridium [Richharia et Westbrook, 2010] qui intègrent des versions du modèle paramétrique MBE (***M**ulti**B**and **E**xcitation*).

Les modèles MBE et STC (***S**inusoidal **T**ransform **C**oding*) appartiennent au groupe de codage paramétrique. Ces modèles extraient les caractéristiques des composantes sinusoïdales du signal de parole afin de transmettre ces paramètres au décodeur. Comme pour les modèles temporels à forme d'onde, les études sur les modèles paramétriques sinusoïdaux ont également commencé dans les années 80. Les modèles paramétriques possèdent aussi pour objectif d'améliorer la qualité audio du modèle de codage LPC-10, mais en offrant des solutions différentes des modèles à forme d'onde. Les prochaines sections décrivent les différentes approches utilisées par les modèles MBE et STC afin de modéliser les signaux.

2.2.1 MBE (MultiBand Excitation) (1985)

Après des observations de spectres d'amplitudes mixtes, les auteurs [Griffin et Lim, 1985] proposent une analyse par sous-bandes avec le modèle MBE (***M**ulti**B**and **E**xcitation*). L'analyse par sous-bandes permet de représenter les alternances de voisement possibles dans les spectres mixtes. Le nombre de sous-bandes varie selon la valeur de la fréquence fondamentale ω_0 et chaque sous-bande est centrée sur un partiel du signal harmonique. Pour créer le spectre de synthèse, le modèle MBE utilise trois paramètres : la valeur du pitch, l'enveloppe spectrale $|H(\omega)|$ et les décisions de voisement des sous-bandes [Griffin et Lim, 1985]. Les prochaines parties du document décrivent comment le modèle obtient ces trois paramètres.

Création d'un spectre harmonique

Dans le modèle MBE, chaque sous-bande du spectre est centrée sur un partiel du spectre harmonique. Afin d'obtenir le spectre harmonique $|P(\omega)|$, le modèle crée un train d'impulsions dans le domaine temporel avec la valeur du pitch trouvée et lui applique ensuite une transformée de Fourier. La figure 2.13 [Griffin et Lim, 1985] montre l'exemple d'un spectre harmonique créé avec un train d'impulsions.

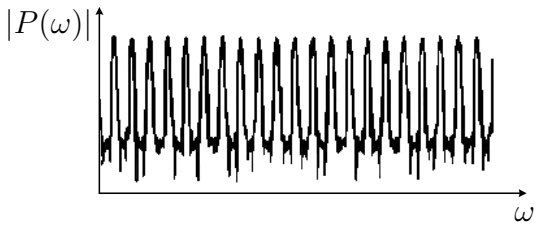


Figure 2.13 Exemple d'un spectre harmonique créé par le modèle MBE

Par la suite, le modèle calcule l'enveloppe spectrale du signal afin d'ajuster les amplitudes du spectre.

Calcul pour l'enveloppe spectrale

Pour obtenir les valeurs de l'enveloppe spectrale, le modèle minimise l'erreur entre le spectre du signal original $X(\omega)$ et le spectre du signal de synthèse $\hat{X}(\omega)$ avec l'équation 2.10 [Griffin et Lim, 1988]. Dans l'équation 2.10, la variable $|E(\omega)|$ représente une excitation de type harmonique $|P(\omega)|$ ou de type bruit blanc $|U(\omega)|$.

$$\varepsilon = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left[|X(\omega)| - |\hat{X}(\omega)|\right]^2 d\omega \qquad \text{où } |\hat{X}(\omega)| = |H(\omega)||E(\omega)| \tag{2.10}$$

La minimisation de l'erreur ε dans l'équation 2.10 avec tous les paramètres simultanément nécessite beaucoup de calculs. Afin de diminuer cette complexité, le modèle MBE effectue les calculs de l'erreur en sous-bande m. L'équation 2.11 [Griffin et Lim, 1988] montre les modifications effectuées sur l'équation 2.10 afin de calculer l'erreur en sous-bande m. Durant le calcul de l'erreur en sous-bande m, la valeur de l'enveloppe spectrale reste constante $|A_m|$.

$$\varepsilon_m = \frac{1}{2\pi}\int_{a_m}^{b_m}\left[|X(\omega)| - |A_m||E(\omega)|\right]^2 d\omega \tag{2.11}$$

$$a_m = \text{Limite inférieure de la sous bande } m$$
$$b_m = \text{Limite supérieure de la sous bande } m$$

L'erreur ε_m de l'équation 2.11 [Griffin et Lim, 1988] se minimise en posant que $|A_m|$ vaut

$$|A_m| = \frac{\int_{a_m}^{b_m} |X(\omega)||E(\omega)|\, d\omega}{\int_{a_m}^{b_m} |E(\omega)|^2\, d\omega} \tag{2.12}$$

La figure 2.14 [Griffin et Lim, 1988] montre le résultat des calculs de l'équation 2.12 sur un spectre complet.

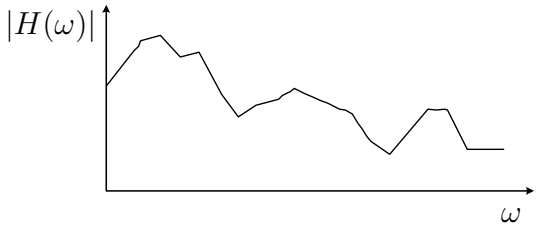


Figure 2.14 Enveloppe spectrale $|H(\omega)|$ à partir des valeurs calculées $|A_m|$

Par la suite, le modèle utilise les erreurs en sous-bande afin de déterminer les sous-bandes non-voisées.

Détection des sous-bandes voisées et non-voisées dans le spectre

Le modèle compare l'erreur ε_m obtenue de l'équation 2.11 avec un seuil préétabli pour déterminer le degré de voisement de la sous-bande. Les auteurs [Griffin et Lim, 1988] posent la valeur du seuil à 0.2 afin de déterminer si la sous-bande est déclarée voisée ou non (cf. équation 2.13).

$$\xi_m = \frac{\varepsilon_m}{\frac{1}{2\pi}\int_{a_m}^{b_m} |X(\omega)|^2 d\omega} \tag{2.13}$$

Si $\xi_m \leqslant 0.2$ sous-bande m voisée

Si $\xi_m > 0.2$ sous-bande m non-voisée

Lorsque le modèle déclare une sous-bande non-voisée, le modèle recalcule la valeur de l'amplitude $|A_m|$ de l'équation 2.12 avec du bruit blanc $|U(\omega)|$ comme excitation ($|E(\omega)| = |U(\omega)|$). La figure 2.15 [Griffin et Lim, 1988] montre un exemple de décisions prises par le

modèle avec le critère de l'équation 2.13. Dans la figure 2.15, un état à 1 représente une partie voisée tandis qu'un état à 0 indique une partie non-voisée.

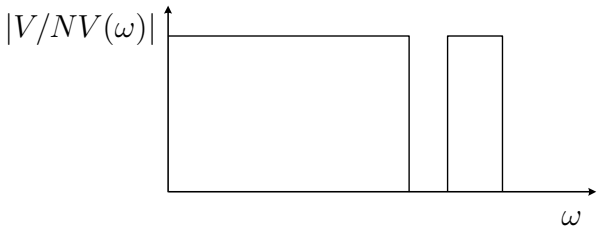


Figure 2.15 Décisions prises par le modèle MBE pour le voisement

La prochaine section donne l'exemple d'un spectre de synthèse créé avec les paramètres calculés précédemment.

L'exemple d'un signal de synthèse créé au décodeur

Pour la synthèse des signaux, le décodeur du modèle MBE reçoit comme paramètres : la valeur du pitch, l'enveloppe spectrale et les décisions de voisement. Avec la valeur du pitch, le modèle crée un spectre harmonique ainsi qu'un spectre de bruit (cf. figure 2.16 [Griffin et Lim, 1988]).

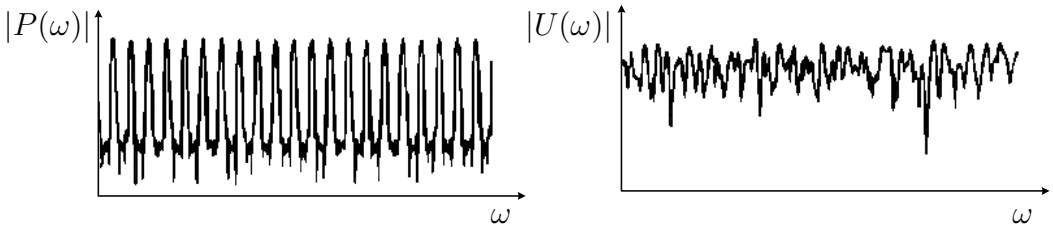


Figure 2.16 Excitations créées par le modèle MBE

Pour créer le signal d'excitation $|E(\omega)|$, le modèle insère un partiel ou du bruit blanc selon les résultats des décisions de voisement pour chaque sous-bande. La figure 2.17 [Griffin et Lim, 1988] montre le résultat du spectre d'excitation $|E(\omega)|$ obtenu à partir des décisions de la figure 2.15 et des excitations de la figure 2.16.

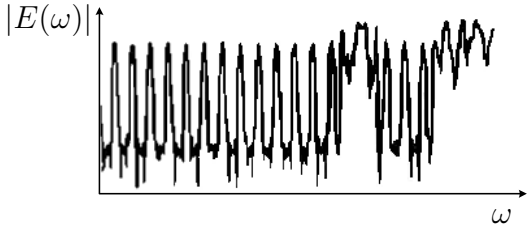


Figure 2.17 Signal d'excitation $E(\omega)$ contenant des partiels et du bruit blanc

Par la suite, avec l'équation 2.14 [Griffin et Lim, 1988], le modèle ajuste l'enveloppe du spectre d'excitation $|E(\omega)|$ avec les amplitudes calculées $|H(\omega)|$ de la figure 2.14. La figure 2.18 [Griffin et Lim, 1988] montre le spectre du signal de synthèse $|\hat{X}(\omega)|$ obtenu avec le modèle MBE et ainsi que le spectre du signal original $|X(\omega)|$.

$$|\hat{X}(\omega)| = |H(\omega)||E(\omega)| \tag{2.14}$$

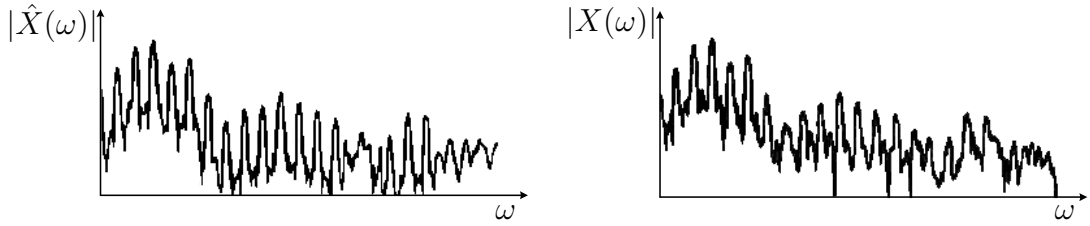


Figure 2.18 Spectre $\hat{X}(\omega)$ créé par le modèle MBE et le spectre original $X(\omega)$

Applications multiples du modèle de codage MBE

Les sections précédentes donnaient les détails de fonctionnement du modèle MBE pour la modélisation du signal de parole. Cependant, le modèle MBE ne se limite pas seulement à la compression des signaux de parole, il s'utilise également pour la modification du signal. Puisque le modèle estime l'enveloppe spectrale et l'excitation indépendamment, des applications peuvent modifier l'un de ces paramètres. Une modification possible consiste à changer la durée d'un signal de parole (*time scale modification*) ou bien la modification du pitch.

La prochaine section présente un second modèle paramétrique sinusoïdal appelé STC (***S**inusoidal **T**ransform **C**oding*). Comme pour le modèle MBE, le modèle STC s'utilise pour des applications autres que la modélisation du signal de parole.

2.2.2 STC (Sinusoidal Transform Coding) (1985)

Étudié dans les années 80, le modèle STC (***S**inusoidal **T**ransform **C**oding*) appartient également au groupe de codage paramétrique. C'est un modèle qui fonctionne autour de 4.8 kbit/s [McAulay et Quatieri, 1987] pour la compression des signaux de parole. Le modèle STC recherche toutes les composantes sinusoïdales du spectre. Contrairement au modèle MBE de la section précédente, le modèle STC n'utilise pas de classification des signaux, ce qui évite les erreurs de classification.

Le modèle STC utilise la représentation sinusoïdale pour tous les types de spectres : voisé, non-voisé et mixte. Pour trouver tous les sommets des sinusoïdes dans un spectre d'amplitudes, le modèle utilise la technique du *peak picking*. La figure 2.19 [McAulay et Quatieri, 1986] montre les sinusoïdes trouvées par le modèle STC avec un spectre voisé.

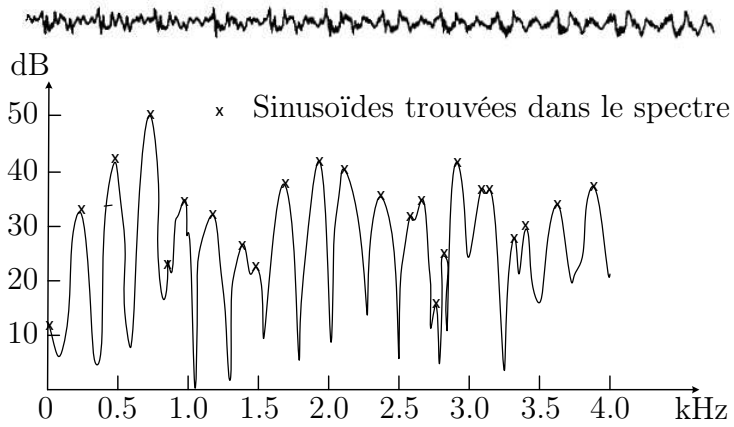


Figure 2.19 Exemple de sinusoïdes trouvées dans un spectre voisé

La figure 2.20 [McAulay et Quatieri, 1986] représente un exemple d'un spectre non-voisé avec les sinusoïdes trouvées par le modèle STC. La représentation d'un spectre non-voisé nécessite plus de sinusoïdes et le modèle STC limite le nombre de sinusoïdes à 100 afin de minimiser les calculs.

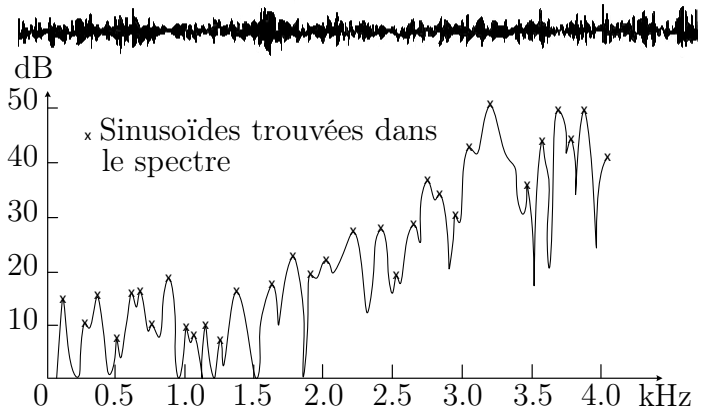


Figure 2.20 Exemple de sinusoïdes trouvées dans un spectre non-voisé

Les figures 2.21 et 2.22 [McAulay et Quatieri, 1986] montrent les schémas de fonctionnement du STC au niveau de l'encodeur et du décodeur.

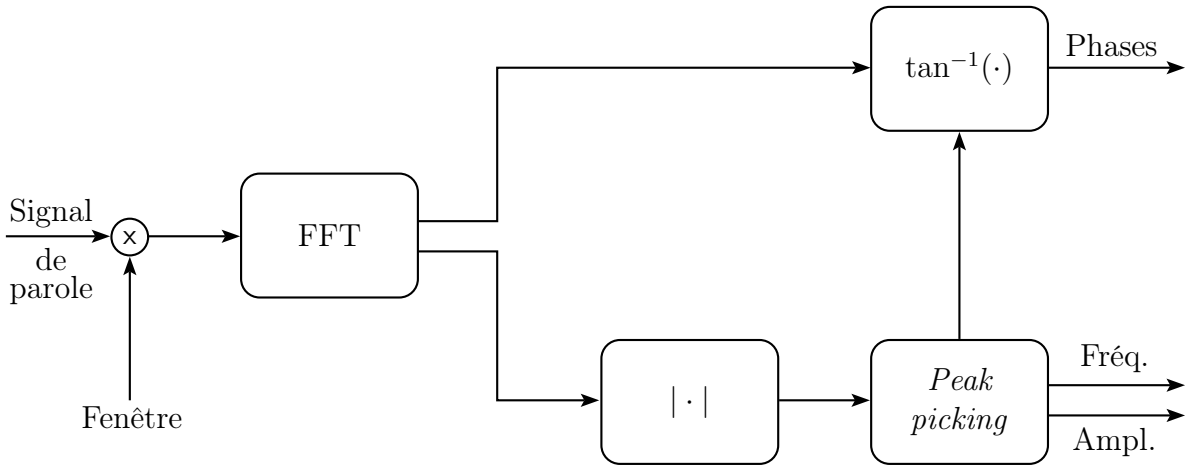


Figure 2.21 Schéma de fonctionnement à l'encodeur du modèle STC

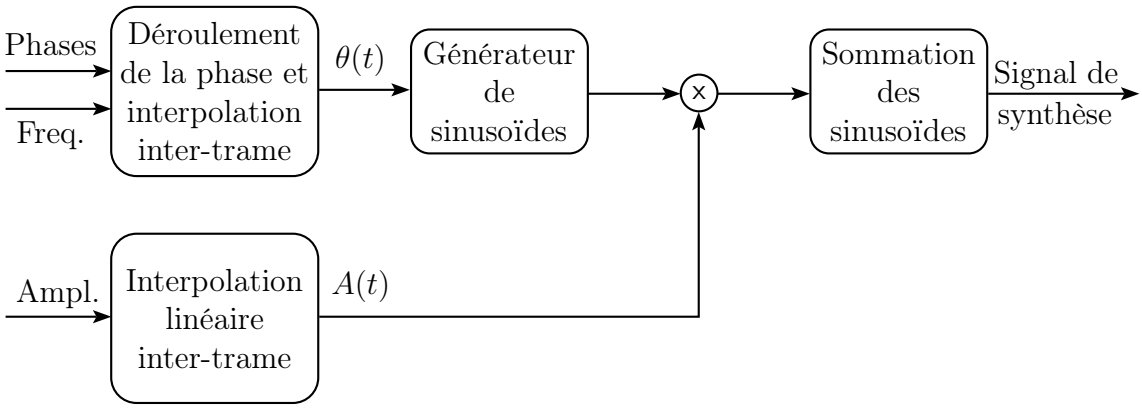


Figure 2.22 Schéma de fonctionnement au décodeur du modèle STC

L'équation 2.15 montre comment le modèle STC représente tous les types de signaux par une sommation de sinusoïdes. Pour obtenir le signal de synthèse, le modèle STC estime les valeurs des amplitudes A, des phases φ et des fréquences ω de toutes les sinusoïdes trouvées.

$$x(n) \approxeq \sum_{l=1}^{L} A_l \cos(\omega_l n + \varphi_l) \tag{2.15}$$

Puisque le nombre de sinusoïdes peut varier entre les trames et créer des discontinuités, les auteurs [McAulay et Quatieri, 1986] proposent le concept de naissances et de morts des

sinusoïdes. Avec ce concept, le modèle vérifie les continuités et les discontinuités possibles aux frontières des trames.

Pour chaque sinusoïde trouvée dans la trame précédente, le modèle vérifie si celle-ci se continue à la trame suivante. Le modèle considère également qu'une sinusoïde peut varier légèrement d'une trame à l'autre. Le modèle possède un seuil afin de vérifier la tolérance de variation qu'une sinusoïde peut subir entre deux trames. La figure 2.23 [McAulay et Quatieri, 1986] montre le résultat du concept de naissances et de morts des sinusoïdes sur quatre trames d'analyse.

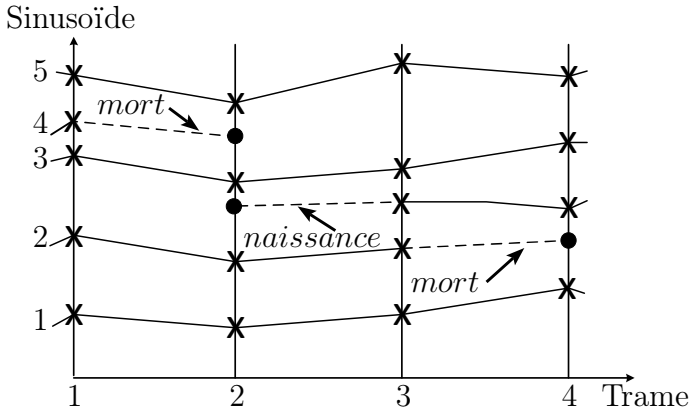


Figure 2.23 Exemple de naissances et de morts des sinusoïdes

La figure 2.24 [McAulay et Quatieri, 1986] montre un exemple d'évolution des sinusoïdes d'une trame non-voisée vers une trame voisée.

Comme pour le modèle MBE de la section précédente, le modèle STC possède également l'avantage de s'utiliser pour d'autres applications que la compression du signal de parole. Le modèle permet une modification de l'échelle du temps communément appelé le *time scaling* [McAulay et Quatieri, 1985]. Cette modification de l'échelle augmente ou diminue la vitesse du signal sans toutefois affecter son contenu fréquentiel comme le pitch. Le modèle STC permet également l'inverse, afin de modifier la valeur du pitch sans affecter la vitesse du signal de parole. De plus, une autre application possible se situe dans la séparation de source afin de discriminer plusieurs locuteurs [Quatieri et Danisewicz, 1990].

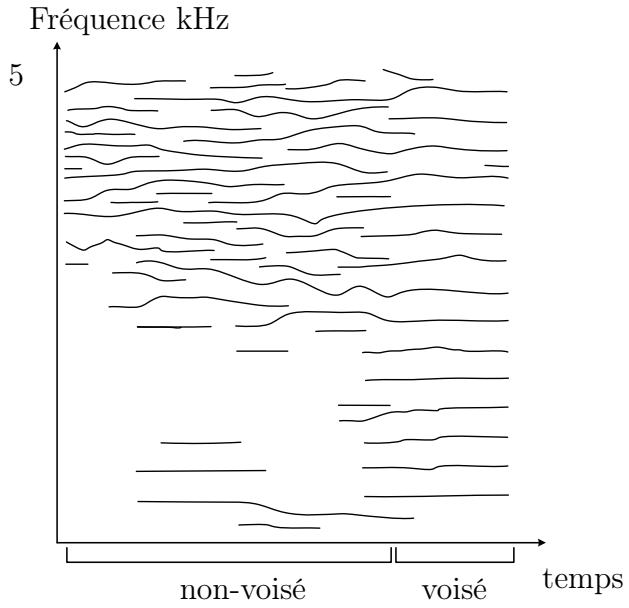


Figure 2.24 Évolution des sinusoïdes entre deux trames (de non-voisée à voisée)

2.2.3 Conclusion sur le codage paramétrique

Les sections précédentes décrivaient deux modèles importants de codage paramétrique de type sinusoïdal : le modèle MBE (***M**ulti**B**and **E**xcitation*) et le modèle STC (***S**inusoidal **T**ransform **C**oding*). Les modèles de codage paramétrique tentent d'extraire les caractéristiques importantes du signal de parole original afin d'utiliser ceux-ci pour créer un signal de synthèse.

Les modèles paramétriques représentent une alternative aux modèles temporels à forme d'onde de la section 2.1 pour la compression des signaux de parole à de plus bas débits. Ils fonctionnent à de plus faibles débits, mais ne réussissent toutefois pas à conserver une qualité audio comparable aux modèles temporels à forme d'onde. Les modèles paramétriques s'utilisent particulièrement pour des applications où le débit disponible est très limité, par exemple pour les communications téléphoniques par satellite.

Les modèles paramétriques présentés dans cette section possèdent l'avantage de s'utiliser pour des applications autre que la compression des signaux de parole. Ces modèles s'emploient également pour des modifications du signal comme par exemple le *time scaling* ou la modification du pitch.

Les sections précédentes décrivaient les modèles de codage paramétrique et à forme d'onde, deux approches de compression utilisées pour les signaux de parole. La prochaine section

décrit les modèles de codage perceptuel par transformée qui s'utilisent pour la compression des signaux audio.

2.3 Modèles de codage perceptuel par transformée

Les modèles perceptuels par transformée représentent l'unique approche à obtenir une bonne qualité audio pour les signaux audio à de faibles débits. Les modèles perceptuels de codage par transformée de première génération comme la norme MPEG-1 layer 3 possèdent des débits de 32 kbit/s à 320 kbit/s [Hardy *et al.*, 2002]. La seconde génération des modèles comme MPEG-2 AAC (***A****dvanced* ***A****udio* ***C****oding*) diminue de moitié le débit de la première génération tout en conservant une même qualité audio. C'est le groupe MPEG (***M****oving* ***P****icture* ***E****xperts* ***G****roup*) qui a proposé ces deux modèles de codage.

Le groupe MPEG provient d'une collaboration de deux organisations : l'organisme ISO (***I****nternational* ***O****rganization for* ***S****tandardization*) et le comité IEC (***I****nternational* ***E****lectro-technical* ***C****ommission*). Le groupe se compose d'experts du monde entier provenant de laboratoires universitaires et industriels. Le groupe établit des normes afin d'obtenir une interopérabilité du codage pour la représentation de différentes sources multimédias numériques. Le groupe MPEG développe également des algorithmes de compression afin de s'assurer d'une utilisation optimale de la bande passante. Cette section du document décrit deux modèles de codage perceptuel par transformée développés par le groupe MPEG : le modèle MP3 (MPEG-1 layer 3) et la famille des modèles AAC (MPEG-2 AAC et MPEG-4 AAC). Toutefois, avant de donner une description de chacun des modèles de codage, la prochaine section décrit des concepts communs utilisés par ceux-ci.

2.3.1 Concepts communs utilisés par les modèles MP3 et AAC

Cette section du document donne une description des concepts communs utilisés par les modèles de codage MP3 et AAC : l'utilisation de modèles psychoacoustiques et l'exploitation de phénomènes de masquage.

Utilisation de la modélisation psychoacoustique

Les modèles perceptuels par transformée utilisent la modélisation psychoacoustique afin de ne pas transmettre de l'information superflue que l'oreille humaine ne peut entendre. L'encodeur effectue une analyse du signal dans le domaine de la transformée afin de créer un modèle psychoacoustique qui décrit le comportement perceptuel de l'oreille humaine en fonction : de la fréquence, de l'intensité et du temps. Cette modélisation psychoacous-

tique tient compte des limites et des faiblesses de l'oreille humaine afin de transmettre uniquement les composantes essentielles du signal.

La figure 2.25 [Rossi, 2007] montre la courbe du seuil d'audition absolu. Cette courbe représente le niveau de pression acoustique moyen en dB (décibel) pour qu'un son sinusoïdal pur soit perçu par l'oreille humaine. Puisque chaque individu possède une courbe unique qui varie selon l'âge, la figure 2.25 représente le seuil d'audition absolu moyen.

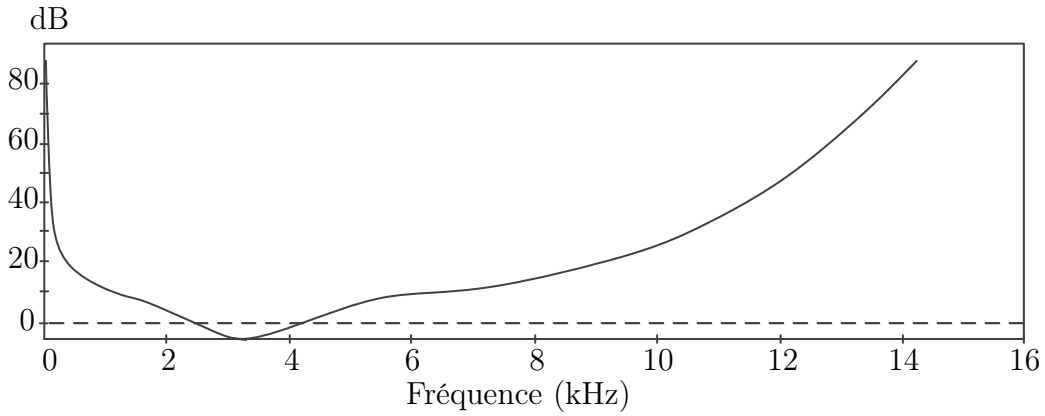


Figure 2.25 Courbe du seuil d'audition absolu de l'oreille humaine

La figure 2.25 montre que l'oreille humaine possède une plage de sensibilité à des fréquences allant de 20 Hz à 16 kHz et que le niveau de sensibilité diffère selon la fréquence. Le niveau de sensibilité maximale se situe autour de 1kHz à 5 kHz.

La courbe de la figure 2.25 correspond à une écoute dans un environnement calme. En présence de sons multiples, la courbe se modifie et le phénomène de masquage survient. Le phénomène se produit lorsqu'un son empêche la perception d'un autre son qui autrement serait audible. Les modèles perceptuels par transformée exploitent ces phénomènes de masquage afin de réduire l'information à transmettre.

Exploitation des phénomènes de masquage

Il existe deux types de phénomènes de masquage : l'un temporel et l'autre fréquentiel. Le phénomène de masquage temporel provient de l'inertie temporelle du système d'audition tandis que le phénomène de masquage fréquentiel provient du comportement en fréquence de la membrane basilaire de la cochlée au sein de l'oreille interne.

Le masquage temporel se produit avant et après l'apparition d'un son masquant de forte intensité (cf. figure 2.26 [Spanias *et al.*, 2007]). Après un son à fort décibel, l'oreille ne perçoit pas un son à plus faible intensité qu'après d'un certain laps de temps. Le pré-

masquage reste quelques millisecondes (2 ms à 5 ms) tandis que le post-masquage dure plus longtemps (100 ms à 200 ms).

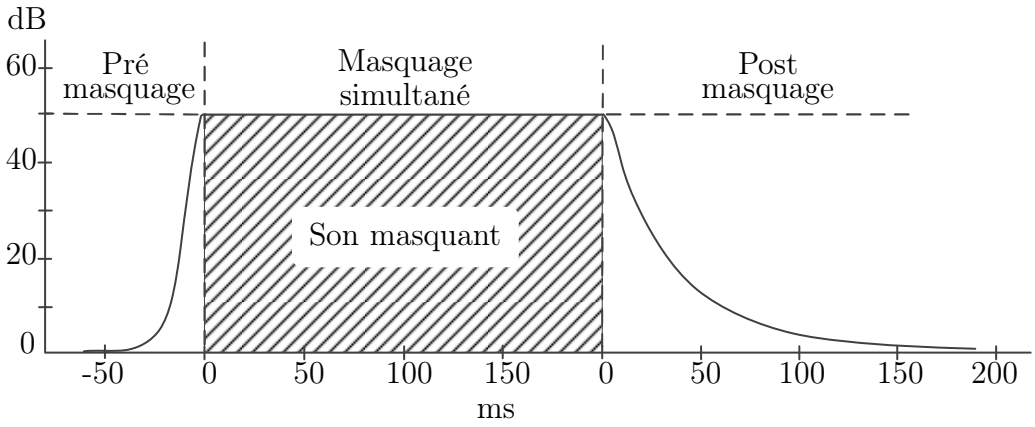


Figure 2.26 Masquage temporel

Le phénomène de masquage fréquentiel survient lorsqu'une raie fréquentielle de forte intensité dissimule les fréquences voisines de plus faibles intensités (cf. figure 2.27 [Rossi, 2007]). La fréquence et l'intensité du signal possèdent une influence sur les caractéristiques du masque.

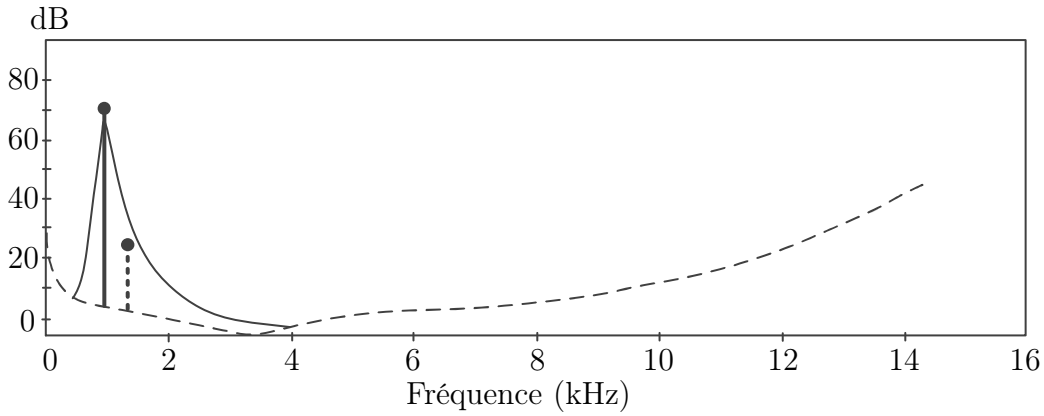


Figure 2.27 Masquage fréquentiel

La figure 2.28 [Brandenburg et Chiariglione, 2003] montre que l'intensité du signal modifie la courbe de masquage sur la largeur du masque.

De plus, la figure 2.29 [Rossi, 2007] montre que l'oreille possède une plus grande sensibilité en basse fréquence ce qui explique des bandes plus étroites. L'oreille interne se comporte comme un filtre passe-bande psychoacoustique centré sur une fréquence. Les largeurs de bande de la figure 2.29 représentent la largeur de bande des filtres auditifs de l'oreille

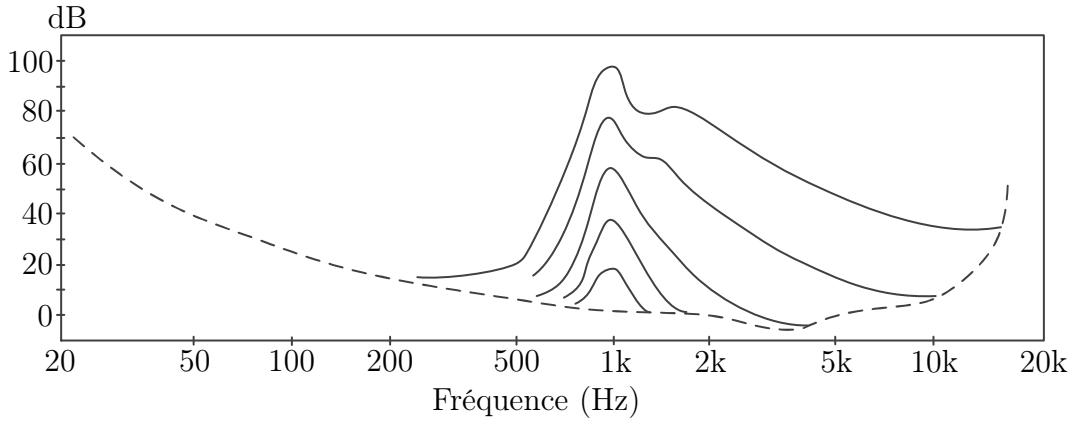


Figure 2.28 Largeur des masques selon l'intensité

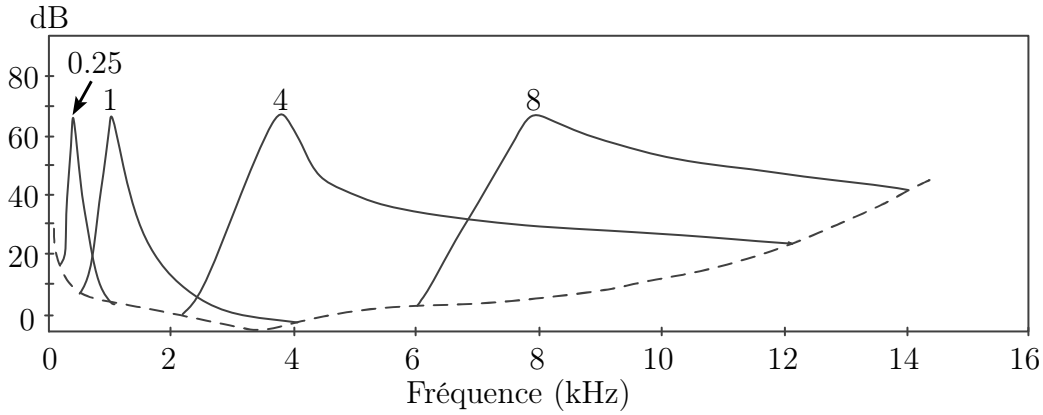


Figure 2.29 Largeur des masques selon la valeur de la fréquence

interne (la cochlée). Ainsi, l'oreille interne analyse les sons complexes comme un banc de filtres passe-bande.

La figure 2.29 montre que les masques possèdent des largeurs différentes selon la valeur de la fréquence du son pur et qu'elles s'élargissent vers les fréquences plus élevées. Cette évolution non-linéaire des bandes critiques complique le calcul du masque psychoacoustique. Afin de simplifier les calculs, une transformation de linéarité s'effectue sur les bandes critiques afin de les transposer sur l'échelle bark au lieu de l'unité des hertz.

Linéarisation des largeurs des bandes critiques

L'échelle bark introduit une linéarité pour les bandes critiques dans le spectre. Ainsi, quelle que soit la largeur de la bande critique elle mesurera toujours 1 bark. La figure 2.30 montre les mêmes masques de la figure 2.29 [Rossi, 2007], mais avec l'échelle bark au lieu des fréquences.

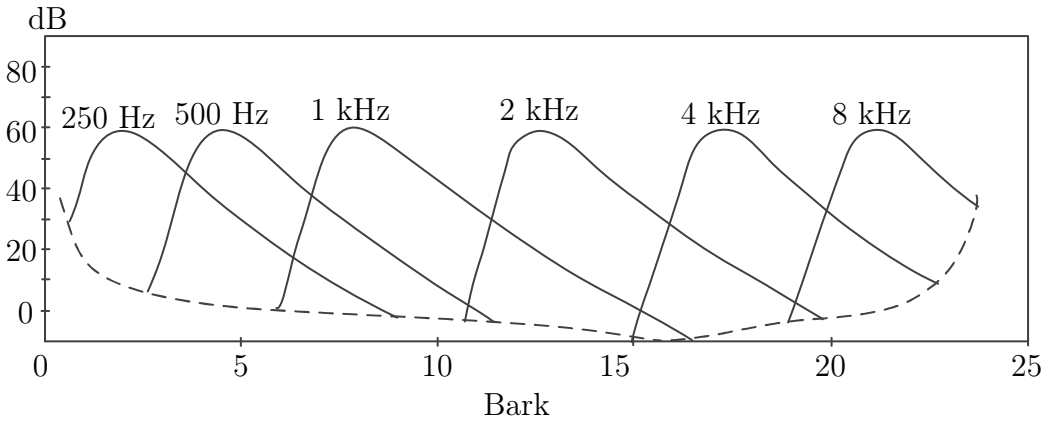


Figure 2.30 Largeur des masques avec une échelle bark

La figure 2.31 [Pan, 1995] montre les bandes critiques en fréquence et transposées en échelle bark utilisée dans les modèles de codage MPEG.

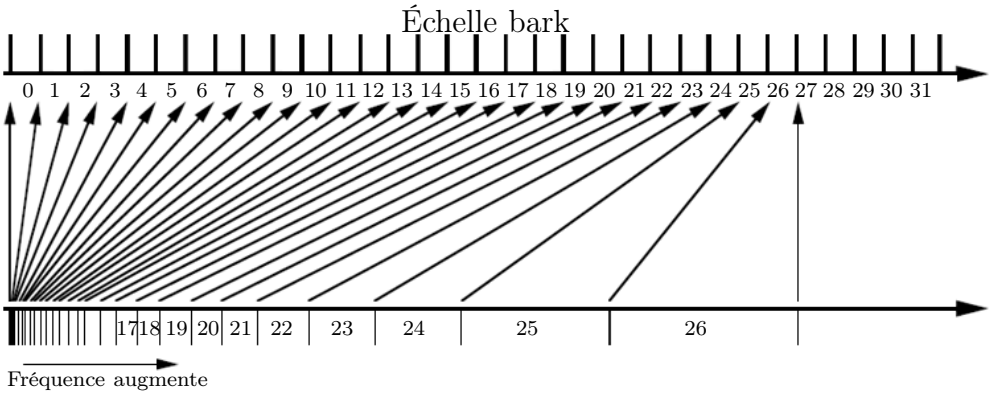


Figure 2.31 Largeur des bandes critiques en hertz et en bark

Cette section a décrit la modélisation psychoacoustique et les phénomènes de masquage souvent utilisés par les modèles perceptuels par transformée. Les modèles perceptuels utilisent ces concepts afin de créer le modèle psychoacoustique qui définit le seuil de masquage et par conséquent la limite d'audibilité. Le modèle psychoacoustique contient un ensemble de règles qui indique les sons qui seront masqués. Les modèles de codage MP3 et AAC utilisent ces règles afin d'éliminer l'information jugée superflue à ne pas transmettre. Les prochaines sections du document décrivent les principes de fonctionnement du modèle MPEG-1 layer 3 et des modèles de la famille AAC.

2.3.2 MPEG-1 Layer 3 : MP3 (1993)

Le modèle MPEG-1 layer 3 est souvent appelé MP3 dû à l'extension du fichier « .mp3 » de synthèse qu'il crée. Il appartient au premier standard audio proposé par le groupe MPEG en 1993 [ISO/IEC-11172-3, 1993]. Le modèle MPEG-1 layer 3 appartient au standard MPEG-1 audio qui contient trois modes d'opération appelés couches ou layers.

Le taux de compression et le niveau de complexité augmentent du premier layer au troisième layer. Le standard MPEG-1 audio traite tous les types de signaux ayant des fréquences d'échantillonnage de 32 kHz, 44.1 kHz et 48 kHz. La fréquence d'échantillonnage par défaut est de 44.1 kHz. Les signaux peuvent être monophoniques ou stéréophoniques. L'encodage des signaux stéréophoniques s'effectue sous la forme double monophonique (deux canaux indépendants), stéréophonique ou stéréophonique jointe. Pour l'encodage stéréophonique jointe, l'encodeur exploite les corrélations entre les canaux afin de réduire le débit.

Le tableau 2.2 montre les débits et le niveau de complexité de chaque layer du standard MPEG-1 audio [Rivier, 2003][Hardy *et al.*, 2002]. Il donne également un aperçu des caractéristiques importantes de chaque layer du standard. Le tableau montre que le troisième layer possède un taux de compression plus élevé ainsi qu'un niveau de complexité plus grand comparativement aux layers inférieurs.

Tableau 2.2 Caractéristiques des trois layers du standard MPEG-1 audio

	Complexité	Plage de débits	Caractéristiques
Layer 1	basse	32 à 448 kbit/s	Application d'un banc de filtres Quantification uniforme Utilisation de masques fréquentiels
Layer 2	moyenne	32 à 384 kbit/s	Application d'un banc de filtres Quantification uniforme Utilisation de masques fréquentiels et temporels
Layer 3	élevée	32 à 320 kbit/s	Application d'un banc de filtres et d'une transformée MDCT Quantification non-uniforme Utilisation de masques fréquentiels et temporels Utilisation du codage de Huffman

Ce document décrit en particulier le troisième layer du standard MPEG-1 audio, le modèle de codage perceptuel par transformée MPEG-1 layer 3. Afin de mieux comprendre le fonctionnement du modèle MPEG-1 layer 3, la figure 2.32 propose le schéma de fonction-

nement général d'un modèle de codage perceptuel par transformée. Le modèle général de la figure 2.32 se décompose en trois modules importants : la segmentation du signal en sous-bande, la modélisation psychoacoustique et la quantification/encodage.

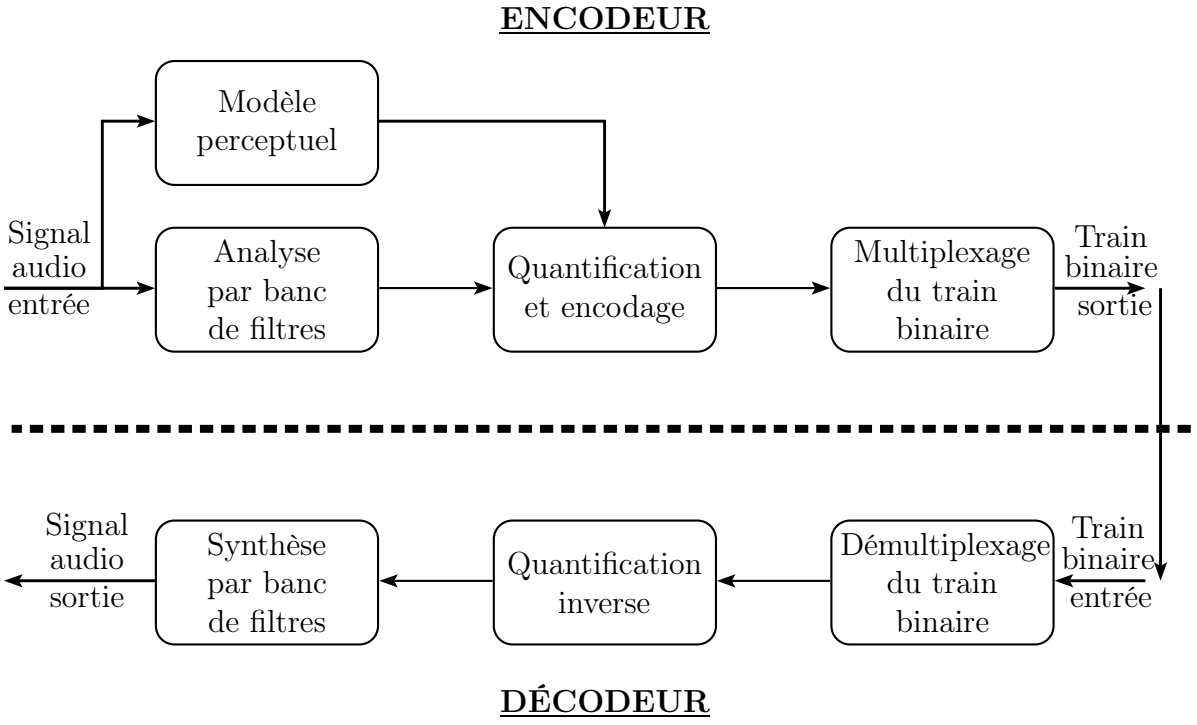


Figure 2.32 Schéma général d'un modèle de codage perceptuel par transformée

Le modèle de codage MPEG-1 layer 3 possède un schéma de fonctionnement (cf. figure 2.33 [Brandenburg, 1999]) plus complexe que celui de la figure 2.32, mais il se décompose également en trois modules importants. Les prochaines parties du document décrivent le fonctionnement des trois modules : la segmentation du signal en sous-bande, la modélisation psychoacoustique et la quantification/encodage pour le modèle MPEG-1 layer 3.

Banc de filtres et transformation MDCT (Modified Discrete Cosine Transform)

Comme mentionné précédemment, le codage par transformée possède trois grands modules dont le premier vise à segmenter le signal en sous-bandes fréquentielles. Le troisième layer du standard MPEG-1 utilise comme les layers précédents un banc de filtres sur le signal audio d'entrée. Le banc de filtre sépare le signal en 32 sous-bandes d'égales largeurs fréquentielles, mais contrairement aux layers précédents le modèle du layer 3 applique également une transformée MDCT (***M**odified **D**iscrete **Co**sine **T**ransform*) sur chaque sous-bande. La transformée MDCT décompose le signal sur une base de cosinus.

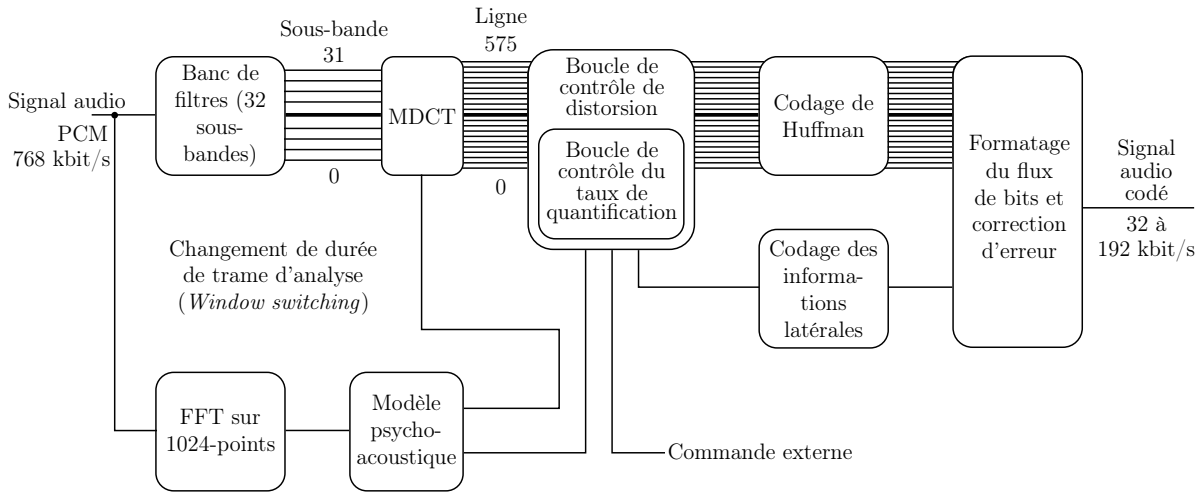


Figure 2.33 Schéma de fonctionnement à l'encodeur du modèle MPEG-1 layer 3

Les transformées utilisées dans le modèle possèdent différentes longueurs de 6-points ou 18-points. Le modèle choisit la longueur de la transformée selon la résolution nécessaire pour chaque sous-bande. Ces différentes longueurs adaptent la résolution en fréquence afin d'obtenir une meilleure approximation des bandes critiques. De plus, le modèle possède également des trames d'analyse de longueurs variables pour traiter les transitions sonores rapides et prévenir la génération de pré-échos. Le nombre maximal de subdivisions est de 576 (32 sous-bandes x 18-points pour la transformée MDCT) [Rivier, 2003].

Modélisation perceptuelle

Parallèlement au banc de filtres et à la transformée MDCT, le modèle applique une transformée de Fourier de 1024-points sur le signal d'entrée afin d'obtenir son spectre. C'est sur ce spectre que le modèle détermine les courbes de masquage pour chacune des sous-bandes à l'aide de leur modèle psychoacoustique.

Quantification et codage entropique

Lors de la phase de quantification, le modèle utilise deux boucles d'itérations imbriquées : une boucle extérieure de contrôle de distorsion et une boucle intérieure de taux de quantification [Brandenburg, 1999]. La boucle extérieure maintient le bruit de quantification sous un seuil de masquage alors que la boucle intérieure tente de parvenir au débit désiré.

La boucle extérieure quantifie et encode les coefficients des transformées MDCT de chaque sous-bande. Pour minimiser les erreurs, le modèle utilise une quantification non-uniforme sur les coefficients de la transformée MDCT. Le modèle donne un nombre de bits plus élevé pour les zones où l'oreille possède une grande sensibilité et réduit le nombre de bits pour les zones moins perceptibles. Si le bruit se situe au-dessus du seuil, la boucle modifie le nombre alloué de bits et relance la boucle extérieure.

Les données quantifiées sont ensuite compressées avec un encodage entropique dans la boucle intérieure. Le modèle utilise un encodage entropique sans perte de type Huffman qui se base sur des tables. Le codage de Huffman réduit la redondance dans la suite des données numériques tout en n'introduisant aucune perte de données. Le modèle MPEG-1 layer 3 dispose de différentes tables de Huffman qui s'utilisent selon la position dans le spectre à encoder. Le modèle utilise des tables différentes dans trois régions spectrales afin de tenir compte des particularités statistiques des signaux spectraux [Rivier, 2003].

La boucle intérieure se poursuit jusqu'à ce que soit respecté le volume maximal de données autorisées en fonction du débit demandé. Si la boucle détecte un signal trop complexe à encoder pour le nombre de bits disponibles, elle éliminera certaines bandes de fréquences. L'efficacité de l'encodage entropique reste modeste au niveau de la compression, mais s'ajoute aux techniques développées dans ce système afin d'obtenir un taux de compression le plus élevé possible. Les deux boucles se répètent jusqu'à ce que le bruit de quantification se situe sous un seuil de masquage et que le débit demandé soit atteint.

Le modèle MPEG-1 layer 3 audio décrit dans cette section représente la première génération de modèle par transformée proposée par le groupe MPEG. La prochaine section décrit la famille de codage AAC (***A****dvanced* ***A****udio* ***C****oding*) qui représente la deuxième et la quatrième génération de modèle MPEG. La deuxième génération diminue de moitié le débit de la première génération (MPEG-1 audio) tout en conservant une même qualité audio.

2.3.3 Famille AAC (Advanced Audio Coding) de MPEG (1997)

Le modèle AAC appartient à la deuxième (MPEG-2) et à la quatrième (MPEG-4) génération de modèle de codage audio MPEG. Le standard MPEG-2 se divise en deux phases dont la première phase appelée MPEG-2 BC (MPEG-2 ***B****ackward* ***C****ompatible*) est compatible avec le standard MPEG-1 audio. La deuxième phase se nomme MPEG-2 NBC (***N****on-****B****ackward* ***C****ompatible*) qui est non rétrocompatible.

Brève description du modèle MPEG-2 BC (MPEG-2 Backward Compatible)

La normalisation du modèle MPEG-2 BC s'est finalisée en 1994. Cette norme n'inclut pas de nouvelles techniques de codage, mais ajoute deux nouvelles extensions à la norme MPEG-1 audio : le codage en multicanal 5.1 et le codage à des fréquences d'échantillonnage plus faibles (16 kHz, 22.05 kHz et 24 kHz). Ces fréquences d'échantillonnage plus faibles permettent au modèle d'atteindre une meilleure efficacité à des débits plus bas.

Le modèle MPEG-2 BC reste compatible de façon descendante et ascendante avec les différents layers de la norme MPEG-1 audio. Un codec multicanal MPEG-2 peut décoder des signaux MPEG-1 monophonique ou stéréophonique, et inversement. Ainsi, les modèles MPEG-1 audio qui ne disposent que de deux canaux peuvent reproduire un signal stéréo de base à partir d'un flux numérique du modèle MPEG-2 BC multicanal.

Famille des modèles AAC (*Non-Backward Compatible*)

C'est au début de 1994 que des tests démontrent que l'introduction d'un nouvel algorithme de compression pouvait grandement améliorer les performances de codage. C'est également le début de la seconde phase nommée MPEG-2 AAC qui est non rétrocompatible avec les modèles de compression précédents. Cette deuxième phase s'est terminée en 1997 par la création d'un standard [ISO/IEC-13818-7, 1997]. Le standard MPEG-2 AAC représente un nouveau modèle qui procure deux fois plus d'efficacité que le standard MPEG-1 audio et le modèle MPEG-2 BC. La figure 2.34 [Bosi *et al.*, 1997] montre le schéma de fonctionnement à l'encodeur du standard MPEG-2 AAC.

Le modèle MPEG-2 AAC améliore la restitution multicanal et augmente considérablement les performances de débits par rapport au modèle MPEG-1 layer 3. La restitution des canaux s'effectuait déjà avec le modèle MPEG-2 BC, mais le modèle MPEG-2 AAC augmente les possibilités du multicanal. Avec le modèle MPEG-2 AAC, il est possible d'obtenir 48 canaux en pleine résolution, 16 canaux en basse fréquence et 16 flux de données avec des fréquences d'échantillonnage de 8 kHz à 96 kHz. Pour des sons stéréophoniques, le modèle MPEG-2 AAC possède une qualité perceptuelle presque transparente pour des débits de 96 kbit/s à 128 kbit/s [ISO/IEC/JTC1/SC29/WG11, 1998].

Les performances du modèle MPEG-2 AAC proviennent d'amélioration et d'intégration de nouvelles techniques dans le standard. Le modèle MPEG-2 AAC améliore l'encodage entropique et perfectionne le codage stéréophonique par rapport au standard MPEG-1 audio. Le modèle intègre également un algorithme de mise en forme du bruit dans le domaine temporel (TNS, ***T****emporal* ***N****oise* ***S****haping*) ce qui permet d'améliorer la restitution du bruit [Herre et Johnston, 1996]. L'algorithme TNS permet de contrôler la dispersion tem-

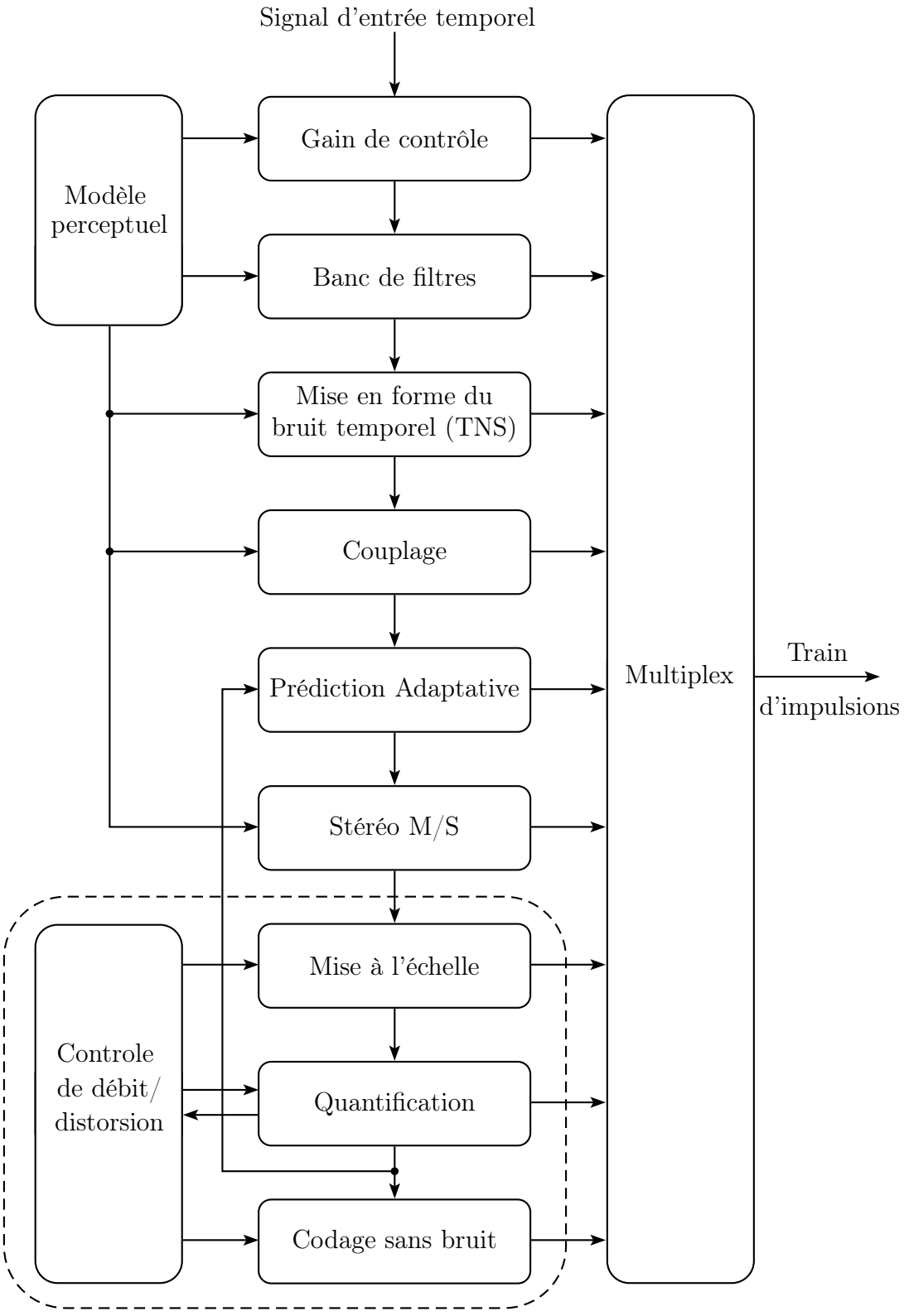
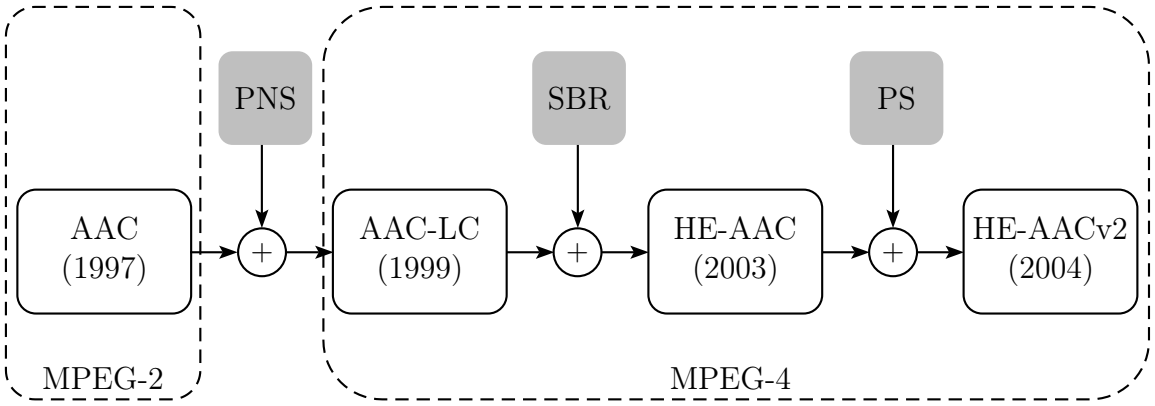


Figure 2.34 Schéma de fonctionnement à l'encodeur du modèle MPEG-2 AAC

porelle du bruit de quantification afin de s'assurer que sa forme temporelle s'adapte à la distribution énergétique du signal d'entrée.

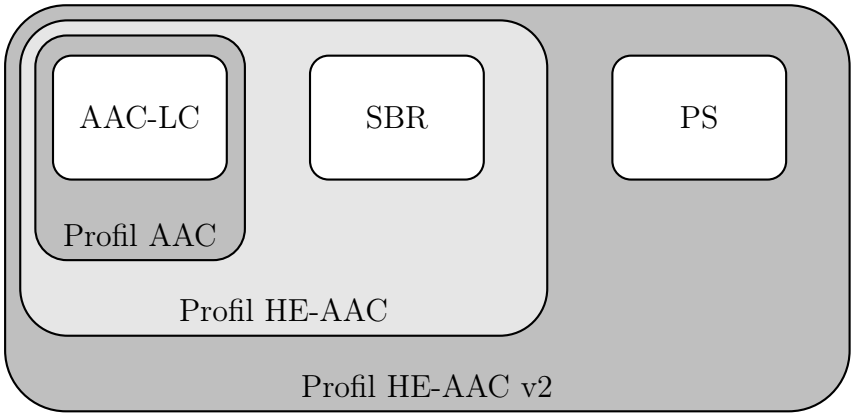
De plus, toujours en 1994, le groupe MPEG fait un appel à propositions pour la quatrième génération de modèle de codage appelé MPEG-4. Le groupe décide d'intégrer le modèle AAC dans le standard MPEG-4 audio à la fin de l'année de 1998 [ISO/IEC-14496-3, 2009]. Le standard MPEG-4 audio ajoute de nouvelles fonctionnalités et fonctionne à des débits plus faibles, sous les 64 kbit/s par canal [Herre et Grill, 2000]. La figure 2.35 [FraunhoferIIS, 2012] montre l'évolution des différents modèles AAC du standard MPEG-2 audio au standard MPEG-4 audio.



AAC, ***A**dvanced **A**udio **C**oding*
AAC-LC, ***A**dvanced **A**udio **C**oding - **L**ow **C**omplexity*
HE-AAC, ***H**igh **E**fficiency - **A**dvanced **A**udio **C**oding*
HE-AAC v2, ***H**igh **E**fficiency - **A**dvanced **A**udio **C**oding **V**ersion 2*
PNS, ***P**erceptual **N**oise **S**ubstitution*
SBR, ***S**pectral **B**and **R**eplication*
PS, ***P**arametric **S**tereo*

Figure 2.35 Versions AAC des standards MPEG-2 audio et MPEG-4 audio

Dans le standard MPEG-4 audio, le modèle AAC possède différents profils avec des niveaux hiérarchiques (cf. figure 2.36 [ISO/IEC14496-3, 2005]). Le niveau hiérarchique le plus élevé lit des fichiers encodés avec un codeur d'un niveau inférieur. Par exemple, le modèle HE-AAC peut décoder un fichier du profil AAC, mais ne peut pas décoder un fichier provenant du profil HE-AAC v2. Le profil HE-AAC v2 de la figure 2.36 représente le profil le plus performant de la famille des modèles AAC, il possède un débit de 24 kbit/s à 32 kbit/s pour un signal stéréophonique.



AAC,	***A**dvanced **A**udio **C**oding*
AAC-LC,	***A**dvanced **A**udio **C**oding - **L**ow **C**omplexity*
HE-AAC,	***H**igh **E**fficiency - **A**dvanced **A**udio **C**oding*
HE-AAC v2,	***H**igh **E**fficiency - **A**dvanced **A**udio **C**oding **V**ersion 2*
SBR,	***S**pectral **B**and **R**eplication*
PS,	***P**arametric **S**tereo*

Figure 2.36 Profils hiérarchiques des versions AAC dans le standard MPEG-4 audio

Les prochaines parties décrivent brièvement les fonctionnalités ajoutées aux différents modèles de la famille AAC de la figure 2.35 : la substitution perceptuelle de bruit (PNS, ***P**erceptual **N**oise **S**ubstitution*), la reconstruction de la bande spectrale (SBR, ***S**pectral **B**and **R**eplication*) et la stéréophonie paramétrique (PS, ***P**arametric **S**tereo*).

Substitution perceptuelle de bruit (PNS, *Perceptual Noise Substitution*)

La figure 2.37 [Herre et Schultz, 1998] montre, avec les encadrés en gris, le fonctionnement de la méthode de substitution perceptuelle du bruit. La méthode de substitution perceptuelle de bruit ne transmet que quelques paramètres afin de reproduire certaines parties de bruit du spectre.

Ainsi, au lieu de transmettre les coefficients spectraux d'une sous-bande avec des caractéristiques de bruit, l'encodeur envoie un indicateur afin d'annoncer une substitution de bruit et de l'énergie de cette sous-bande. Au niveau du décodeur, le modèle crée du bruit avec l'énergie indiquée et l'insère dans le spectre pour représenter les coefficients spectraux.

Reconstruction de la bande spectrale (SBR, *Spectral Band Replication*)

La technique SBR se base sur la corrélation existante entre le haut (haute fréquence) et le bas du spectre (basse fréquence). La figure 2.38 [Meltzer et Moser, 2006] montre comment

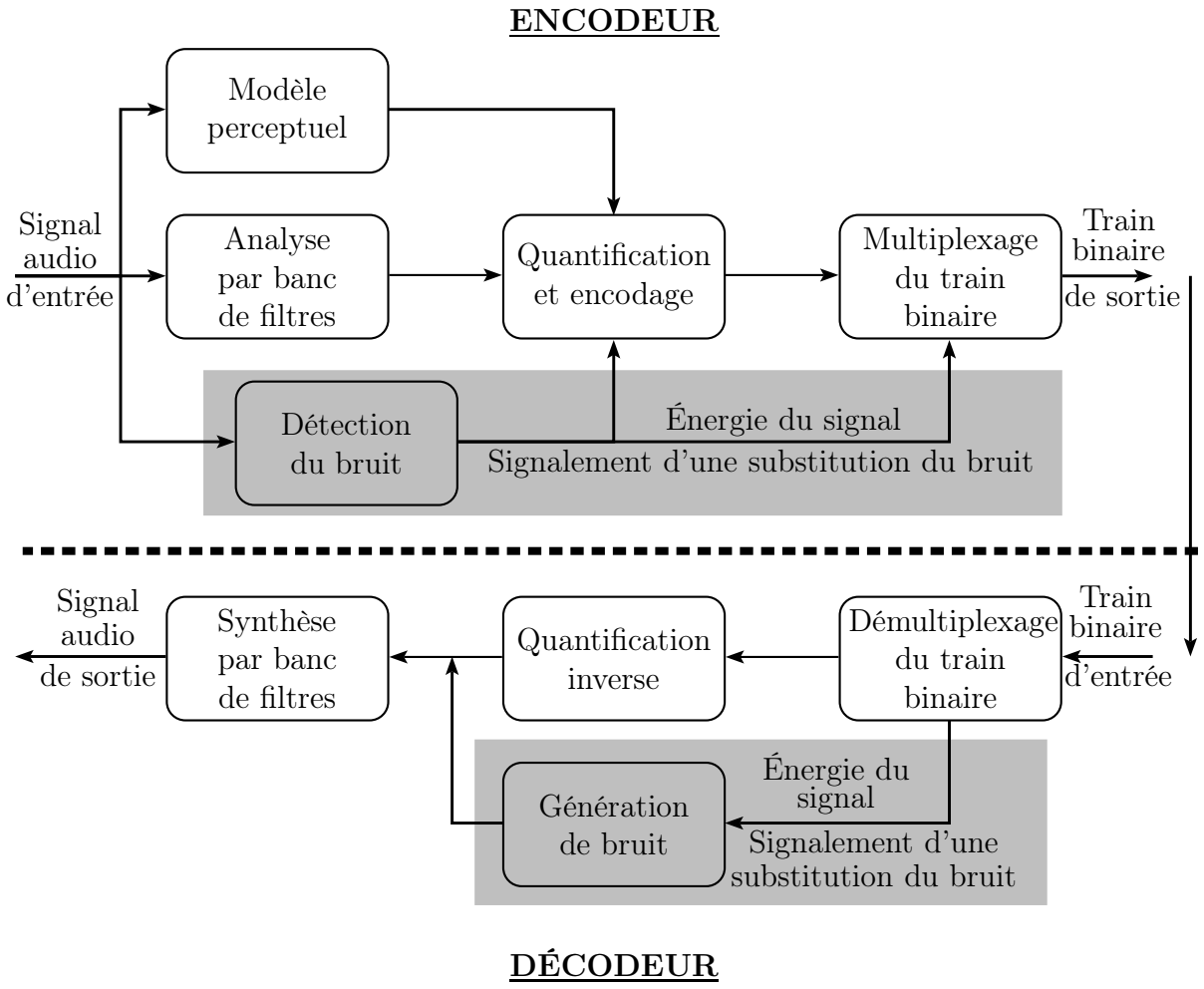
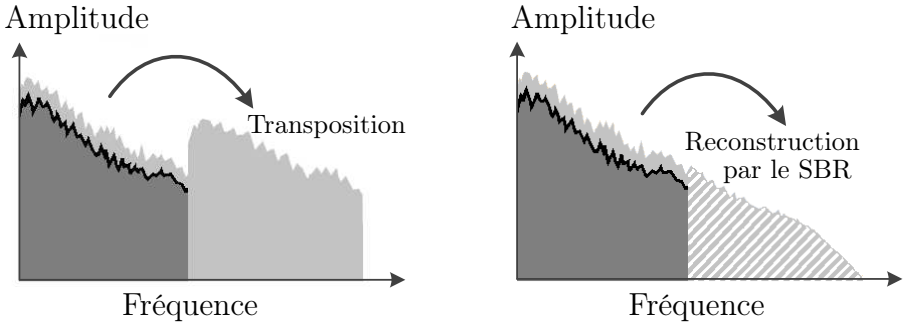


Figure 2.37 Principe de la substitution perceptuelle de bruit (PNS)

l'algorithme transpose une certaine partie du bas du spectre vers le haut. La technique SBR modifie la bande copiée pour que la copie ressemble le plus possible à l'enveloppe du au haut du spectre original. La reconstruction au décodeur des hautes fréquences avec la technique SBR nécessite peu de débit soit de 1 kbit/s à 3 kbit/s [Meltzer et Moser, 2006].

Stéréophonie paramétrique (PS, *Parametric Stereo*)

La stéréophonie paramétrique améliore la compression des signaux à bas débit. Cette technologie est optimisée pour une utilisation sur une plage de débit entre 16 kbit/s et 40 kbit/s. Elle permet d'obtenir une qualité audio à un débit autour de 24 kbit/s. L'encodeur utilise la stéréophonie paramétrique afin d'extraire l'image stéréophonique du signal audio et transmet uniquement que la représentation monophonique du signal au décodeur (cf. figure 2.39 [Meltzer et Moser, 2006]).



a) Reconstruction des hautes fréquences par transposition

b) Ajustement de l'enveloppe dans les hautes fréquences

Figure 2.38 Principe de la reconstruction de la bande spectrale (SBR)

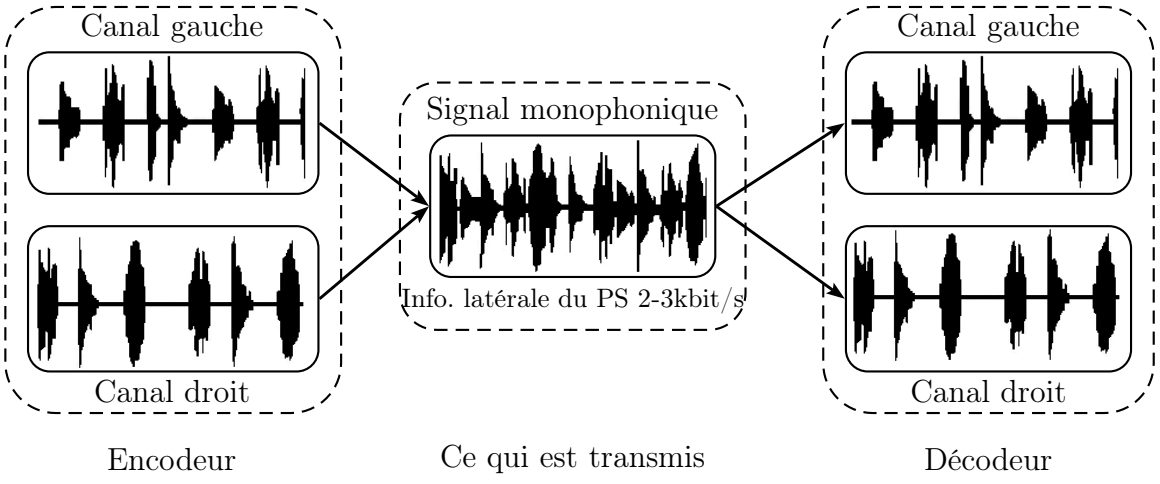


Figure 2.39 Principe de la stéréophonie paramétrique (PS)

Comme pour la technique SBR présentée précédemment, l'information de la stéréophonie paramétrique représente une petite quantité d'information à transmettre 2 kbit/s à 3 kbit/s qui s'ajoute au signal monophonique afin de créer un signal stéréo de qualité. L'utilisation de la stéréophonie paramétrique augmente la qualité du signal audio lors de conditions de faible débit.

2.3.4 Conclusion sur le codage perceptuel par transformée

Les sections précédentes présentaient le modèle MPEG-1 layer 3 et la famille des modèles AAC (***A****dvanced* ***A****udio* ***C****oding*) qui appartiennent au groupe de codage perceptuel par transformée. Les modèles de codage perceptuel par transformée représentent l'unique approche de la figure 2.1 qui compresse les signaux audio avec une bonne qualité à faible

débit. Lors de conditions de faibles débits, aucun modèle de codage de la figure 2.1 ne réussit à obtenir une qualité uniforme pour tous les types de signaux audio. Actuellement, l'unique moyen de compresser tous les signaux audio consiste en un codec hybride universel. La prochaine section décrit un modèle hybride universel appelé USAC (***U**nified **S**peech and **A**udio **Co**ding*) proposé par le groupe MPEG.

2.4 Modèle de codage hybride universel

Le modèle de codage hybride représente actuellement l'unique moyen d'obtenir un codec universel pour la compression de tous les types de signaux audio à faible débit. Un codec hybride intègre au moins deux modèles de codage et un classificateur qui sélectionne le modèle à exécuter selon le type de signal à traiter. Cette section décrit le standard USAC (***U**nified **S**peech and **A**udio **Co**ding*) proposé par le groupe MPEG (***M**oving **P**icture **E**xperts **G**roup*).

2.4.1 MPEG-D Part 3 : USAC (Unified Speech and Audio Coding) (2012)

C'est en octobre 2007 que le groupe MPEG publie un appel de propositions afin d'obtenir un modèle universel qui compresse tous les types de signaux audio [JTC1/SC29/WG11, 2007]. Le futur modèle universel s'utilisera pour des applications multimédias diffusées en continu (*streaming*) sur des appareils mobiles. Le modèle devra fonctionner à de faibles débits et obtenir une meilleure qualité que le meilleur des standards existants sur chaque type de contenu c'est-à-dire les signaux de parole et les signaux audio.

C'est en 2008 que le groupe MPEG sélectionne le codec USAC (***U**nified **S**peech and **A**udio **Co**ding*) comme modèle de référence 0 (RM0, ***R**eference **M**odel **0***) afin d'obtenir un modèle universel audio. Après cette sélection, une collaboration intensive s'effectue durant quatre années avec plusieurs laboratoires afin d'améliorer la référence RM0. Cette collaboration se termine en 2012 avec un standard [ISO/IEC-23003-3, 2012].

La combinaison de deux modèles de codage

Comme mentionné précédemment, le standard USAC représente un exemple d'un codec hybride universel qui contient plusieurs modèles de codage. Ainsi, le standard USAC intègre une version du standard MPEG-4 HE-AAC v2 et une version du standard AMR-WB+ (*Extended **A**daptive **M**ulti **R**ate - **W**ide**B**and*) [Neuendorf *et al.*, 2009].

La section 2.3.3 de ce chapitre présentait les caractéristiques du standard HE-AAC v2. De plus, la section 2.1.3 décrivait également le modèle de codage ACELP (***A**lgebraic **C**ode-**E**xcited **L**inear **P**rediction*) dont l'AMR-WB+ intègre une version.

La figure 2.40 [Neuendorf *et al.*, 2009] montre le schéma général de fonctionnement du standard USAC (version RM0). Le schéma montre que le standard contient deux modes distincts pour le traitement des signaux audio et qu'il possède un classificateur afin de sélectionner le mode à exécuter.

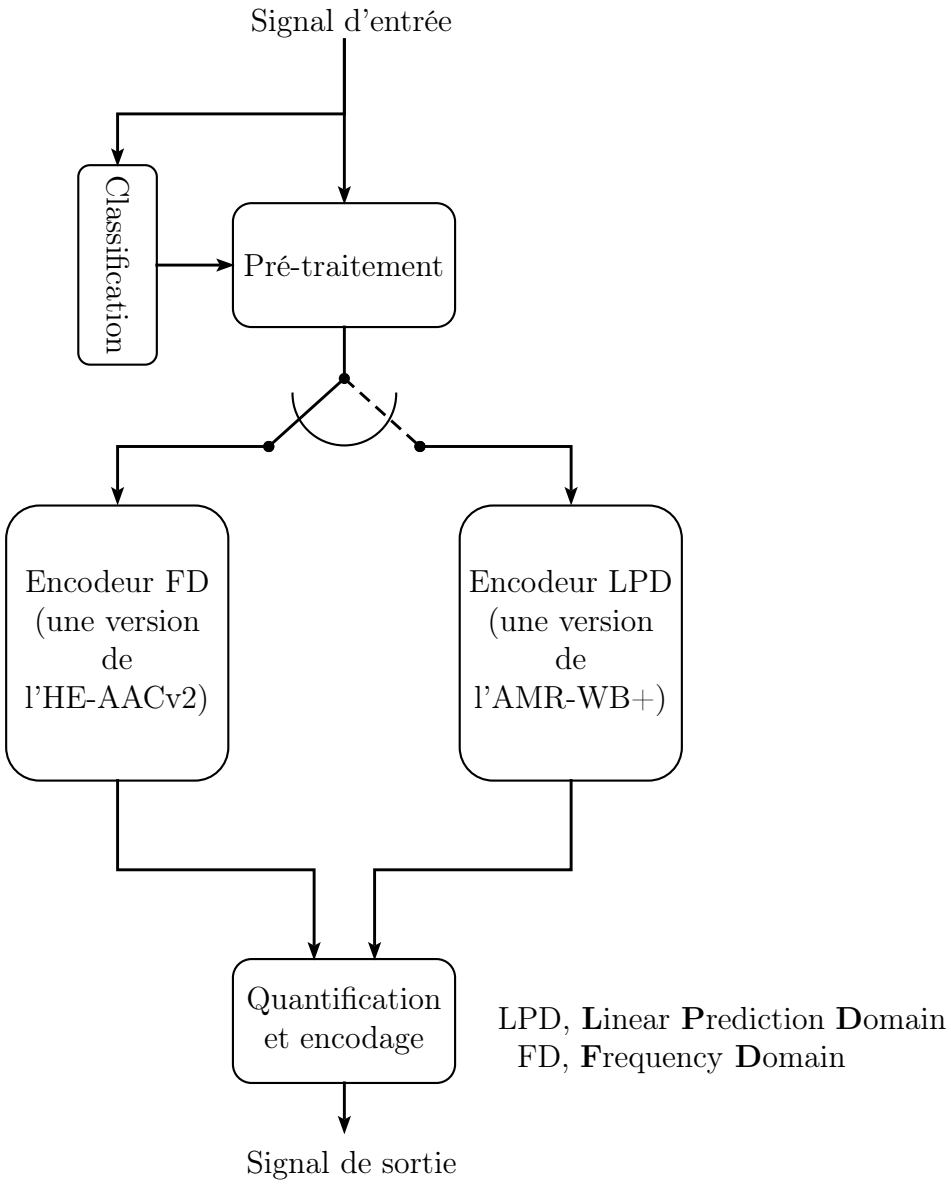


Figure 2.40 Schéma général à l'encodeur du modèle MPEG-D USAC (version RM0)

Le module FD (***F**requency **D**omain*) se compose d'une version dérivée du standard HE-AAC v2 et fonctionne dans le domaine de la transformée. Le second module LPD (***L**inear **P**redictive **D**omain*) représente une version dérivée du standard AMR-WB+ et fonctionne dans le domaine de la prédiction linéaire.

La figure 2.40 montre également que le modèle possède des modules communs aux deux modes FD et LPD : un pré-traitement et une quantification/encodage communs. La version standardisée USAC possède également des éléments d'unification supplémentaires comme la technique FAC (***F**orward **A**liasing **C**ancellation*) [Neuendorf *et al.*, 2012] qui permet de gérer les recouvrements des fenêtres entres les différents modèles de codage.

2.4.2 Conclusion sur le codage universel hybride

Le standard USAC présenté dans cette section représente actuellement une solution afin d'obtenir un modèle de compression hybride universel. Il obtient de bonnes performances pour tous les types de signaux audio.

2.5 Conclusion du chapitre

Ce chapitre présentait l'état actuel des recherches au niveau de la compression de tous les types de signaux audio à faible débit. Le chapitre démontre qu'il existe toujours une distinction entre les approches utilisées pour la compression des signaux de parole et pour la compression des signaux audio. Pour l'instant, l'unique moyen d'obtenir un modèle universel consiste à utiliser au moins deux modèles de codage et un classificateur afin de sélectionner le modèle à exécuter. La section 2.4.1 décrivait l'exemple d'un codec hybride universel appelé USAC (***U**nified **S**peech and **A**udio **Co**ding*) qui compresse tous les types de signaux audio avec une qualité uniforme.

CHAPITRE 3

MODÈLE D'ANALYSE-SYNTHÈSE PROPOSÉ

Ce chapitre présente les détails du modèle d'analyse-synthèse développé pour la compression des signaux de parole qui fonctionne entièrement dans le domaine de la transformée. Le modèle développé démontre qu'il est possible d'atteindre une qualité perceptuelle transparente sans nécessairement suivre l'évolution de la forme d'onde du signal original.

Ce modèle de codage pour les signaux de parole, qui fonctionne entièrement dans le domaine des fréquences contribue à paver la voie vers un codec réellement unifié puisque le domaine fréquentiel est traditionnellement réservé aux signaux audio. La figure 3.1 montre le schéma de fonctionnement du modèle développé qui utilise une approche de type harmonique-plus-bruit afin d'obtenir un modèle paramétrique simple avec le plus faible débit possible.

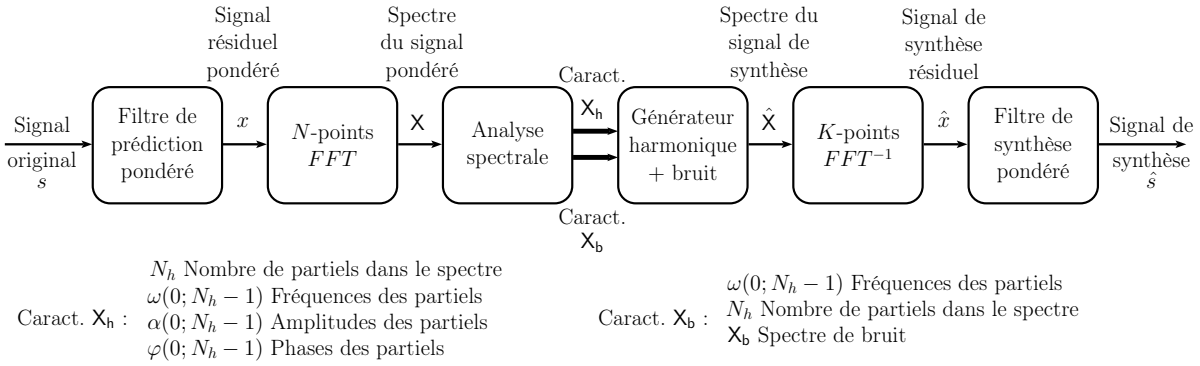


Figure 3.1 Principe général de fonctionnement du modèle proposé

La figure 3.1 montre une relative indépendance entre l'analyse (l'encodeur) et la synthèse (décodeur) par l'utilisation de longueurs différentes des transformées de Fourier et de Fourier inverse. Avec cette relative indépendance, le modèle démontre qu'il ne possède pas la contrainte de reconstruction parfaite qui caractérise les approches fréquentielles existantes.

Ainsi, le spectre possède un nombre de points plus élevé lors de l'analyse afin d'obtenir une grande précision, tandis que lors de la synthèse, le spectre possède un nombre de points plus faible pour bien suivre l'évolution du pitch du signal. Bien que le spectre de synthèse possède un nombre de points plus faible, le modèle réussit toutefois à bien suivre

l'évolution du pitch grâce à un générateur d'impulsions de grande précision. L'utilisation de ce générateur n'augmente pas la complexité de calculs, car la précision du générateur provient d'une table précalculée.

Déroulement du chapitre

Ce chapitre donne les détails du fonctionnement du modèle d'analyse-synthèse afin qu'il puisse utiliser des longueurs de transformée différentes tout en ayant un bon suivi du pitch. Ce chapitre commence par une description des spécifications générales du modèle. Il donne ensuite les étapes afin d'obtenir le spectre d'analyse pour que le modèle extrait les paramètres de ce spectre. Par la suite, le chapitre décrit comment le modèle crée le spectre de synthèse harmonique avec le générateur d'impulsions de sinusoïdes précalculées. Le chapitre donne également les détails du recouvrement entre les segments de synthèse afin d'obtenir la trame de sortie. Finalement, ce chapitre se termine par une conclusion de ce chapitre.

3.1 Spécifications générales du modèle

Le modèle utilise des signaux de parole PCM (***P**ulse **C**ode **M**odulation*) monophonique ayant une fréquence d'échantillonnage de 16 kHz comme signaux d'entrée. Le modèle segmente le signal d'entrée en trame de 12 ms (192 échantillons). Il utilise des trames de 12 ms afin de faciliter les manipulations lors de la conception, en particulier pour l'utilisation des différentes fenêtres. Cependant, le modèle a également été conçu pour une utilisation avec des trames de 10 ms, ce qui représente la longueur standard dans le domaine de la compression des signaux de parole.

Chaque nouvelle trame de 12 ms s'ajoute à la fin d'une mémoire tampon de 40 ms (640 échantillons) de type FIFO (***F**irst **I**n **F**irst **O**ut*), destinée pour la transformée de Fourier. Pour que la longueur reste constante à 40 ms dans la mémoire tampon le modèle retire les 12 ms de trame plus ancienne. Le modèle possède un délai algorithmique de 24 ms.

En raison des différentes longueurs des transformées de Fourier et de Fourier inverse, le modèle utilise de nombreuses fenêtres afin que ces différentes longueurs puissent se synchroniser entre l'analyse et la synthèse.

Fenêtrage dans le modèle

Pour s'assurer que toutes les fenêtres possèdent un recouvrement parfait entre elles, le modèle utilise des fenêtres w avec des extrémités symétriques (cf. figure 3.2).

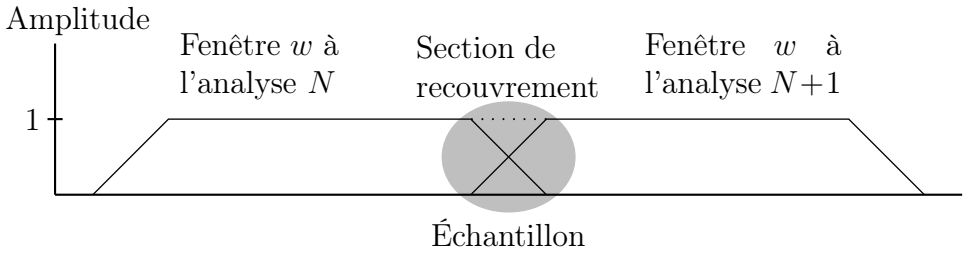


Figure 3.2 Symétrie des extrémités des fenêtres

Cette section a donné une description générale du modèle et de son mode de fonctionnement. Les prochaines sections décrivent les détails sur l'analyse du spectre et l'extraction de ses paramètres.

3.2 Analyse du spectre

La figure 3.1 montre que l'analyse du spectre permet d'obtenir les paramètres suivants : le nombre de partiels N_h, les fréquences ω, les amplitudes α et les phases φ de ces partiels. Pour obtenir le spectre d'analyse, le modèle applique une transformée de Fourier sur le signal de parole résiduel.

Filtre de prédiction à court-terme utilisé

Le modèle utilise un filtre de prédiction à court-terme sur la mémoire tampon de 40 ms afin d'obtenir le signal résiduel x pour la transformée de Fourier. Le filtre de prédiction de l'équation 3.1 possède un ordre $P = 16$, ce qui signifie qu'il possède 16 coefficient a_k et intègre également un filtre perceptuel d'une valeur $\gamma = 0.96$ (critère perceptuel). Avec cette valeur γ, le filtre perceptuel modifie légèrement les formants afin de rendre le spectre légèrement plus plat.

$$P_w(z) = A(z/\gamma) = 1 - \sum_{k=0}^{P-1} a_k \gamma^k z^{-k} \qquad \gamma = 0.96 \quad \text{et} \quad P = 16 \tag{3.1}$$

Le signal résiduel original x obtenu à partir du filtre de prédiction pondéré de l'équation 3.1 représente le signal de référence du modèle et possède la même longueur que la mémoire tampon, c'est-à-dire 40 ms (640 échantillons). C'est sur ce signal résiduel x que le

modèle applique une transformée de Fourier afin d'obtenir le spectre original X destiné pour l'analyse.

3.2.1 Calcul du spectre pour l'analyse

Avant l'utilisation de la transformée de Fourier de $N = 1024$-points, le modèle applique des pré-traitements de fenêtrage et de *zero padding* sur le signal résiduel x.

Fenêtrage pour l'analyse

Afin d'éviter le phénomène d'étalement spectral, le modèle fenêtre le signal résiduel x avec l'équation 3.2.

$$\tilde{x}[n] = x[n] \cdot w_{\text{fft}}[n] \qquad n = [0, ..., 640[,\ n \in \mathbb{N} \tag{3.2}$$

La fenêtre w_{fft} de la figure 3.3 possède la même longueur que le signal résiduel x, c'est-à-dire 40 ms (640 échantillons). Elle possède des extrémités en forme de demi-Hanning (cf. équations 3.3 et 3.4) ainsi qu'un centre plat (cf. équation 3.5).

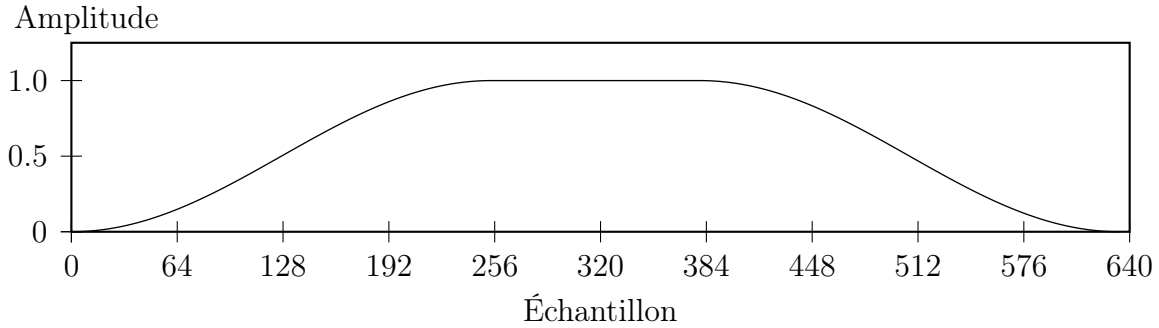


Figure 3.3 Fenêtre w_{fft} utilisée sur le signal résiduel x avant la FFT

$$w_{han} = 0.5\left(1 - cos\left(\frac{2\pi i}{L_{han} - 1}\right)\right) \qquad 0 \leqslant i < L_{han} \quad \text{où} \quad L_{han} = 192 \tag{3.3}$$

$$w'_{han} = w_{han}(L_{han} - 1 - i) \qquad 0 \leqslant i < L_{han} \quad \text{où} \quad L_{han} = 192 \tag{3.4}$$

$$w_{\text{fft}}[n] = \begin{cases} w_{han}[i] & 0 \leqslant n < 192 \\ & 0 \leqslant i < 192 \\ 1 & 192 \leqslant n < 448 \\ w'_{han}[i] & 448 \leqslant n < 640 \\ & 0 \leqslant i < 192 \end{cases} \tag{3.5}$$

Pour que le signal résiduel x de 640 échantillons possède la même longueur que la transformée de Fourier de 1024-points, le modèle utilise la technique de complétion de zéros (*zero padding*).

Technique de complétion de zéros (*zero padding*)

La figure 3.4 montre la technique d'ajout de zéros (*zero padding*) aux extrémités du signal résiduel fenêtré x afin d'obtenir la même longueur que les $N = 1024$-points de la transformée de Fourier.

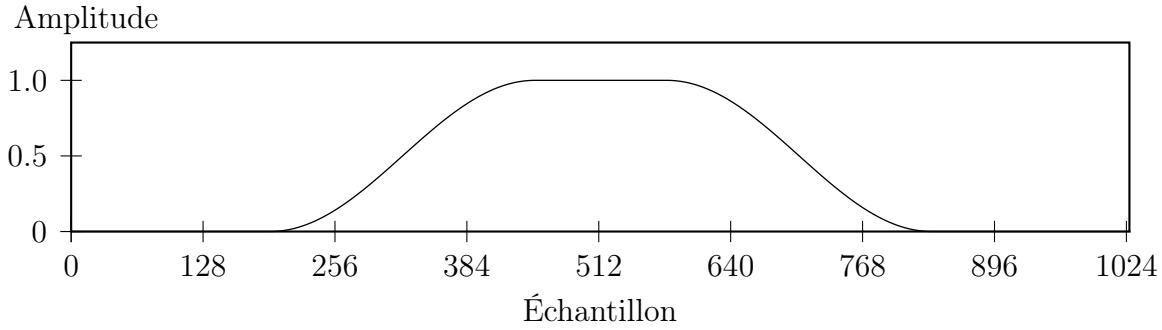


Figure 3.4 Longueur et forme de la fenêtre appliquée sur le signal résiduel x

Utilisation de la transformée de Fourier

C'est avec le signal résiduel fenêtré $\tilde{x}$ de 1024 échantillons de la figure 3.4 que le modèle applique la transformée de Fourier de l'équation 3.6.

$$\mathsf{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j2\pi kn/N} \qquad \begin{aligned} N &= 1024 \\ n &= [0, ..., N[,\ n \in \mathbb{N} \\ k &= [0, ..., N[,\ n \in \mathbb{N} \end{aligned} \tag{3.6}$$

Avec l'équation 3.6, le modèle obtient un spectre X de 512-points avec une résolution de 15.625 Hz par canal. C'est avec ce spectre X que le modèle effectue une analyse afin d'extraire les paramètres nécessaires pour créer le signal de synthèse.

3.2.2 Recherche de la fréquence fondamentale

La valeur de la fréquence fondamentale ω_0 varie selon l'âge et le sexe du locuteur. En moyenne, la fréquence fondamentale de la voix d'un homme adulte se situe autour de 100 Hz à 150 Hz tandis que pour la femme adulte, elle se situe aux alentours de 140 Hz à 240 Hz [Calliope, 1989].

Lors de l'analyse, la recherche de la fréquence fondamentale ω_0 s'effectue à partir du 5^{e} canal jusqu'au 30^{e} canal du spectre d'amplitudes $|\mathsf{X}|$ (cf. équation 3.7), ce qui assure un balayage de toute la plage de fréquences fondamentales possibles. Ces valeurs représentent en unité hertz 78.125 Hz et 468.750 Hz, puisque chaque canal du spectre d'amplitudes vaut 15.625 Hz.

Pour déterminer la fréquence fondamentale, le modèle recherche l'autocorrélation la plus élevée sur le spectre original X. L'équation 3.7 montre que l'autocorrélation se calcule avec la fréquence fondamentale hypothétique ω_h (du 5^{e} canal au 30^{e} canal) ainsi qu'avec le deuxième $(2 \cdot \omega_h)$ et le troisième $(3 \cdot \omega_h)$ multiple entier de la fréquence fondamentale ω_h.

$$R_\omega[k] = \sum_{i=1}^{3} |\mathsf{X}[ik]|^2 \qquad k = 5, ..., 30 \text{ où } k \in \mathbb{N} \tag{3.7}$$

La valeur d'autocorrélation $R_\omega[k]$ la plus élevée détermine la valeur de la fréquence fondamentale ω_0 du spectre X (cf. équation 3.8).

$$\underset{k=5,...,30 \text{ où } k \in \mathbb{N}}{\arg\max} R_\omega[k] \Rightarrow k_0 = \omega_0 \qquad \omega_0 = \text{fréquence fondamentale trouvée} \tag{3.8}$$

Par la suite, le modèle recherche le degré de voisement du spectre X en déterminant le nombre de partiels avec la fréquence fondamentale trouvée ω_0.

3.2.3 Recherche des partiels dans le spectre

Le modèle utilise une approche de type harmonique-plus-bruit pour la modélisation du signal de parole, ce qui signifie que les partiels du spectre harmonique se situent à des multiples de la fréquence fondamentale ω_0. Le modèle utilise cette caractéristique harmonique pour la création d'un filtre en peigne P avec des sommets aux multiples de la fréquence fondamentale ω_0.

Le modèle procède par la méthode de *peak picking* pour la recherche des partiels. Ainsi, le filtre en peigne P se déplace dans le spectre pour calculer le niveau de corrélation croisée et déterminer les positions des partiels lors de corrélations élevées.

Afin d'obtenir des positions précises des partiels, le modèle effectue deux types de recherches : une première recherche globale afin de trouver un partiel et une seconde recherche plus précise pour déterminer la position exacte du partiel.

Recherche globale pour trouver des partiels

La figure 3.5 montre l'exemple d'un filtre en peigne P avec une fréquence fondamentale ω_0 au 16$^{\text{e}}$ canal (250 Hz) du spectre d'amplitudes $|\mathsf{X}|$. Chaque sommet du peigne P possède un espacement régulier de la valeur de la fréquence fondamentale ω_0 et se représente avec 5 points possédant des gains de 1.0, 0.8 et 0.3 (cf. figure 3.5).

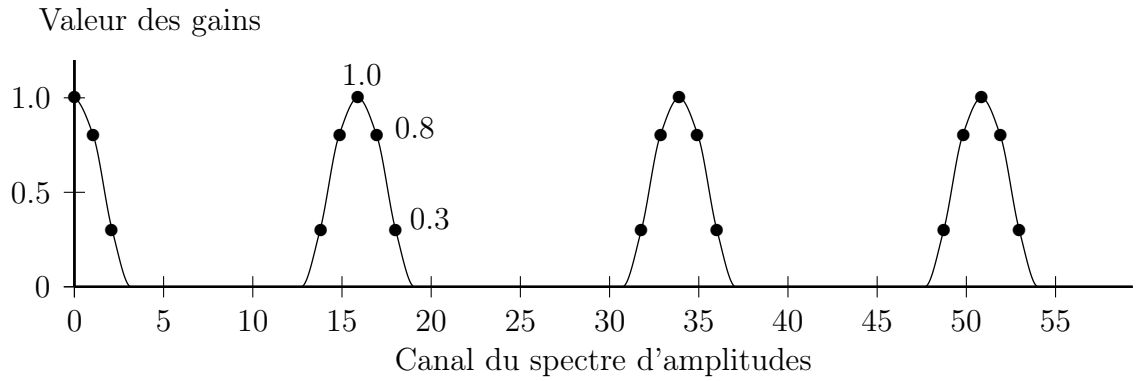


Figure 3.5 Exemple d'un filtre en peigne P avec $\omega_0 = 16$ (250 Hz)

L'équation 3.9 montre que la recherche de partiels s'effectue par une corrélation croisée entre le spectre d'amplitudes original $|\mathsf{X}|$ et le spectre d'amplitudes du filtre en peigne $|\mathsf{P}|$.

$$R[k] = \sum_{i=0}^{L_{\mathsf{P}}-1} |\mathsf{X}[k+i]| \cdot |\mathsf{P}[i]| \qquad \begin{aligned} L_{\mathsf{P}} &= \text{Longueur du filtre en peigne } \mathsf{P} \\ k &= k + \omega_0 \\ &\text{Initialement } k = 0 \end{aligned} \tag{3.9}$$

Le filtre P se déplace dans le spectre X avec un pas égal à la valeur de la fréquence fondamentale ω_0 pour calculer la corrélation croisée avec l'équation 3.9. Le calcul de la corrélation s'effectue tant que la valeur $R[k]$ se situe au-dessus d'un seuil préétabli.

Cette première recherche dans le spectre d'amplitudes détermine la présence de partiels. Une seconde recherche s'effectue autour du partiel trouvé afin de déterminer précisément sa position.

Recherche locale pour la position exacte du partiel

Afin d'obtenir la position précise des partiels, le modèle calcule une seconde corrélation locale R_l pour chaque partiel trouvé k_l avec l'équation 3.10.

$$R_l[k'] = \sum_{i=0}^{L_{\mathsf{P}}-1} |\mathsf{X}[k'+i]| \cdot |P[i]| \qquad \begin{aligned} L_P &= \text{Longueur du filtre en peigne } P \\ k' &= k_l - 2, ..., k_l + 2 \end{aligned} \tag{3.10}$$

Avec l'équation 3.10, le modèle recherche la corrélation croisée locale la plus élevée à ± 2 canaux autour du partiel trouvé k_l. Le résultat de l'équation 3.10 possède un impact sur la position du peigne pour la suite de la recherche de partiels, car l'équation 3.11 montre que la corrélation maximum locale trouvée dans l'équation 3.10 modifie la position du filtre en peigne P. Le modèle ajuste la positionne du filtre en peigne pour la recherche du prochain partiel avec la corrélation maximum locale trouvée par l'équation 3.11.

$$\underset{k' \in [k_l-2,...,k_l+2],\ k' \in \mathbb{N}}{\arg\max} R_l[k'] = k_p \quad \text{Position du fitre en peigne } \mathsf{P} \tag{3.11}$$

La figure 3.6 montre un exemple de partiels trouvés à l'aide du filtre en peigne P de la figure 3.5. À la fin du processus de recherche du *peak picking*, le modèle connaît le nombre de partiels N_h que le spectre d'amplitudes contient.

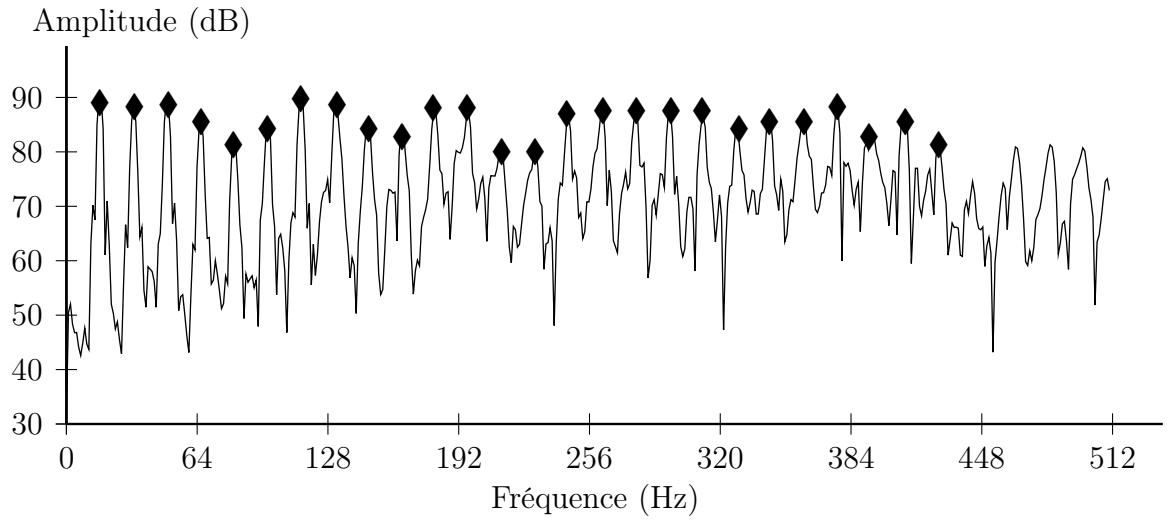


Figure 3.6 Exemple des partiels trouvés avec le filtre en peigne de la fig. 3.5

La prochaine section montre comment le modèle extrait les paramètres des partiels du spectre original X à l'aide de la fréquence fondamentale ω_0 et le nombre de partiels N_h.

3.2.4 Extraction des paramètres provenant du spectre

Avant d'extraire les paramètres des partiels du spectre, le modèle effectue un second calcul de la fréquence fondamentale ω_0 afin de tenir compte de toutes les recherches de corrélations globales et locales effectuées par la section 3.2.3.

Après toutes les recherches effectuées sur le spectre, le modèle utilise l'équation 3.12 avec la valeur de la dernière harmonique ω_{N_h-1} et le nombre de partiels trouvés N_h afin d'obtenir un meilleur estimé de la fréquence fondamentale ω_0.

$$\omega_0 = \frac{\omega_{N_h-1}}{N_h} \tag{3.12}$$

Le modèle utilise la fréquence fondamentale ω_0 de l'équation 3.12 afin de calculer les paramètres de chaque partiel i : la fréquence $\omega[i]$, l'amplitude $\alpha[i]$ et la phase $\varphi[i]$.

Calcul de la fréquence des partiels

La fréquence $\omega[i]$ d'un partiel i représente un multiple de la fréquence fondamentale ω_0 et s'obtient avec l'équation 3.13.

$$\omega[i] = j \cdot \omega_0 \qquad \begin{aligned} j &= [1, ..., N_h],\ j \in \mathbb{N} \\ i &= [0, ..., N_h[,\ i \in \mathbb{N} \end{aligned} \tag{3.13}$$

Calcul de l'amplitude des partiels

Afin d'obtenir une meilleure représentation de la valeur de l'amplitude du partiel, le modèle utilise cinq canaux du spectre d'amplitudes $|\mathsf{X}|$ pour le calcul de l'équation 3.14.

$$\alpha[i] = \sqrt{\sum_{k=\omega[i]-2}^{\omega[i]+2} |\mathsf{X}[k]|} \qquad i = [0, ..., N_h[,\ i \in \mathbb{N} \tag{3.14}$$

Calcul de la phase des partiels

Finalement, la phase du partiel $\varphi[i]$ se calcule avec l'équation 3.15.

$$\varphi[i] = \arctan\left(\frac{Im(\mathsf{X}[\omega[i]])}{Re(\mathsf{X}[\omega[i]])}\right) \qquad i = [0, ..., N_h[,\ i \in \mathbb{N} \tag{3.15}$$

Le tableau 3.1 donne un récapitulatif des paramètres que le modèle obtient avec l'analyse du spectre original X.

Tableau 3.1 Éléments obtenus lors de l'analyse du spectre

N_h	Nombre de partiels dans le spectre
$\omega[0, ..., N_h[$	Fréquences des partiels
$\alpha[0, ..., N_h[$	Amplitudes des partiels
$\varphi[0, ..., N_h[$	Phases des partiels

Cette section du document a décrit la partie analyse du spectre qui extrait tous les paramètres importants des partiels pour la création du spectre de synthèse. La prochaine

section explique les détails de la création du spectre de synthèse et particulièrement le fonctionnement du générateur d'impulsions de sinusoïdes précalculées.

3.3 Synthèse du spectre

La figure 3.1 montrait que le modèle utilise différentes longueurs pour les transformées de Fourier et de Fourier inverse. Lors de l'analyse, le spectre contient une grande résolution de 1024-points afin d'extraire les valeurs précises du spectre original X, tandis que pour la synthèse, le spectre $\hat{\mathsf{X}}$ possède un nombre de points plus faible de 512 - points afin de bien suivre l'évolution du pitch. Les prochaines sections expliquent comment le modèle réussit à suivre rapidement et précisément l'évolution du pitch sur un spectre contenant peu de canaux, à l'aide d'un générateur d'impulsions de sinusoïdes précalculées .

Au début de la synthèse, tous les points du spectre possèdent des valeurs nulles. C'est le générateur d'impulsions qui ajoutent les partiels avec les paramètres trouvés lors de l'analyse.

3.3.1 Générateur d'impulsions de sinusoïdes précalculées

Afin d'augmenter la précision du spectre et sans augmenter la complexité du modèle, celui-ci possède un générateur d'impulsions avec une table précalculée. Cette table augmente la précision du spectre en permettant le positionnement d'un partiel entre deux canaux du spectre. Avec les 16 impulsions de la table, le spectre obtient une résolution de $1/16$ entre deux canaux.

Création de la table d'impulsions de sinusoïdes

Le modèle utilise une table précalculée afin de positionner précisément les partiels entre deux canaux du spectre. La table segmente en 16 régions d'égales largeurs la distance entre deux canaux.

Pour obtenir la table d'impulsions précalculées, le modèle commence par créer des sinusoïdes dans le domaine temporel. Ces 16 sinusoïdes possèdent toutes la même fréquence fondamentale et c'est valeur de la phase qui varie. L'équation 3.16 montre la création des 16 sinusoïdes dans le domaine du temps.

$$\begin{cases} \omega_p & \text{Fréquence de référence posée à 25 (25}^{\text{e}}\text{ canal du spectre)} \\ N_{su} & \text{Nombre de sous-unité entre deux canaux de la FFT} \\ N_s & \text{Longueur de la FFT}^{-1}\text{ pour la synthèse} \\ m & \text{Nombre de cosinus} \end{cases}$$

$$c_m = cos\left(\frac{2\pi \cdot \omega_p \cdot n}{N_s} + \frac{2\pi \cdot k}{N_s \cdot N_{su}}\right) \qquad \begin{array}{ll} n = & [0, ..., 512] \\ k = & [0, ..., 16[\\ m = & [0, ..., 16[\end{array} \tag{3.16}$$

L'équation 3.16 pose la fréquence $\omega_p = 25$ (25$^{\text{e}}$ canal du spectre) pour les sinusoïdes et incrémente leur phase de $\pi/8$ afin de créer les 16 sinusoïdes. La fréquence $\omega_p = 25$ représente un choix arbitraire car le générateur utilise uniquement les valeurs de pas entre les canaux du spectre, ainsi une autre valeur de fréquence donnerait les mêmes valeurs de pas. Avant d'appliquer des transformées de Fourier sur ces sinusoïdes, le modèle leur applique un fenêtrage afin d'éviter le phénomène d'étalement spectral (cf. équation 3.17).

$$\tilde{c}_m[n] = c_m[n] \cdot w_{imp}[n] \qquad [0, ..., 512[, n \in \mathbb{N} \tag{3.17}$$

La fenêtre w_{imp} utilisée sur les sinusoïdes (cf. figure 3.7) possède des extrémités de demi-Hanning de 192 échantillons (cf. équations 3.18 et 3.19) et un centre plat de 128 échantillons (cf. équation 3.20).

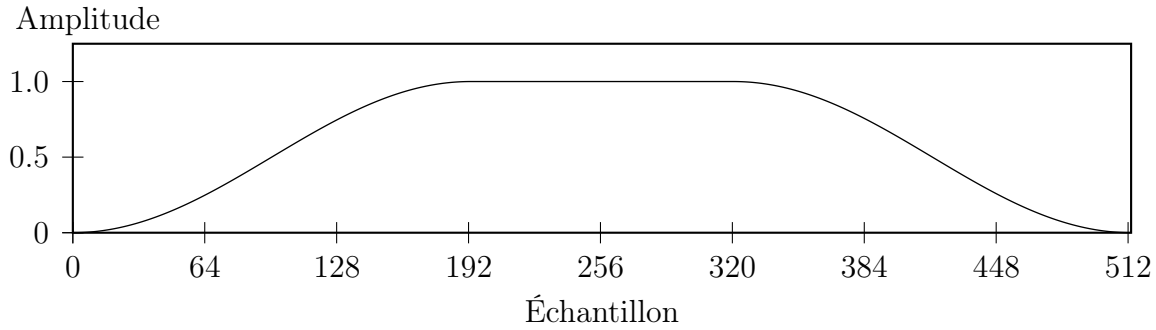


Figure 3.7 Longueur et forme de la fenêtre appliquée sur les sinusoïdes du tableau c_m

$$w_{han} = 0.5\left(1 - cos\left(\frac{2\pi i}{L_{han} - 1}\right)\right) \qquad 0 \leqslant i < L_{han} \quad \text{où} \quad L_{han} = 192 \tag{3.18}$$

$$w'_{han} = w_{han}(L_{han} - 1 - i) \qquad 0 \leqslant i < L_{han} \quad \text{où} \quad L_{han} = 192 \tag{3.19}$$

$$w_{imp}[n] = \begin{cases} w_{han}[i] & 0 \leqslant n < 192 \\ & 0 \leqslant i < 192 \\ 1 & 192 \leqslant n < 320 \\ w'_{han}[i] & 320 \leqslant n < 512 \\ & 0 \leqslant i < 192 \end{cases} \tag{3.20}$$

Par la suite, le modèle applique une transformée de Fourier de 512-points sur chaque sinusoïde afin d'obtenir les impulsions C_m pour la table (cf. équation 3.21).

$$\mathsf{C}_m[k] = \sum_{n=0}^{N-1} c_m[n] e^{-2\pi jkn/N} \qquad \begin{aligned} N &= 512 \\ n &= [0, ..., 512[,\ n \in \mathbb{N} \\ m &= [0, ..., 16[,\ m \in \mathbb{N} \\ k &= [0, ..., 256[,\ k \in \mathbb{N} \end{aligned} \tag{3.21}$$

Avant l'ajout des impulsions dans la table, le modèle les normalise avec l'équation 3.22 et conserve huit canaux de spectre des impulsions pour la table (cf. équation 3.23). Le nombre de canaux conservé représente le meilleur compromis afin de réduire le nombre de canaux et d'obtenir la meilleure représentation possible.

$$norm = \sum_{k=\omega_p-3}^{\omega_p+4} |\mathsf{C}_m[k]|^2 \tag{3.22}$$

$$\mathsf{C}_m[k'] = \frac{\mathsf{C}_m[k']}{norm} \qquad k' = [k-3, ..., k+4[,\ k' \in \mathbb{C} \tag{3.23}$$

L'équation 3.24 montre que la table T_{imp} contient la position des 16 impulsions.

$$T_{\text{imp}}[m, k'] = \mathsf{C}_m[k'] \qquad \begin{aligned} k' &= [k-3, ..., k+4[,\ k' \in \mathbb{C} \\ m &= [0, ..., N_{su}[,\ m \in \mathbb{N} \end{aligned} \tag{3.24}$$

Avec cette table précalculée, le modèle améliore la précision du spectre tout en n'augmentant pas la complexité des calculs. Cette table permet de positionner précisément les partiels entre 2 canaux du spectre. Les prochaines sections du document expliquent comment le générateur d'impulsions ajoute les partiels dans le spectre de synthèse harmonique $\hat{\mathsf{X}}_{\mathsf{h}}$.

3.3.2 Calcul de la position du partiel dans le spectre

Avec la table précalculée d'impulsions, le modèle a la possibilité de positionner les partiels entre deux canaux du spectre. Pour déterminer avec précision la position entre deux canaux, le modèle augmente la valeur de la fréquence fondamentale par une multiplication de 8 (cf. équation 3.25).

$$\omega'[i] = \omega[i] \cdot 8 \qquad i = [0, ..., N_h[,\ i \in \mathbb{N} \tag{3.25}$$

La multiplication par 8 de l'équation 3.25 augmente la précision de la partie fractionnaire de la fréquence ce qui implique également une augmentation de la précision de la position entre deux canaux. L'équation 3.26 montre le calcul afin de déterminer la partie entière de la position du canal dans le spectre. La division par 16 dans l'équation 3.26 considère l'augmentation de la résolution par l'équation 3.25 ainsi que le nombre de points plus faible pour le spectre de synthèse (512 points au lieu de 1024 points pour le spectre d'analyse).

$$k = \omega'[i] \;/\; 16 \qquad \begin{aligned} i &= [0, ..., N_h[,\ i \in \mathbb{N} \\ k &= \text{position dans le spectre } \hat{\mathsf{X}}_\mathsf{h} \end{aligned} \tag{3.26}$$

L'équation 3.27 détermine la position fractionnaire entre deux canaux du spectre avec une précision de 1/16. Le résultat de l'équation 3.27 donne l'indice pour le tableau d'impulsions T_{imp} afin de sélectionner la bonne impulsion.

$$m = \omega'[i] \mod 16 \tag{3.27}$$

La prochaine section explique comment le modèle ajoute les partiels dans le spectre de synthèse $\hat{\mathsf{X}}_\mathsf{h}$ avec le générateur d'impulsions de sinusoïdes.

3.3.3 Ajout des partiels dans le spectre

Le générateur crée entièrement le spectre harmonique, ce qui signifie qu'initialement le spectre contient uniquement des valeurs nulles. Par la suite, le générateur ajoute les partiels trouvés durant l'analyse. Le générateur utilise l'amplitude α, la fréquence ω et la phase φ pour ajouter un partiel à la fois. Il utilise les identités trigonométriques de l'équation 3.28 pour l'ajout de partiels dans le spectre.

$$\begin{aligned} \cos(a+b) &= \cos a \cdot \cos b - \sin a \cdot \sin b \\ \sin(a+b) &= \sin a \cdot \cos b + \cos a \cdot \sin b \end{aligned} \tag{3.28}$$

L'équation 3.29 montre comment le modèle ajoute chaque partiel avec les impulsions $\bar{\mathsf{C}}_m$ de la table précalculée T_{imp}. L'équation 3.29 possède la même forme que l'équation 3.28. Le modèle utilise huit canaux du spectre afin de créer un partiel. Le symbole $+=$ indique que l'équation 3.29 est récursive.

$$\begin{aligned} Re(\hat{\mathsf{X}}_\mathsf{h}[k']) &+= \alpha[i]cos(\varphi[i]) \cdot Re(\bar{\mathsf{C}}_m[j]) - \alpha[i]sin(\varphi[i]) \cdot Im(\bar{\mathsf{C}}_m[j]) \\ Im(\hat{\mathsf{X}}_\mathsf{h}[k']) &+= \alpha[i]sin(\varphi[i]) \cdot Re(\bar{\mathsf{C}}_m[j]) + \alpha[i]cos(\varphi[i]) \cdot Im(\bar{\mathsf{C}}_m[j]) \end{aligned} \tag{3.29}$$

$$\text{où} \begin{cases} k' = [k-3, ..., k+4],\ k' \in \mathbb{N} \\ i = [0, ..., N_h[,\ i \in \mathbb{N} \\ j = [0, ..., N_c[,\ j \in \mathbb{N} \end{cases}$$

Après l'ajout de tous les partiels dans le spectre harmonique $\hat{\mathsf{X}}_\mathsf{h}$, le modèle applique une transformée de Fourier inverse avec l'équation 3.30, afin d'obtenir le signal de synthèse harmonique temporel $\hat{x}_h$ d'une longueur de 16 ms (256 échantillons).

$$\hat{x}_h[n] = \frac{1}{K} \sum_{k=0}^{K-1} \hat{\mathsf{X}}_\mathsf{h}[k] e^{2\pi jkn/K} \qquad \begin{aligned} K &= 512 \\ k &= [0, ..., K[,\ k \in \mathbb{N} \\ n &= [0, ..., 256[,\ n \in \mathbb{N} \end{aligned} \tag{3.30}$$

Avec le signal de synthèse $\hat{x}_h$ obtenu, le modèle applique un recouvrement entre les segments afin d'obtenir la trame de synthèse.

3.4 Recouvrement entre les trames

Le signal de synthèse obtenu possède une longueur de 16 ms, mais le modèle crée des trames de sortie de 12 ms. Ainsi, cette section décrit les étapes que le modèle effectue pour le recouvrement entre les segments de synthèse afin d'obtenir des trames de sortie de 12 ms. La première étape consiste à ajuster le gain d'énergie du signal de synthèse harmonique $\hat{x}_h$ avec le signal original harmonique x_h. Par la suite, le modèle applique le filtre de synthèse pondéré et finalement, il effectue le recouvrement avec les segments de synthèse antérieurs pour obtenir une trame de 12 ms.

3.4.1 Ajustement du gain du signal de synthèse

Afin de s'assurer que le signal de synthèse possède la même énergie que le signal original, le modèle applique un gain d'énergie G_h au signal harmonique de synthèse $\hat{x}_h$. Le gain G_h compare l'énergie entre les signaux harmonique original x_h et harmonique de synthèse $\hat{x}_h$. Pour obtenir le signal harmonique original x_h, le modèle utilise la fréquence de coupure ω_c afin de séparer la partie harmonique de la partie bruit (cf. équation 3.31).

L'équation 3.31 montre que le modèle modélise le signal de parole x avec une partie harmonique x_h et une partie non-harmonique x_b. La segmentation du signal de parole x en une partie harmonique x_h et une partie bruit x_b s'effectue dans le domaine des fréquences.

$$x = x_h + x_b \tag{3.31}$$

Segmentation du signal original : harmonique et bruit

Pour segmenter le signal original avec la fréquence de coupure ω_c, le modèle utilise la fréquence fondamentale ω_0 et le nombre d'harmoniques N_h trouvés lors de l'analyse dans la section 3.2 (cf. équation 3.32).

$$\omega_c = (\omega_0 \cdot N_h) + \frac{\omega_0}{2} \tag{3.32}$$

La segmentation de la partie harmonique et la partie bruit s'effectue dans le domaine des fréquences. Avant d'appliquer la transformée de Fourier, le modèle fenêtre le signal original avec des demi-cosinus pour éviter le phénomène d'étalement spectral (cf. équation 3.33).

$$\ddot{x}[n] = x[n] \cdot w_{cos}[n] \qquad n \in [0, ..., 256[, n \in \mathbb{N} \tag{3.33}$$

La fenêtre w_{cos} de la figure 3.8 possède des extrémités de demi-cosinus de 64 échantillons (cf. équations 3.34 et 3.35) et un centre plat de 128 échantillons (cf. équation 3.36).

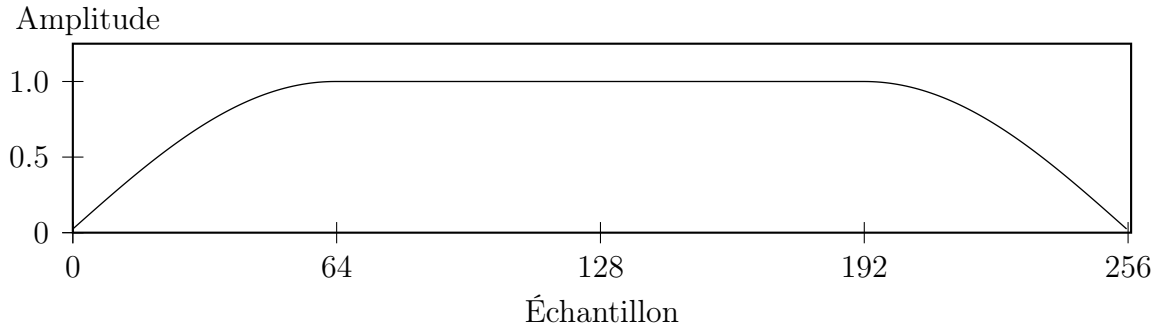


Figure 3.8 Longueur et forme de la fenêtre appliquée sur le signal x

$$w_c = sin\left(\frac{\pi \cdot i}{2 \cdot L_c}\right) \qquad 0 \leqslant i < L_c \quad \text{où} \;\; L_c = 64 \tag{3.34}$$

$$w'_c = w_c(L_c - 1 - i) \qquad 0 \leqslant i < L_c \quad \text{où} \;\; L_c = 64 \tag{3.35}$$

$$w_{cos}[n] = \begin{cases} w_c[i] & 0 \leqslant n < 64 \\ & 0 \leqslant i < 64 \\ 1 & 64 \leqslant n < 192 \\ w'_c[i] & 192 \leqslant n < 256 \\ & 0 \leqslant i < 64 \end{cases} \tag{3.36}$$

Après le fenêtrage, le modèle applique une transformée de Fourier de 256-points sur le signal x avec l'équation 3.37.

$$\ddot{\mathrm{X}}[k] = \sum_{n=0}^{M-1} \ddot{x}[n] e^{-j2\pi kn/M} \qquad \begin{aligned} M = &\;\; 256 \\ n = &\;\; [0, ..., M[,\; n \in \mathbb{N} \\ k = &\;\; [0, ..., M[,\; k \in \mathbb{N} \end{aligned} \tag{3.37}$$

Avec la fréquence de coupure calculée à l'équation 3.32, le modèle segmente le spectre d'amplitudes $|\ddot{\mathsf{X}}|$ pour obtenir un spectre harmonique $\ddot{\mathsf{X}}_{\mathsf{h}}$ et un spectre de bruit $\ddot{\mathsf{X}}_{\mathsf{b}}$. Les équations 3.38 et 3.39 montrent les contenus du spectre harmonique $\ddot{\mathsf{X}}_{\mathsf{h}}$ et du spectre de bruit $\ddot{\mathsf{X}}_{\mathsf{b}}$.

$$|\ddot{\mathsf{X}}_{\mathsf{h}}[k]| = \begin{cases} |\ddot{\mathsf{X}}[k]| & k = [0, ..., \frac{\omega_c}{4}[,\ k \in \mathbb{N} \\ \\ 0 & k = [\frac{\omega_c}{4}, ..., 128[,\ k \in \mathbb{N} \end{cases} \tag{3.38}$$

$$|\ddot{\mathsf{X}}_{\mathsf{b}}[k]| = \begin{cases} 0 & k = [0, ..., \frac{\omega_c}{4}[,\ k \in \mathbb{N} \\ \\ |\ddot{\mathsf{X}}[k]| & k = [\frac{\omega_c}{4}, ..., 128[,\ k \in \mathbb{N} \end{cases} \tag{3.39}$$

Le modèle divise par quatre la valeur de la fréquence de coupure ω_c dans les équations 3.38 et 3.39, car la valeur ω_c provient d'un spectre de 1024-points. Le modèle applique par la suite des transformées de Fourier inverse sur les spectres harmonique $\ddot{\mathsf{X}}_{\mathsf{h}}$ et de bruit $\ddot{\mathsf{X}}_{\mathsf{b}}$ afin d'obtenir ces signaux dans le domaine temporel (cf. équations 3.40 et 3.41).

$$\ddot{x}_h[n] = \frac{1}{M} \sum_{k=0}^{M-1} \ddot{\mathsf{X}}_{\mathsf{h}}[k] e^{2\pi jkn/M} \qquad \begin{aligned} M &= 256 \\ k &= [0, ..., M[,\ k \in \mathbb{N} \\ n &= [0, ..., M[,\ n \in \mathbb{N} \end{aligned} \tag{3.40}$$

$$\ddot{x}_b[n] = \frac{1}{M} \sum_{k=0}^{M-1} \ddot{\mathsf{X}}_{\mathsf{b}}[k] e^{2\pi jkn/M} \qquad \begin{aligned} M &= 256 \\ k &= [0, ..., M[,\ k \in \mathbb{N} \\ n &= [0, ..., M[,\ n \in \mathbb{N} \end{aligned} \tag{3.41}$$

Le modèle utilise les signaux originaux harmonique $\ddot{x}_h$ et de bruit $\ddot{x}_b$ afin de les comparer avec les signaux de synthèse. La prochaine partie explique comment le modèle calcule le gain d'énergie pour la partie harmonique.

Fenêtrage des signaux pour le calcul du gain

Afin de s'assurer que la partie harmonique possède la bonne énergie, le modèle applique un gain sur le signal de synthèse $\hat{x}_h$. Avant de calculer le gain, le modèle s'assure que les signaux à comparer possèdent la même forme de fenêtre, c'est-à-dire une fenêtre avec des extrémités de demi-Hanning de 64 échantillons et un centre plat de 128 échantillons (cf. figure 3.9).

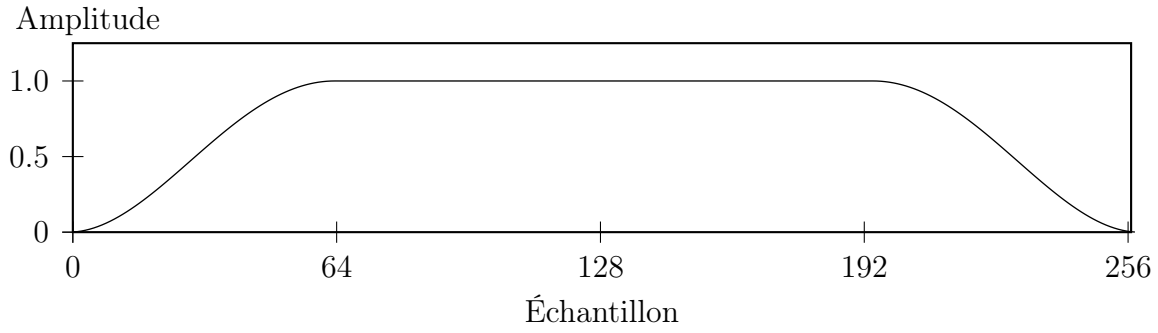


Figure 3.9 Longueur et forme de la fenêtre pour le calcul du gain harmonique

Pour obtenir la forme de la fenêtre de la figure 3.9 sur le signal original harmonique $\ddot{x}_h$, le modèle applique une seconde fois la fenêtre de demi-cosinus de la figure 3.8.

Dans la situation du signal de synthèse harmonique $\hat{x}_h$, le modèle applique une fenêtre de demi-cosinus modifiée w_{hm} (cf. équation 3.42).

$$\hat{\hat{x}}_h[n] = \hat{x}_h[n] \cdot w_{hm}[n] \qquad n \in [0, ..., 256[, n \in \mathbb{N} \tag{3.42}$$

La figure 3.10 montre la fenêtre de demi-cosinus modifiée w_{hm} qui tient compte du fenêtrage des impulsions du générateur w_{imp}, comme l'indique l'équation 3.45. Le modèle applique un décalage de 128 échantillons sur la fenêtre w_{imp} de l'équation 3.45 pour utiliser uniquement les 256 échantillons nécessaires.

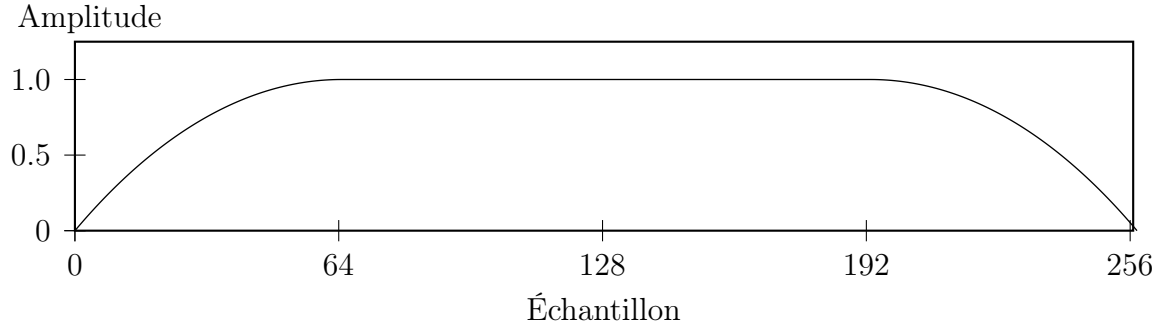


Figure 3.10 Longueur et forme de la fenêtre appliquée sur le signal de synthèse $\hat{x}_h$

$$w_{hm} = 0.5\left(1 - cos\left(\frac{2\pi i}{L_{hm} - 1}\right)\right) \qquad 0 \leqslant i < L_{hm} \quad \text{où} \quad L_{hm} = 64 \tag{3.43}$$

$$w'_{hm} = w_{han}(L_{han} - 1 - i) \qquad 0 \leqslant i < L_{hm} \quad \text{où} \quad L_{han} = 64 \tag{3.44}$$

$$w_{hm}[n] = \begin{cases} \dfrac{w_{hm}[n]}{w_{imp}[128+i]} & \begin{matrix} 0 \leqslant n < 64 \\ 0 \leqslant i < 64 \end{matrix} \\ 1 & 64 \leqslant n < 192 \\ \dfrac{w'_{hm}[n]}{w'_{imp}[128+i]} & \begin{matrix} 192 \leqslant n < 256 \\ 0 \leqslant i < 64 \end{matrix} \end{cases} \tag{3.45}$$

La figure 3.11 montre la portion de la fenêtre w_{imp} utilisée pour l'équation 3.45.

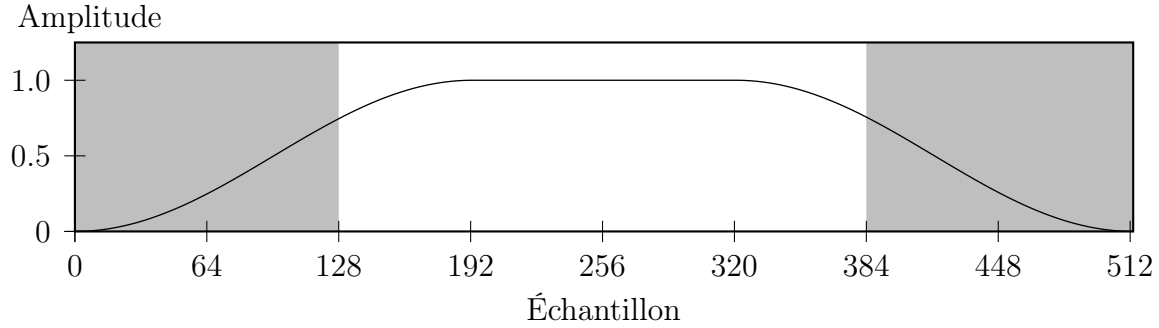


Figure 3.11 Partie de la fenêtre w_{imp} utilisée pour le calcul du gain harmonique

Avec les signaux originaux $\ddot{x}_h$ et de synthèse $\hat{\ddot{x}}_h$ qui possèdent la même forme, le modèle peut ainsi effectuer le calcul du gain d'énergie pour le signal synthèse.

Calcul du gain harmonique

L'équation 3.46 applique un gain sur le signal de synthèse $\hat{\ddot{x}}_h$ afin de s'assurer de posséder la même énergie que le signal original. L'équation 3.47 montre comment se calcule le gain d'énergie avec le signal original $\ddot{x}_h$ et le signal de synthèse $\hat{\ddot{x}}_h$.

$$\hat{\ddot{x}}_h = \hat{\ddot{x}}_h \cdot G_h \qquad n = \ [0, ..., 256[, \, n \in \mathbb{N} \tag{3.46}$$

$$G_h = \sqrt{\frac{\ddot{x}_h[n]^2}{\hat{\ddot{x}}_h[n]^2}} \qquad n = \ [0, ..., 256[, \, n \in \mathbb{N} \tag{3.47}$$

Cette section a décrit comment le modèle s'assure que le signal de synthèse harmonique résiduel $\hat{\ddot{x}}_h$ possède la même énergie que le signal original résiduel $\ddot{x}_h$. La prochaine section décrit le filtrage du signal de synthèse résiduel $\hat{\ddot{x}}_h$ ainsi que le recouvrement des segments de synthèse afin d'obtenir des trames de sortie de 12 ms.

3.4.2 Utilisation du filtre de synthèse pondéré

Cette section décrit le filtre de synthèse S_w appliqué sur le signal de synthèse résiduel harmonique $\hat{\ddot{x}}_h$. L'équation 3.48 montre le filtre de synthèse utilisé sur le signal résiduel et qui possède les mêmes valeurs de coefficients que le filtre de prédiction à court-terme P_w de l'équation 3.1.

$$S_w(z) = \frac{1}{P_w(z)} = \frac{1}{A(z/\gamma)} = \frac{1}{1 - \sum_{k=0}^{P-1} a_k \gamma^k z^{-k}} \qquad \gamma = 0.96 \quad \text{et} \quad P = 16 \tag{3.48}$$

Utilisation du filtre de synthèse

Le modèle filtre le signal harmonique résiduel $\hat{\ddot{x}}_h$ avec le filtre S_w de l'équation 3.48 afin d'obtenir le signal de synthèse de parole $\hat{s}_h$.

$$\hat{s}_h[n] = \hat{\tilde{x}}_h[n] + \sum_{i=0}^{P-1} a_p[i]\hat{\tilde{x}}_h[n-i] \qquad \begin{aligned} P = &\ 16 \\ n = &\ [0, ..., N_{\hat{s}}[,\ n \in \mathbb{N} \end{aligned} \tag{3.49}$$

Afin de conserver l'état du filtre de l'équation 3.49 pour la trame suivante $t+1$, le signal de synthèse $\hat{s}_h$ possède deux fois la longueur d'une trame ($N_{\hat{s}} = 2N_{\hat{x}}$). La figure 3.12 montre la longueur et la forme du signal de synthèse $\hat{s}$.

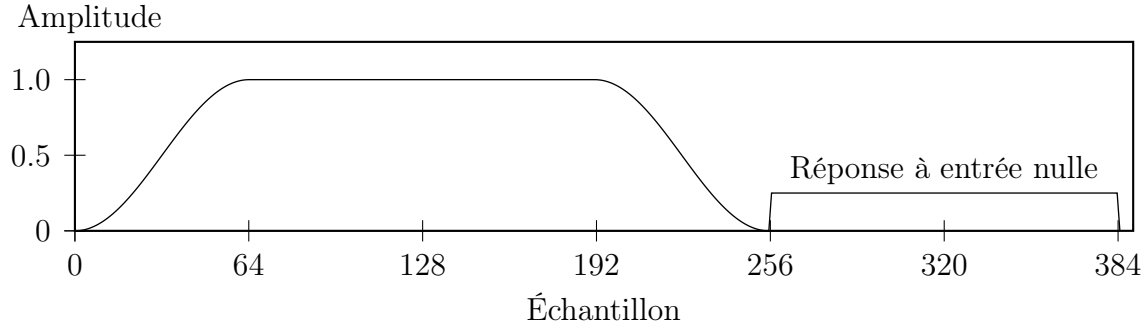


Figure 3.12 Signal de synthèse $\hat{s}$ à la sortie du filtre de synthèse

Pour obtenir le signal complet $\hat{s}$, le modèle additionne la partie harmonique $\hat{s}_h$ et la partie bruit $\hat{s}_b$ (cf. équation 3.50). Le chapitre 4 donne les détails de la quantification et de l'utilisation de générateurs de bruit pour la partie bruit $\hat{s}_b$.

$$\hat{s}[n] = \hat{s}_h[n] + \hat{s}_b[n] \qquad n = \ [0, ..., N_{\hat{s}}[,\ n \in \mathbb{N} \tag{3.50}$$

Par la suite, le modèle applique un recouvrement entre les signaux de synthèse afin d'obtenir une trame de 12 ms.

Recouvrement entre les segments de synthèse

Le segment de synthèse $\hat{s}$ possède deux fois la longueur d'une trame, car il contient également la réponse à entrée nulle du filtre de synthèse pour la prochaine trame $t+1$. Ainsi, pour obtenir une trame complète, le modèle effectue un recouvrement avec le signal de synthèse passé $\hat{s}^{-1}$ (cf. équation 3.51).

$$\hat{s}[n] = \hat{s}[n] + \hat{s}^{-1}[n+192] \qquad n = [0, ..., N_{\hat{s}}/2[,\ n \in \mathbb{N} \tag{3.51}$$

La figure 3.13 donne un exemple de recouvrement entre les segments de synthèse $\hat{s}$ et montre comment le modèle crée des trames de 12 ms avec ces segments.

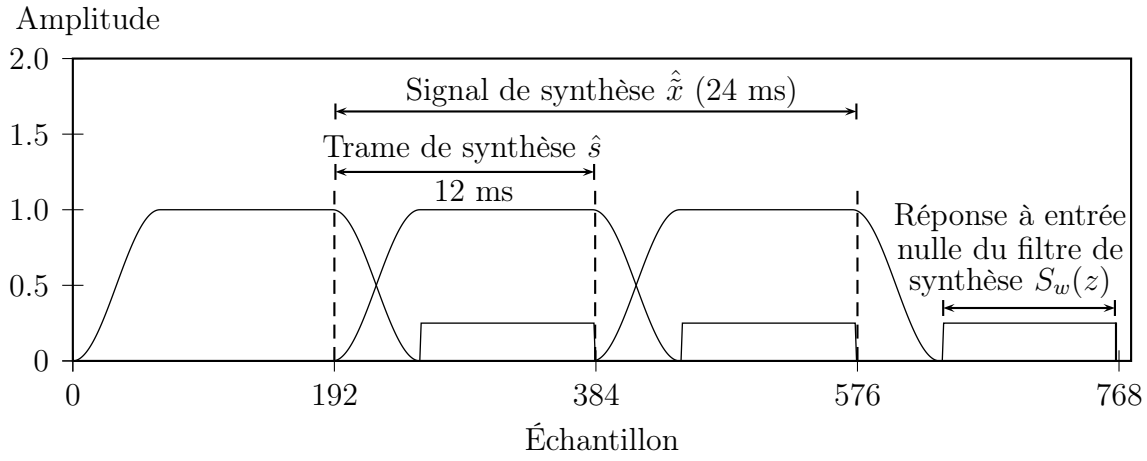


Figure 3.13 Exemple d'un recouvrement entre les fenêtres de synthèse

3.5 Conclusion du chapitre

Ce chapitre présentait les détails de fonctionnement du modèle d'analyse-synthèse développé pour cette thèse. Il expliquait comment le modèle avec des longueurs de transformée de Fourier différentes réussit à bien suivre l'évolution du pitch. Le spectre d'analyse possède un nombre de canaux plus élevé afin d'obtenir une grande précision pour l'analyse tandis que le spectre de synthèse possède un nombre de canaux plus faible, afin de bien suivre l'évolution du pitch avec une grande précision. La table précalculée du générateur d'impulsions augmente la précision du spectre de synthèse sans toutefois, augmenter la complexité des calculs. Ce chapitre présentait également les nombreuses et différentes fenêtres utilisées sur les signaux afin d'obtenir une bonne synchronisation entre l'analyse et la synthèse ainsi qu'un recouvrement parfait entre les trames de sortie. Les tests subjectifs MUSHRA (***MU****ltiple* ***S****timuli with* ***H****idden* ***R****eference and* ***A****nchor*) du chapitre 5 démontrent que le signal de synthèse du modèle de ce chapitre possède une qualité perceptuelle transparente.

CHAPITRE 4

CODEC À PARTIR DU MODÈLE D'ANALYSE-SYNTHÈSE

Le dernier chapitre présentait un modèle d'analyse-synthèse pour la compression des signaux de parole qui fonctionne entièrement dans le domaine de la transformée de Fourier. Les tests du chapitre 5 montrent que le modèle réussit à obtenir un signal de synthèse avec une qualité perceptuelle transparente sans nécessairement suivre la forme d'onde du signal original. Ce chapitre propose un codec développé à partir de ce modèle d'analyse-synthèse. Les tests du chapitre 5 montrent que le codec proposé dans ce chapitre obtient une bonne qualité avec un score de 4 sur 5 sur une échelle MOS (***M**ean **O**pinion **S**core*) pour des débits de 24 kbit/s et 30 kbit/s.

Organisation du chapitre

Ce chapitre commence par une description générale du modèle quantifié (codec) avec tous les paramètres que l'encodeur transmet au décodeur. Par la suite, le chapitre se divise en trois grandes sections afin de décrire les techniques de compression et de quantification développées pour les différentes parties : harmonique, bruit des spectres mixtes et bruit des spectres non-harmoniques.

Dans ce chapitre, le modèle quantifié propose une méthode afin de réduire le nombre de phases à transmettre au décodeur sans toutefois affecter le nombre de phases dans le spectre de synthèse. De plus, avec ces phases transmises, le modèle leur applique une quantification prédictive et introduit par le fait même la prédiction long-terme dans le domaine des fréquences. Finalement, ce chapitre définit une nouvelle partie appelée de transition pour les spectres mixtes et qui commence après la fréquence de coupure. La bande de transition se situe entre la partie harmonique et la partie bruit des spectres mixtes.

4.1 Description des paramètres transmis au décodeur

Le modèle développé possède la particularité d'avoir un nombre de degrés de liberté élevé dû à ses nombreux paramètres. Ainsi, il existe un grand nombre de configurations possibles et ce chapitre présente un exemple de configuration. Le tableau 4.1 présente tous les

paramètres du modèle en les segmentant en trois groupes de paramètres distincts : toujours transmis, pour la synthèse de la partie harmonique et pour la synthèse de la partie bruit.

Tableau 4.1 Paramètres transmis au décodeur

Toujours transmis	Partie harmonique	Partie bruit
Coefficients du filtre LPC	Amplitudes des partiels α	Coefficients du spectre $\mathsf{X_b}$
Nombre de partiels trouvé N_h	Phases des partiels φ	Gains d'énergie des générateurs de bruit
Fréquence fondamentale ω_0		
Etat du SFM (***S**pectral **F**latness **M**easure*)		
Gains temporels du signal de synthèse		

Le premier groupe du tableau 4.1 contient les paramètres que l'encodeur transmet en tout temps au décodeur. Ce groupe contient les coefficients du filtre de prédiction, le nombre de partiels trouvé N_h dans le spectre d'analyse et la fréquence fondamentale calculée ω_0. C'est avec le nombre de partiels N_h et la fréquence fondamentale ω_0 que le décodeur détermine le type de spectre qu'il synthétisera, c'est-à-dire un spectre harmonique, mixte ou non-harmonique. De plus, l'encodeur transmet l'état du SFM (***S**pectral **F**latness **M**easure*) qui détermine le type de compression et de quantification à effectuer sur la partie bruit de tous les types de spectres. Finalement, ce groupe contient des valeurs de gain temporel afin d'ajuster le signal de synthèse. C'est le seul groupe de paramètres du tableau 4.1 qui possède un débit constant lors de la transmission au décodeur.

Le second groupe du tableau 4.1 représente les paramètres nécessaires pour la synthèse de la partie harmonique du spectre. Pour créer cette partie, l'encodeur utilise les valeurs d'amplitudes α et de phases φ des partiels trouvés durant l'analyse ainsi que la fréquence fondamentale ω_0 déjà transmise par le premier groupe décrit précédemment. Le nombre de paramètres à transmettre varie selon le degré de voisement du spectre.

Le troisième et dernier groupe du tableau 4.1 contient les paramètres utilisés afin de créer la partie bruit du spectre de synthèse. Ce groupe possède également un nombre variable de paramètres comme pour la partie harmonique du spectre. L'encodeur transmet un certain nombre de canaux de la transformée de Fourier afin de représenter des parties du spectre considérées plus tonales et par conséquent, plus difficiles à modéliser avec des générateurs

de bruit. De plus, le modèle transmet des gains d'énergie afin d'ajuster l'enveloppe du spectre de bruit provenant des générateurs.

Il est important de mentionner que le modèle applique une configuration différente pour la partie bruit provenant d'un spectre mixte et d'un bruit provenant d'un spectre non-harmonique. Le modèle détermine que la partie bruit provient d'un spectre non-harmonique lorsque la fréquence fondamentale ω_0 possède une valeur de zéro. Le modèle applique des configurations distinctes pour ces différents parties bruits, mais les principes de modélisation et de quantification restent les mêmes tels que l'utilisation de générateurs de bruit. La prochaine section présente le schéma de fonctionnement général du modèle au décodeur.

4.1.1 Schéma de fonctionnement général au décodeur

Au niveau du décodeur, le modèle sépare le processus de synthèse pour la partie harmonique et la partie non-harmonique afin qu'ils puissent fonctionner en parallèle (cf. figure 4.1). Le décodeur utilise une configuration modulaire afin d'offrir une plus grande flexibilité au modèle pour de futures modifications possibles.

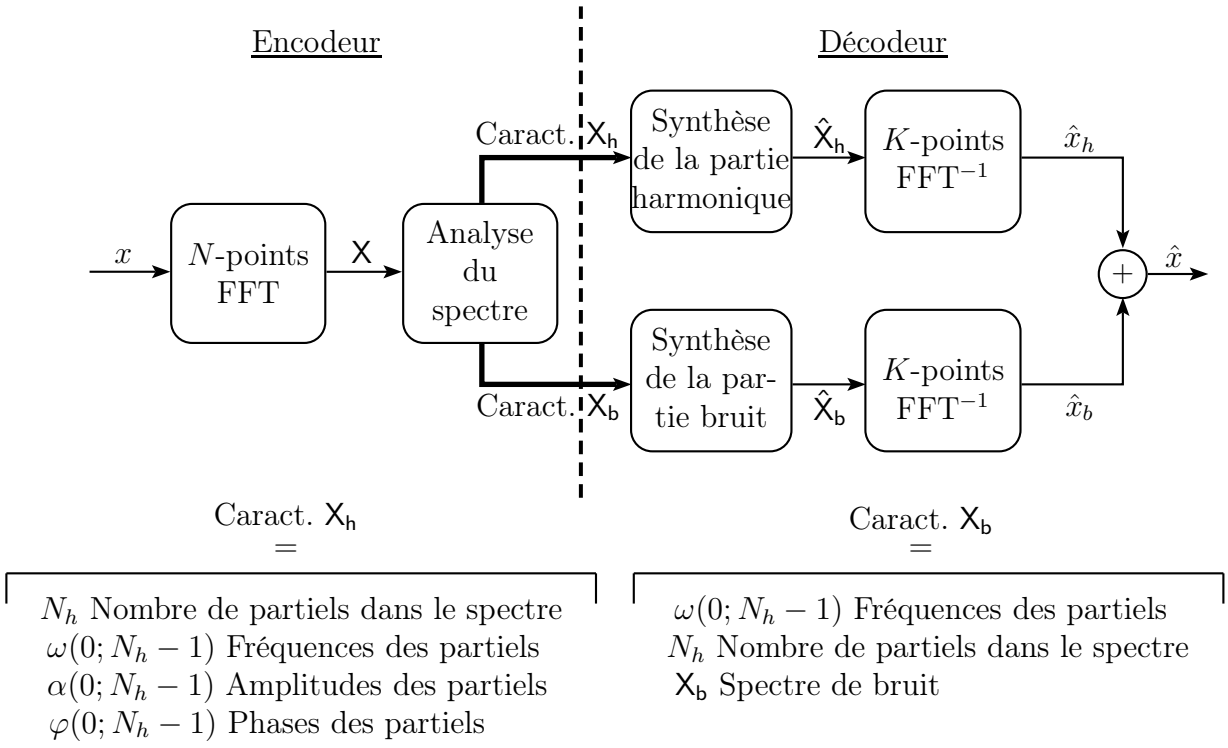


Figure 4.1 Schéma de fonctionnement général du modèle quantifié

Pour séparer le spectre en une partie harmonique $\mathsf{X_h}$ et une partie bruit $\mathsf{X_b}$, le modèle utilise la fréquence de coupure ω_c qui se calcule avec l'équation 4.1. La variable ω_0 dans l'équation 4.1 représente la fréquence fondamentale.

$$\omega_c = (\omega_0 \cdot N_h) + \frac{\omega_0}{2} \tag{4.1}$$

La synthèse de la partie bruit de la figue 4.1 utilise le spectre de bruit original $\mathsf{X_b}$ afin de calculer des gains d'énergie. Le spectre de bruit original $\mathsf{X_b}$ s'obtient durant le processus de calcul du gain d'énergie harmonique G_h. Pour plus de détails, voir la section 3.4.1 du chapitre de la description du modèle d'analyse-synthèse.

Pour la création du codec à partir du modèle d'analyse-synthèse, l'encodeur applique des techniques de quantification vectorielle sur les paramètres à transmettre. Le modèle utilise l'algorithme des K-moyennes afin d'obtenir des dictionnaires quasi-optimaux.

4.1.2 Brève description de la quantification vectorielle

La quantification permet l'approximation d'une valeur par une valeur appartenant à un ensemble dénombrable et à toutes fins pratiques fini. Ainsi, la quantification vectorielle représente tous les vecteurs v de dimension M par un vecteur $\hat{v}_i$ de même dimension, mais qui appartient à un ensemble fini C contenant L vecteurs. Les vecteurs $\hat{v}_i$ se nomment des représentants et le nombre L de représentants dépend du nombre alloué de bits au dictionnaire C.

Ainsi, une quantification vectorielle de dimension M et de taille L se définit comme une application Q de $\mathbb{R}^M$ vers C [Gersho, 1992] :

$$Q : \mathbb{R}^M \to C \qquad C = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L) \tag{4.2a}$$

$$\hat{v}_i \in \mathbb{R}^M \text{ pour chaque } i \in \mathcal{J} \equiv \{1, 2, ..., L\}$$

$$\text{Ainsi } Q(v) = \hat{v}_i \qquad C = \{\hat{v}_i \in \mathbb{R}^M : i = 1, 2, ..., L\} \tag{4.2b}$$

L'utilisation de la quantification Q dans l'espace $\mathbb{R}^M$ crée des partitions R_i de l'espace en L régions, R_i pour $i \in \mathcal{J}$. Ces régions R_i appelées cellules ou régions de Voronoï sont déterminées par [Gersho, 1992] :

$$R_i = \{v \in \mathbb{R}^M : Q(v) = \hat{v}_i\} \tag{4.3}$$

Les régions de Voronoï R_i de l'équation 4.3 possèdent les caractéristiques suivantes [Gersho, 1992] :

$$\bigcup_{i=1}^{L} R_i = \mathbb{R}^M \tag{4.4}$$

et

$$R_i \bigcap R_j = \varnothing \text{ pour } i \neq j \tag{4.5}$$

Cette section a donné une brève description de la quantification vectorielle en définissant les caractéristiques et le contenu d'un dictionnaire. La prochaine partie explique la quantification vectorielle de type gain-forme que le modèle quantifié utilise.

Quantification vectorielle gain-forme

Le modèle quantifié utilise fréquemment de la quantification de type gain-forme sur les paramètres à transmettre pour réduire l'étendue des dictionnaires, et ainsi les rendre plus efficaces. Pour diminuer cette étendue, le modèle applique un gain sur les vecteurs v à quantifier afin de normaliser les vecteurs $\underline{v}$. L'application de ce gain réduit l'étendue du dictionnaire et augmente l'efficacité du taux de compression. Lors de la quantification vectorielle gain-forme, le modèle utilise l'équation 4.6 pour calculer le gain des vecteurs v de dimension M.

$$G_{\text{norm}} = \frac{1}{M}\sqrt{\sum_{i=0}^{M-1} v_i^2} \tag{4.6}$$

Par la suite, le modèle applique le gain G_{norm} sur le vecteur v afin d'obtenir le vecteur normalisé $\underline{v}$ comme le montre l'équation 4.7.

$$\underline{v} = \frac{v}{G_{\text{norm}}} \tag{4.7}$$

Ainsi, la quantification vectorielle gain-forme utilise deux dictionnaires pour représenter les vecteurs v : un dictionnaire pour les vecteurs normalisés et un dictionnaire pour les gains de normalisation. Au niveau du décodeur, le modèle utilise l'équation 4.8 afin de retrouver les valeurs du vecteur quantifiées $\hat{v}$.

$$\hat{v} = \hat{\underline{v}} \cdot \hat{G}_{\text{norm}} \tag{4.8}$$

Cette section du document a donné la structure des dictionnaires utilisée dans le modèle. La prochaine section donne les détails de la construction d'un dictionnaire optimal à l'aide de l'algorithme des K-moyennes.

Création du dictionnaire optimal avec l'algorithme des K-moyennes

Afin d'obtenir des dictionnaires optimaux, le modèle utilise l'algorithme des K-moyennes. Cet algorithme construit un dictionnaire optimal à partir d'une séquence d'apprentissages qui doit représenter statiquement la source à coder. La figure 4.2 montre les étapes de l'algorithme des K-moyennes.

L'algorithme décrit à la figure 4.2 divise les observations en régions de Voronoï, où chaque observation appartient à la région R_i ayant la moyenne la plus proche. C'est un algorithme de type itératif non supervisé qui commence par un dictionnaire initial. À chaque itération, l'algorithme applique deux opérations : la classification du nouveau vecteur v et l'optimisation avec un recalcul des centroïdes $\hat{v}_i$.

La première opération consiste à classifier le nouveau vecteur v en suivant la règle des plus proches voisins. Cette opération classe le nouveau vecteur v afin de déterminer à quelle région de Voronoï R_i il appartient, parmi tous les représentants $\hat{v}_i$. La seconde opération effectue un recalcul des représentants $\hat{v}_i$ puisque l'ajout d'un nouveau vecteur v dans une région R_i modifie les valeurs des représentants $\hat{v}_i$.

Puisque chaque itération modifie la structure du dictionnaire, le modèle calcule les valeurs de distorsion de chaque région (somme de la différence entre l'observation et le centroïde) afin de déterminer une réduction ou non de la distorsion moyenne. L'algorithme

1- Initialisation du dictionnaire $\{\hat{v}_1^0, v_2^0, \ldots \hat{v}_i^0 \ldots, \hat{v}_L^0\}$

2- Classification des vecteurs v avec le centroïde le plus proche

$$d(v, \hat{v}_i) = \|v - \hat{v}_i\|^2 = \sum_{j=1}^{M} (v(j) - \hat{v}_i(j))^2$$

3- Optimisation par le recalcul des centroïdes

$$\hat{v}_i = \frac{\sum\limits_{v_j \in C_i} v_j}{\sum\limits_{v_j \in C_i} 1} = \text{moyenne des vecteurs } v_i \text{ dans la région } i$$

4- Calcul de la distorsion $D^k = \sum\limits_{i=1}^{L} \sum\limits_{v_j \in C_i} \|v_j - C_i\|^2$

5- Condition d'arrêt : $\left| \frac{D^k - D^{k-1}}{D^k} \right| \leq$ seuil

Sortie : Dictionnaire $\{\hat{v}_1, \hat{v}_2, \ldots \hat{v}_i \ldots, \hat{v}_L\}$

Figure 4.2 Algorithme des K-moyennes

des K-moyennes s'arrête lorsque la différence de distorsion entre la distorsion actuelle et la distorsion précédente se situe sous un seuil préétabli.

Les prochaines sections donnent les détails sur les techniques de compression et de quantification effectuées sur les différents paramètres transmis au décodeur en commençant par les paramètres toujours transmis.

4.2 Compression des paramètres toujours transmis

Le tableau 4.2 montre les paramètres toujours transmis au décodeur. Le nombre de paramètres transmis reste toujours constant d'une analyse à l'autre contrairement aux paramètres des parties harmonique et bruit.

Chaque paramètre du tableau 4.2 utilise des dictionnaires de quantification différents. Le modèle utilise des dictionnaires de quantification vectorielle de dimension 16 pour les coefficients du filtre LPC et un dictionnaire de dimension 5 pour les gains temporels. Le vecteur des gains temporels contient un gain pour la partie harmonique et quatre gains pour la partie bruit.

Tableau 4.2 Liste des paramètres toujours transmis au décodeur

Coefficients a_k du filtre LPC (16 coefficients)
Nombre de partiels trouvé N_h
Fréquence fondamentale ω_0
État du SFM (***S**pectral **F**latness **M**easure*) (0 ou 1)
Gains temporels du signal de synthèse (5 gains)

Les autres paramètres du tableau 4.2 utilisent des quantifications scalaires. Le tableau 4.3 indique les cinq dictionnaires développés pour les paramètres toujours transmis du tableau 4.2.

Tableau 4.3 Description des dictionnaires des paramètres toujours transmis

Dictionnaire pour les 16 coefficients a_k :
$Q : \mathbb{R}^M \rightarrow C_{\text{lpc}}$ $\quad C_{\text{lpc}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 16$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Dictionnaire pour le nombre de partiels trouvé N_h :
$Q : \mathbb{R}^M \rightarrow C_{N_h}$ $\quad C_{N_h} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où M=1 $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Dictionnaire pour la fréquence fondamentale ω_0 :
$Q : \mathbb{R}^M \rightarrow C_{\omega_0}$ $\quad C_{\omega_0} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 1$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Dictionnaire pour l'état du SFM :
$Q : \mathbb{R}^M \rightarrow C_{\text{sfm}}$ $\quad C_{\text{sfm}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 1$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Dictionnaire pour les 5 gains temporels :
$Q : \mathbb{R}^M \rightarrow C_{G_t}$ $\quad C_{G_t} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 5$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

La prochaine section donne les détails sur les techniques de compression effectuées sur les paramètres de la partie harmonique du spectre.

4.3 Compression de la partie harmonique

Cette section présente les techniques de compression proposées pour les paramètres de la partie harmonique du spectre de synthèse. Pour la création des partiels, le décodeur utilise : les valeurs des phases φ, les valeurs d'amplitudes α, la fréquence fondamentale ω_0 et le

nombre de partiels N_h. Cette section présente particulièrement les techniques développées pour les valeurs des phases φ et des amplitudes α puisque la section précédente a déjà présenté les techniques pour la fréquence fondamentale ω_0 et le nombre de partiels N_h.

Une grande particularité du modèle consiste sur le fait, que le décodeur crée entièrement le spectre harmonique. Au début de la synthèse, le spectre possède des valeurs nulles et c'est le générateur d'impulsions qui ajoute les partiels. Le modèle d'analyse-synthèse du chapitre précédent utilisait toutes les valeurs des phases et des amplitudes trouvées pour la création du spectre. D'ailleurs, les tests subjectifs du chapitre 5 démontrent que le signal de sortie du modèle d'analyse-synthèse possède une qualité perceptuelle transparente. Afin de réduire le débit du modèle quantifié, la prochaine partie propose une méthode de réduction du nombre de phases à transmettre au décodeur qui affecte peu la qualité du signal de synthèse.

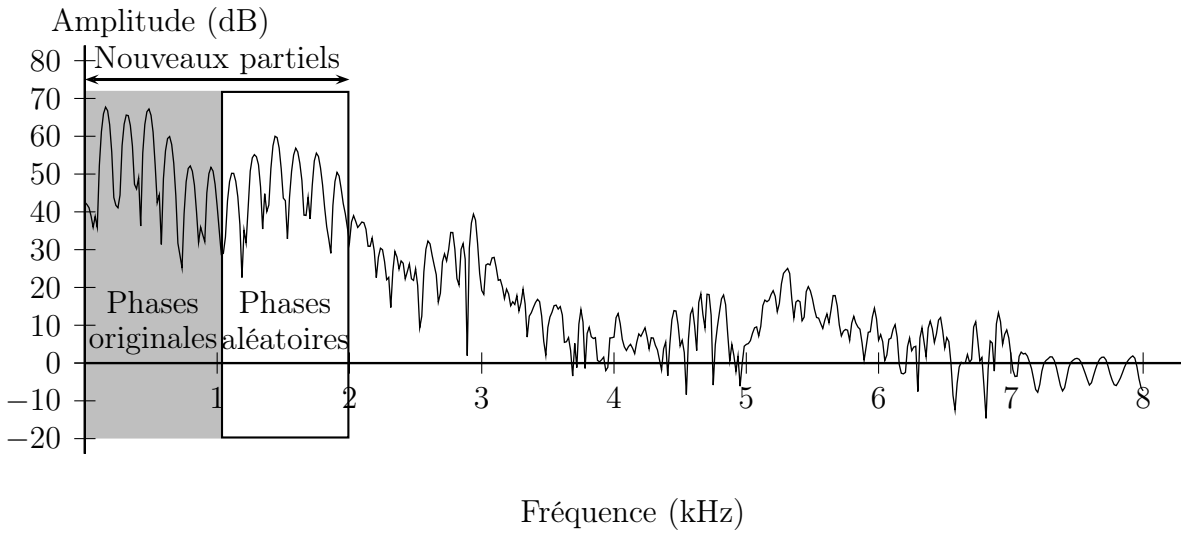
4.3.1 Méthode développée pour diminuer le nombre de phases transmis

Le modèle d'analyse-synthèse du chapitre précédent utilisait toutes les valeurs des phases φ et des amplitudes α originales afin de créer le spectre de synthèse harmonique $\hat{\mathsf{X}}_\mathsf{h}$. Cependant, des expérimentations sur les phases du modèle démontrent quelques particularités, dont la moindre importance des valeurs absolues (réelles) sur l'impact de la qualité perceptuelle du signal de synthèse. Cette partie du document présente une méthode de réduction du nombre de phases à transmettre qui résulte de toutes les expérimentations effectuées sur les phases.

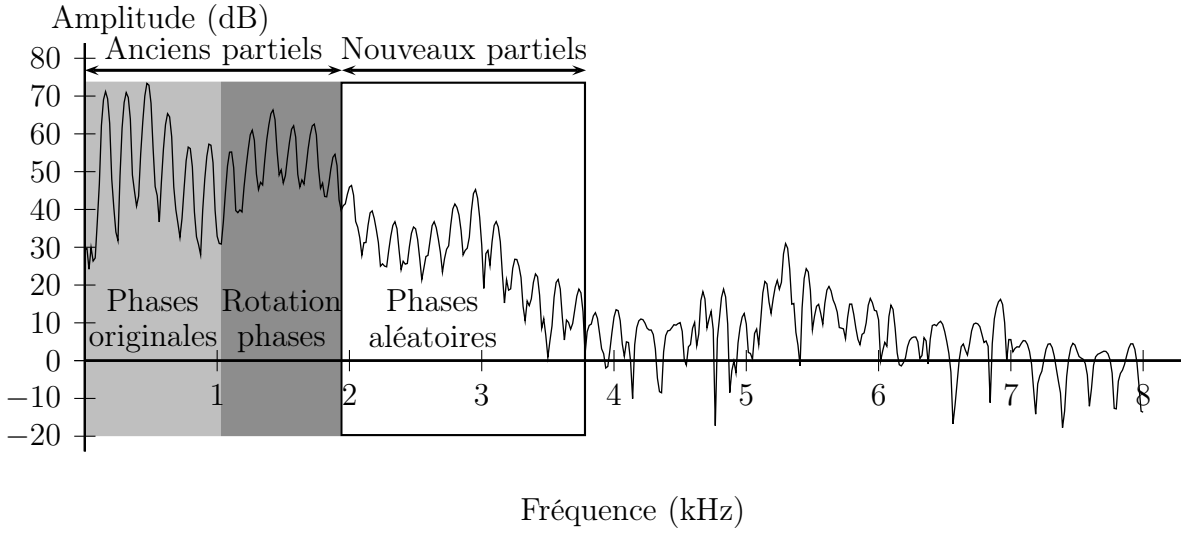
Méthode proposée pour diminuer le nombre de phases à transmettre

L'idée générale de cette méthode consiste à transmettre un nombre minimum de phases originales au décodeur et que le reste des phases s'obtiennent par des valeurs aléatoires ou bien par extrapolation. Pour déterminer les phases qui possèderont des valeurs aléatoires ou extrapolées, le modèle utilise le concept de nouveaux et d'anciens partiels sur le spectre. Ainsi, le modèle attribue des valeurs de phases aléatoires pour les nouveaux partiels et extrapole les valeurs des anciens partiels. La figure 4.3 montre un exemple de spectres avec ce concept.

La figure 4.3(a) montre la situation d'un spectre avec uniquement des nouveaux partiels. Cette situation survient lorsque le spectre précédent $t-1$ est non-harmonique. Durant l'analyse de la figure 4.3(a), le modèle trouve $N_h = 12$ nouveaux partiels dans le spectre.



(a) Premier spectre mixte après un spectre non-harmonique (au temps t)



(b) Spectre mixte suivant le spectre (a) (au temps $t + 1$)

Figure 4.3 Exemple du concept de nouveaux et d'anciens partiels dans un spectre

Ainsi, le modèle d'analyse-synthèse du chapitre précédent transmettrait toutes les valeurs des phases trouvées tandis que le modèle quantifié de ce chapitre transmet uniquement les $N_\varphi = 6$ premières phases du spectre (cf. équation 4.9) au décodeur. La valeur $N_\varphi = 6$ provient de plusieurs écoutes de signaux et représente le nombre minimal de phases afin d'obtenir une bonne qualité perceptuelle lors de l'écoute. De plus, la figure 4.3(a) montre que le décodeur utilise des valeurs aléatoires, entre $-\pi$ et π, pour les phases des partiels qui se situe au-delà de $N_\varphi = 6$ partiels.

$$\varphi'[i] = \varphi[i] \qquad i = [0, ..., N_\varphi[, i \in \mathbb{N} \tag{4.9}$$

La figure 4.3(b) représente l'analyse qui suit le spectre de la figure 4.3(a) au temps $t + 1$ et qui montre la présence d'anciens partiels et de nouveaux partiels. Comme mentionné précédemment, le modèle ne transmet que les $N_\varphi = 6$ premières phases du spectre de la figure 4.3(a). Pour les anciens partiels suivants $N_\varphi = 6$, le décodeur leur applique une rotation de phase entre $-\pi$ et π comme le montre l'équation 4.12. De plus, comme pour la figure 4.3(a), le décodeur attribue des valeurs aléatoires qui se situent entre $-\pi$ à π aux nouveaux partiels.

La valeur de la rotation entre les trames se calcule à l'aide de la longueur de la trame FRAME et de la longueur du spectre de synthèse K (cf. équation 4.10).

$$\mathsf{rot} = \frac{\pi}{2} \cdot \frac{\text{FRAME}}{K\text{-point FFT}} \qquad \mathsf{rot} = \text{Valeur de la rotation entre 2 trames} \tag{4.10}$$

$$\Gamma_\varphi = \mathsf{rot}\Big(\omega[i] + \omega^{-1}[i]\Big) \qquad i = [N_\varphi, ..., N_h^{-1}[, i \in \mathbb{N} \tag{4.11}$$

$$\varphi'[i] = \Big[\Big(\varphi'^{-1}[i] + \Gamma_\varphi\Big) \mod 2\pi\Big] - \pi \qquad i = [N_\varphi, ..., N_h^{-1}[, i \in \mathbb{N} \tag{4.12}$$

De nombreuses écoutes de fichiers démontrent que la méthode proposée pour diminuer le nombre de phases à transmettre affecte peu la qualité du signal de synthèse. L'oreille possède une grande sensibilité pour les valeurs absolues des phases en basse fréquence, mais que par la suite, les valeurs absolues possèdent une moindre importance. Au niveau des hautes fréquences, l'oreille humaine possède une plus grande sensibilité à la cohérence des phases entre les trames plutôt qu'à leurs valeurs absolues.

La méthode décrite dans cette partie réduit le nombre de phases à transmettre et la prochaine section optimise la quantification de ces phases en y proposant une quantification vectorielle prédictive.

4.3.2 Quantification prédictive proposée pour les phases

La section précédente proposait une méthode de diminution du nombre de phases à transmettre au décodeur, mais qui n'affecte pas le nombre de phases dans le spectre de synthèse. Cette section propose une quantification prédictive de ces phases transmises, une technique largement utilisée avec les modèles de codage de parole temporel. L'utilisation de la transformée de Fourier permet d'introduire cet outil de prédiction long-terme dans le domaine fréquentiel avec les phases. Ainsi, lorsqu'il est possible, le modèle effectue une quantification vectorielle prédictive avec les phases des anciens partiels et une quantification vectorielle absolue pour les phases des nouveaux partiels.

Prédiction des phases des anciens partiels

Pour la quantification prédictive des phases, le modèle transmet l'erreur de prédiction au lieu de transmettre la valeur absolue. Les équations 4.14 montrent les étapes pour obtenir l'erreur de prédiction $e_\varphi[i]$. La valeur de rotation rot entre deux trames se calcule avec l'équation 4.13 et nécessite la longueur de la trame $FRAME$ ainsi que la longueur du spectre de synthèse K.

$$\mathsf{rot} = \frac{\pi}{2} \cdot \frac{\text{FRAME}}{K\text{-point FFT}} \qquad \mathsf{rot} = \text{Valeur de la rotation entre 2 trames} \tag{4.13}$$

$$\Gamma_\varphi = \mathsf{rot}\Big(\omega[i] + \omega^{-1}[i]\Big) \tag{4.14a}$$

$$\varphi_p[i] = \Big[\Big(\varphi^{-1}[i] + \Gamma_\varphi\Big) \mod 2\pi\Big] - \pi \tag{4.14b}$$

$$e_\varphi[i] = \Big[\Big(\varphi[i] - \varphi_p[i]\Big) \mod 2\pi\Big] - \pi \tag{4.14c}$$

$$\varphi'[i] = \varphi_p[i] + e_\varphi[i] \tag{4.14d}$$

Au niveau du décodeur, la valeur de la phase $\varphi_p(i)$ s'obtient en additionnant l'erreur de prédiction $e_\varphi(i)$ avec la phase prédite $\varphi_p(i)$ (cf. équation 4.14d). Le décodeur prédit uniquement les $N_\varphi = 6$ premières phases qui proviennent d'anciens partiels. La prochaine partie donne une description générale des dictionnaires de quantification utilisés pour les phases des anciens et des nouveaux partiels.

Description des dictionnaires utilisés pour les phases

Pour la quantification vectorielle prédite et absolue des $N_\varphi = 6$ premières phases, le modèle utilise des vecteurs v de dimension $M = 3$. Le modèle possède deux dictionnaires pour la quantification avec prédiction ($C_{e_{\varphi_1}}$ et $C_{e_{\varphi_2}}$) et deux dictionnaires pour la quantification absolue (C_{φ_1} et C_{φ_2}). Les dictionnaires prédictifs et absolus utilisent la même configuration, ce qui signifie que les premiers dictionnaires ($C_{e_{\varphi_1}}$ et C_{φ_1}) quantifient les trois premières phases et que les seconds dictionnaires ($C_{e_{\varphi_2}}$ et C_{φ_2}) quantifient les trois phases suivantes. Le tableau 4.4 définit les dictionnaires utilisés pour les phases.

Tableau 4.4 Description des dictionnaires pour les phases

Dictionnaire absolu pour les trois premières phases :
$Q : \mathbb{R}^M \to C_{\varphi_1}$ $\quad C_{\varphi_1} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 3$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire absolu pour les trois phases suivantes :
$Q : \mathbb{R}^M \to C_{\varphi_2}$ $\quad C_{\varphi_2} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où M=3 $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire prédictif pour les trois premières phases :
$Q : \mathbb{R}^M \to C_{e_{\varphi_1}}$ $\quad C_{e_{\varphi_1}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 3$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire prédictif pour les trois phases suivantes :
$Q : \mathbb{R}^M \to C_{e_{\varphi_2}}$ $\quad C_{e_{\varphi_2}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
où $M = 3$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Lorsque le nombre de phases ne suffit pas pour remplir correctement les vecteurs v de quantification du tableau 4.4, le modèle utilise des algorithmes pour les remplir. Il est important de mentionner que le premier vecteur est toujours complet puisque le modèle déclare un spectre harmonique seulement s'il possède un minimum de trois partiels.

Le second vecteur de prédiction est incomplet

Cette première situation survient lorsque le modèle ne peut compléter le second vecteur de prédiction. Le modèle vérifie en premier, s'il existe des nouveaux partiels afin de remplir un second vecteur en valeur absolue et utilise par la suite, le second dictionnaire de phases absolues C_{φ_2}.

S'il n'existe pas de nouveaux partiels, le modèle répète le dernier élément valide $e_{\varphi}(i-1)$ du vecteur des erreurs de phases afin de compléter celui-ci. Par la suite, le modèle utilise le second dictionnaire prédictif $C_{e_{\varphi_2}}$ des phases pour la quantification.

Le second vecteur avec des valeurs absolues est incomplet

Cette seconde situation survient lorsque le second vecteur de phases absolues C_{φ_2} ne contient pas assez d'éléments pour compléter le vecteur. Ainsi, pour compléter le vecteur, le modèle répète la dernière valeur de phase valide $\varphi(i-1)$ et utilise le second dictionnaire absolu C_{φ_2} pour la quantification.

La gestion des éléments dans les vecteurs v de quantification complète la section dédiée aux techniques de compression et de quantification développées pour les phases. La prochaine section décrit les techniques développées pour les amplitudes des spectres harmoniques.

4.3.3 Compression des amplitudes

Cette section donne les détails de la compression effectuée sur les amplitudes trouvées dans le spectre durant l'analyse. D'après de nombreux tests d'écoute afin d'obtenir un signal de synthèse de qualité, le modèle utilise toutes les valeurs des amplitudes trouvées durant l'analyse, en limitant toutefois son nombre à 46 amplitudes. Lorsqu'il y a plus de 46 amplitudes de partiels, le modèle utilise une technique inspirée de la méthode SBR (***S****pectral* ***B****and* ***R****eplication*).

Quantification des 46 premières amplitudes

Pour la quantification des 46 premières amplitudes, le modèle regroupe les partiels du spectre en sous-bande avec différentes largeurs afin de tenir compte de la sensibilité de l'oreille en basse fréquence. L'encodeur offre une plus grande résolution pour les amplitudes en basse fréquence, avec des largeurs de sous-bande plus petite et avec plus de bits pour les dictionnaires. L'encodeur diminue graduellement cette résolution vers les hautes fréquences. La figure 4.4 montre comment le modèle regroupe les amplitudes des partiels du spectre en cinq sous-bandes (cinq dictionnaires) pour la quantification des amplitudes α.

La figure 4.4 indique également les différentes dimensions des vecteurs de quantification v utilisés pour les différentes sous-bandes. Lorsqu'un vecteur v ne peut pas être complété par manque de partiels, le modèle complète le vecteur en répétant la dernière valeur d'amplitude valide $\alpha(i-1)$.

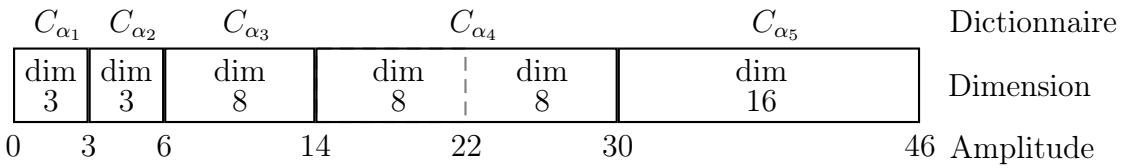


Figure 4.4 Organisation des amplitudes pour les dictionnaires de quantification

Pour diminuer l'étendu des données et la taille du dictionnaire, le modèle utilise la quantification vectorielle gain-forme décrit précédemment dans la section 4.1.2. Ainsi, la quantification des amplitudes s'effectue avec deux dictionnaires : un dictionnaire normalisé et un dictionnaire de gains de normalisation. L'équation 4.15 montre le calcul du gain de normalisation effectué sur chaque vecteur v d'amplitudes de dimension M.

$$G_\alpha = \frac{1}{M}\sqrt{\sum_{i=0}^{M-1} v_i^2} \tag{4.15}$$

Le tableau 4.5 montre les cinq dictionnaires normalisés et le dictionnaire de gains de normalisation utilisés pour la quantification des amplitudes.

Cette section a présenté la quantification appliquée sur les 46 premières amplitudes des partiels trouvés durant l'analyse. Lorsque le spectre possède plus de 46 partiels, le modèle utilise une technique inspirée de la méthode SBR, proposée auparavant par les codeurs perceptuels par transformée et décrite dans le chapitre 2 dans la section 2.3.3.

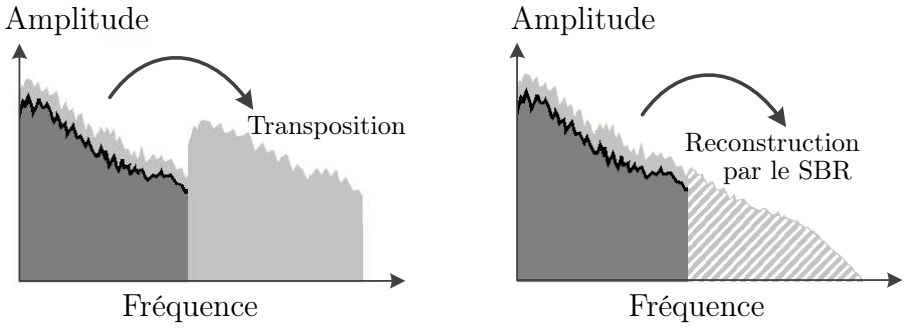
Utilisation d'une technique inspirée de la méthode SBR lors de plus de 46 amplitudes

Le modèle utilise une technique inspirée de la méthode SBR (***S**pectral **B**and **R**eplication*) afin d'obtenir les valeurs des amplitudes qui se situe au-delà de 46 partiels. La figure 4.5 [Meltzer et Moser, 2006] montre le principe de reconstruction de la bande appelée SBR qui a été proposé par les modèles de codage perceptuel par transformée.

Le modèle utilise le même principe de la figure 4.5 en recopiant les valeurs des amplitudes d'une partie de la bande basse vers la bande haute. Le modèle applique également des facteurs de gains sur les amplitudes de la bande recopiée. Chaque amplitude recopiée possède son facteur de gain qui provient d'une table précalculée. Cette table contient des facteurs de gain d'amplitudes qui représente la moyenne des ratios de l'énergie du partiel

Tableau 4.5 Description des dictionnaires pour les amplitudes

Dictionnaire des gains de normalisation des amplitudes
$Q : \mathbb{R}^M \rightarrow C_{\text{norm}_\alpha}$ $\quad C_{\text{norm}_\alpha} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
dimension $M = 5$ $\quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire normalisé pour les amplitudes (position de 1 à 3) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{\alpha_1}$ $\quad \underline{C}_{\alpha_1} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 3$ $\quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire normalisé pour les amplitudes (position de 4 à 6) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{\alpha_2}$ $\quad \underline{C}_{\alpha_2} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 3$ $\quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire normalisé pour les amplitudes (position de 7 à 14) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{\alpha_3}$ $\quad \underline{C}_{\alpha_3} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8$ $\quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire normalisé pour les amplitudes (position de 15 à 30) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{\alpha_4}$ $\quad \underline{C}_{\alpha_4} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8$ $\quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire normalisé pour les amplitudes (position de 31 à 46) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{\alpha_5}$ $\quad \underline{C}_{\alpha_5} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 16$ $\quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$



a) Reconstruction des hautes fréquences par transposition

b) Ajustement de l'enveloppe dans les hautes fréquences

Figure 4.5 Principe de la reconstruction de la bande spectrale (SBR)

par rapport à l'énergie totale moyenne de tous les partiels du spectre. Le calcul du facteur de gain s'est effectué sur une grande base de données et représente la moyenne de tous ces rapports d'énergie. La description de la technique inspirée de la méthode SBR développée

pour les amplitudes, complète la partie de ce chapitre qui définit les paramètres à quantifier pour la partie harmonique du spectre.

4.3.4 Conclusion sur la partie harmonique

Cette section du chapitre décrivait des techniques de compression utilisées sur la partie harmonique du spectre afin de diminuer l'information à transmettre au décodeur. La partie harmonique se compose de deux paramètres à transmettre au décodeur : les phases φ et les amplitudes α des partiels.

Pour la compression des phases, le modèle propose une méthode afin de diminuer le nombre de phases à transmettre sans toutefois réduire le nombre de phases total dans le spectre de synthèse. Cette diminution du nombre de phases permet d'obtenir une bonne qualité perceptuelle du signal de synthèse. De plus, le modèle propose également l'introduction de la prédiction long-terme dans le domaine de la transformée de Fourier avec une quantification prédictive des phases.

Cette section a également présenté la quantification vectorielle gain-forme effectuée sur les 46 premières amplitudes du spectre. Pour les amplitudes suivantes, le modèle utilise une technique inspirée de la méthode SBR (***S**pectral **B**and **R**eplication*) [Meltzer et Moser, 2006] proposé auparavant par les modèles de codage perceptuel par transformée du chapitre 2.

4.4 Compression de la partie bruit des spectres mixtes

Cette section du document présente les techniques de compression appliquées sur la partie bruit provenant de spectres mixtes. Le modèle différencie le bruit provenant de spectres mixtes et le bruit provenant de spectres non-harmoniques, car il leur applique des configurations de traitements différents. Le modèle détermine que la partie bruit provient d'un spectre non-harmonique lorsque la fréquence fondamentale ω_0 possède une valeur de zéro. Malgré le fait qu'ils utilisent des configurations différentes, ils appliquent toutefois les mêmes principes de compression. Ainsi, le bruit provenant de spectres mixtes et le bruit provenant de spectres non-harmoniques utilisent le plus possible les générateurs de bruit afin de réduire le débit et la complexité du codec.

Le modèle utilise des générateurs de bruit sur des bandes du spectre ayant des caractéristiques plus stochastiques tandis qu'il utilise la quantification vectorielle pour des endroits du spectre possédant des caractéristiques plus tonales. Cette section propose en

premier lieu la description du fonctionnement de la partie bruit des spectres mixtes et la section 4.5 donnera une description du fonctionnement de la partie bruit des spectres non-harmoniques.

La partie bruit mixte représente la deuxième partie d'un spectre mixte et se situe après la fréquence de coupure ω_c. C'est avec la fréquence de coupure ω_c de l'équation 4.16 que le modèle sépare le spectre mixte en une partie harmonique et une partie bruit.

$$\omega_c = (\omega_0 \cdot N_h) + \frac{\omega_0}{2} \tag{4.16}$$

De nombreuses expérimentations ont été effectuées sur la partie bruit du spectre afin d'obtenir la meilleure modélisation.

4.4.1 Techniques expérimentées sur la partie bruit

Cette partie énumère les expérimentations appliquées sur les spectres de bruit mixte afin d'obtenir un signal de synthèse avec une bonne qualité perceptuelle. Les techniques mentionnées dans cette partie utilisent uniquement des générateurs de bruit afin de modéliser la partie bruit qui se situe après la fréquence de coupure ω_c.

La première technique essayée par le modèle consiste à modéliser le spectre de bruit mixte par uniquement des générateurs de bruit. Pour cette première technique, le modèle utilise des générateurs de bruit normalisé ainsi que plusieurs gains d'énergie en sous-bandes afin d'obtenir une enveloppe semblable au spectre original. Les écoutes de signaux démontrent que cette première technique n'offre pas une bonne qualité perceptuelle et au contraire, cette technique ajoute un bruit de fond qui diminue l'intelligibilité des locuteurs.

Cette première expérimentation démontre que la partie bruit mixte ne peut conserver un signal de synthèse avec une bonne qualité perceptuelle en utilisant uniquement des générateurs pour la modélisation de la partie bruit. Une hypothèse retenue du manque de qualité durant la première expérimentation pouvait provenir d'une transition trop abrupte entre les parties harmonique et bruit des spectres mixtes. Pour diminuer ce changement radical entre les deux parties, le modèle propose une seconde expérimentation avec une partie de recouvrement entre la partie harmonique et la partie bruit.

Pour obtenir une partie de recouvrement, le modèle augmente artificiellement le nombre de partiels dans le spectre selon la longueur désirée tout en y appliquant une rampe descendante. De plus, le modèle ajoute également dans cette partie de recouvrement du bruit

créé par un générateur tout en y appliquant une rampe ascendante. Cependant, l'utilisation de cette partie de recouvrement ne donne pas les résultats escomptés et au contraire, elle ajoute un bruit de fond comme pour la première technique, mais crée également des voix plus synthétiques aux locuteurs.

Les techniques expérimentées précédemment ne permettent pas d'obtenir un signal de synthèse de qualité avec uniquement des générateurs pour modéliser la partie bruit. Ainsi, le modèle propose d'introduire une troisième partie aux spectres mixtes qui se nomme partie de transition. Cette partie de transition se positionne après la fréquence de coupure ω_c dans le spectre mixte et se situe entre la partie harmonique et la partie bruit.

4.4.2 Ajout de la partie de transition

Afin de réduire le changement trop radical entre les parties harmonique et bruit, le modèle propose d'utiliser une partie de transition après la fréquence de coupure ω_c. La figure 4.6 montre que cette partie se situe entre les parties harmonique et bruit d'un spectre mixte. De nombreuses écoutes démontrent que l'ajout de cette partie augmente la qualité perceptuelle du signal de synthèse comparativement à une modélisation avec uniquement des générateurs de bruit.

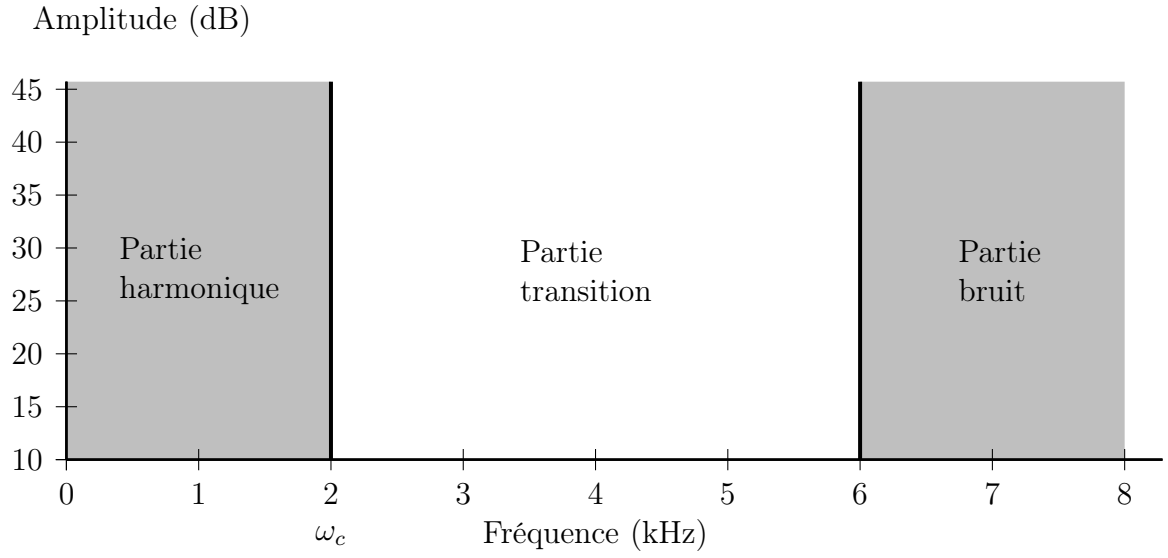


Figure 4.6 Les différentes parties d'un spectre mixte (configuration 1)

La figure 4.6 montre que la bande de transition possède une largeur initiale de 4 kHz qui commence après la fréquence de coupure ω_c. De plus, la largeur de la bande de transition peut augmenter et représenter tout le spectre dans certains cas comme l'indique la figure

4.7. Selon le calcul du coefficient SFM (***S**pectral **F**latness **M**easure*), le modèle augmente la largeur de la partie de transition uniquement si le niveau de tonalité est élevé dans la partie bruit de la figure 4.6 [Gray et Markel, 1974].

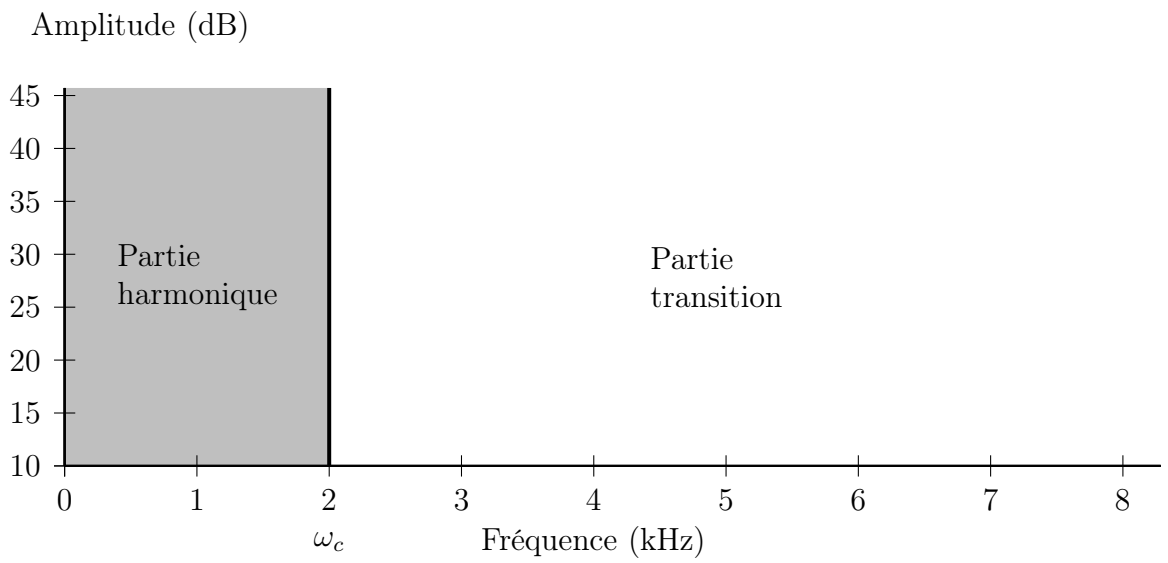


Figure 4.7 Les différentes parties d'un spectre mixte (configuration 2)

Calcul du coefficient de SFM (*Spectral Flatness Measure*)

Pour déterminer le niveau de tonalité d'une bande spectrale, le modèle utilise le coefficient SFM (***S**pectral **F**latness **M**easure*) [Gray et Markel, 1974]. L'équation 4.17 montre que le niveau de tonalité se détermine en comparant les moyennes géométrique et arithmétique d'une partie du spectre d'amplitudes $|\mathsf{X}_\mathsf{b}[k]|$.

$$\text{SFM} = \frac{\sqrt[n]{\prod_{k=0}^{n-1} |\mathsf{X}_\mathsf{b}[k]|}}{\frac{1}{n}\sum_{k=0}^{n-1} |\mathsf{X}_\mathsf{b}[k]|} \qquad k = [0, ..., n[,\ k \in \mathbb{N} \ \text{ et } \ n = \text{nb. de canaux de la FFT} \quad (4.17)$$

Dans l'équation 4.17, la valeur du coefficient SFM varie entre 0 et 1. Lorsque la valeur du SFM s'approche de 1, cela signifie que le spectre possède une distribution d'amplitude à tendance uniforme, tandis qu'une valeur de SFM s'approchant de 0 signifie que le spectre d'amplitudes possède des caractéristiques plus tonales. Le standard MPEG-7 audio utilise

également le coefficient SFM pour déterminer le degré de tonalité de certaines bandes de fréquences [JTC1/SC29/WG11.N6828, 2004].

Avec l'équation 4.17, le modèle calcule le niveau de tonalité afin de déterminer la longueur de la bande de transition. Ainsi, si la partie bruit de la figure 4.6 se situe sous un seuil SFM préétabli, le modèle déclare le segment tonal et la bande de transition devient toute la bande de bruit comme l'indique la figure 4.7. La prochaine partie présente des exemples de bandes de transitions trouvées durant l'analyse de spectres d'amplitudes.

Exemples de bandes de transition dans des spectres mixtes

Cette partie montre des exemples de bandes de transition dans les spectres mixtes trouvés durant l'analyse par le modèle. La figure 4.8 montre un premier exemple d'une partie de transition avec un certain niveau de tonalité, mais sans nécessairement être harmonique.

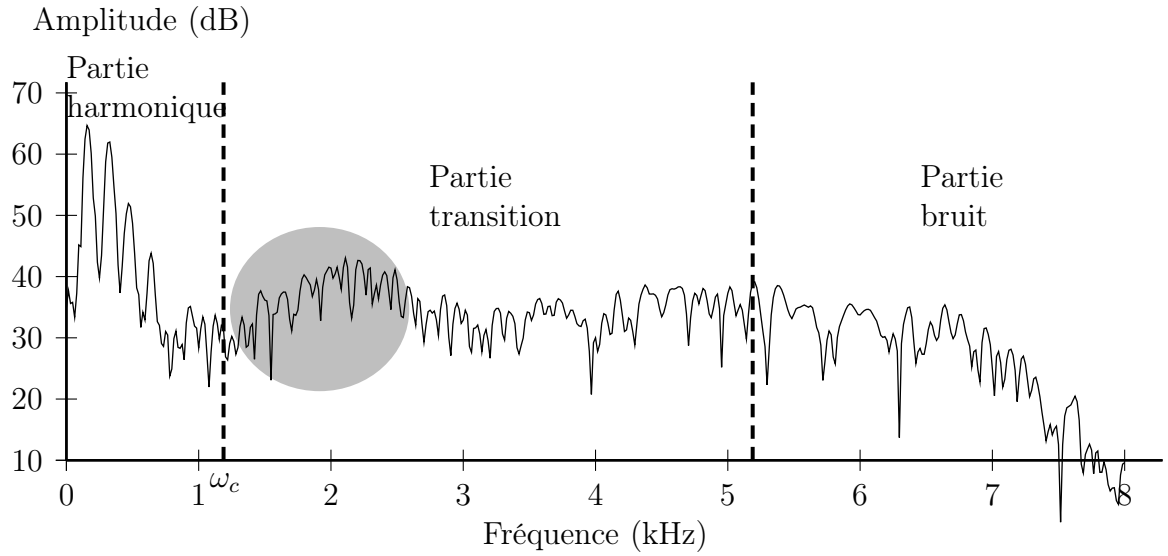


Figure 4.8 Exemple 1 d'un spectre mixte avec trois parties

La figure 4.9 montre un second exemple de partie de transition avec des caractéristiques plus tonales tonales qu'uniformes.

Les figures 4.10 et 4.11 montrent deux exemples lorsque les spectres mixtes possèdent uniquement deux parties : harmonique et de transition. Les spectres des figures 4.10 et 4.11 possèdent des niveaux élevés de tonalité sans être harmoniques. La prochaine section décrit la quantification gain-forme utilisée pour la partie de transition.

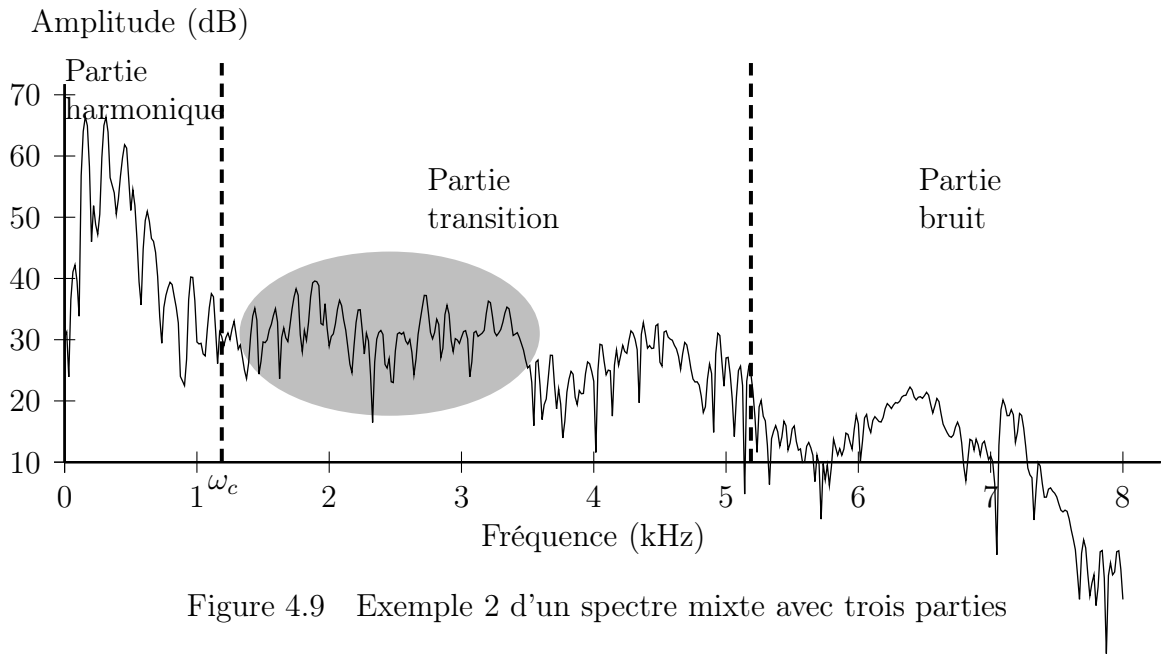


Figure 4.9 Exemple 2 d'un spectre mixte avec trois parties

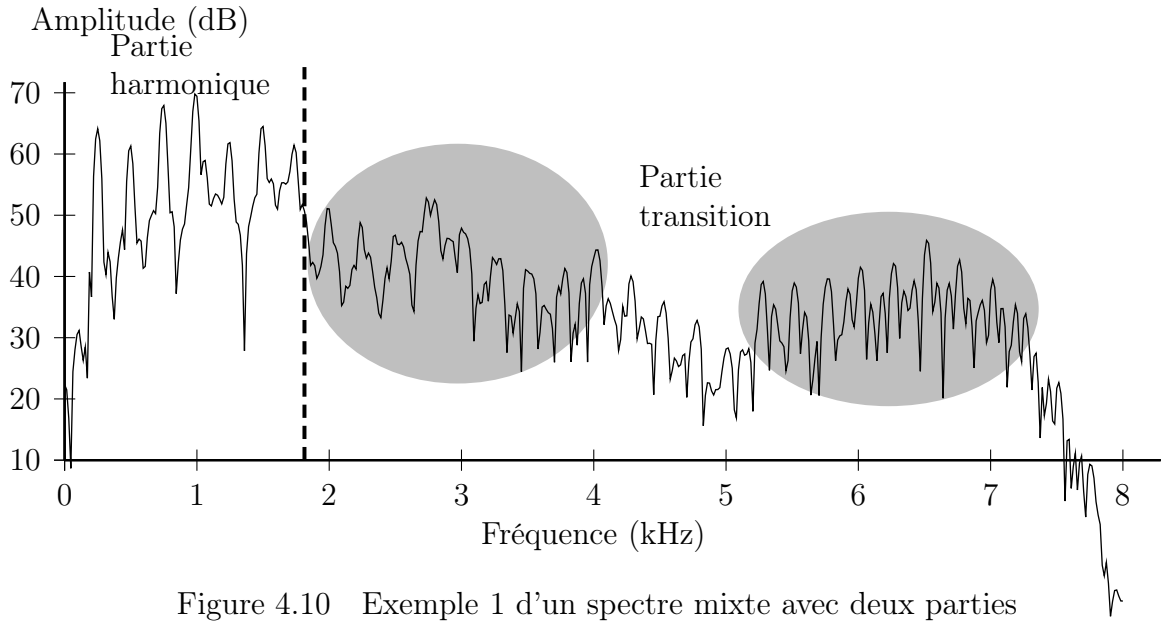


Figure 4.10 Exemple 1 d'un spectre mixte avec deux parties

4.4.3 Quantification vectorielle de la partie de transition

Puisque la partie de transition possède des caractéristiques plus tonales qui la rend difficile à modéliser avec des générateurs de bruit, le modèle utilise une quantification vectorielle gain-forme sur cette partie. Le modèle propose deux différentes configurations de quantification pour la partie transition selon les situations suivantes dans le spectre : trois parties détectées (harmonique, de transition et bruit) dans le spectre (cf. figures 4.6) ou deux parties détectées (harmonique et de transition) dans le spectre (cf. figure 4.7).

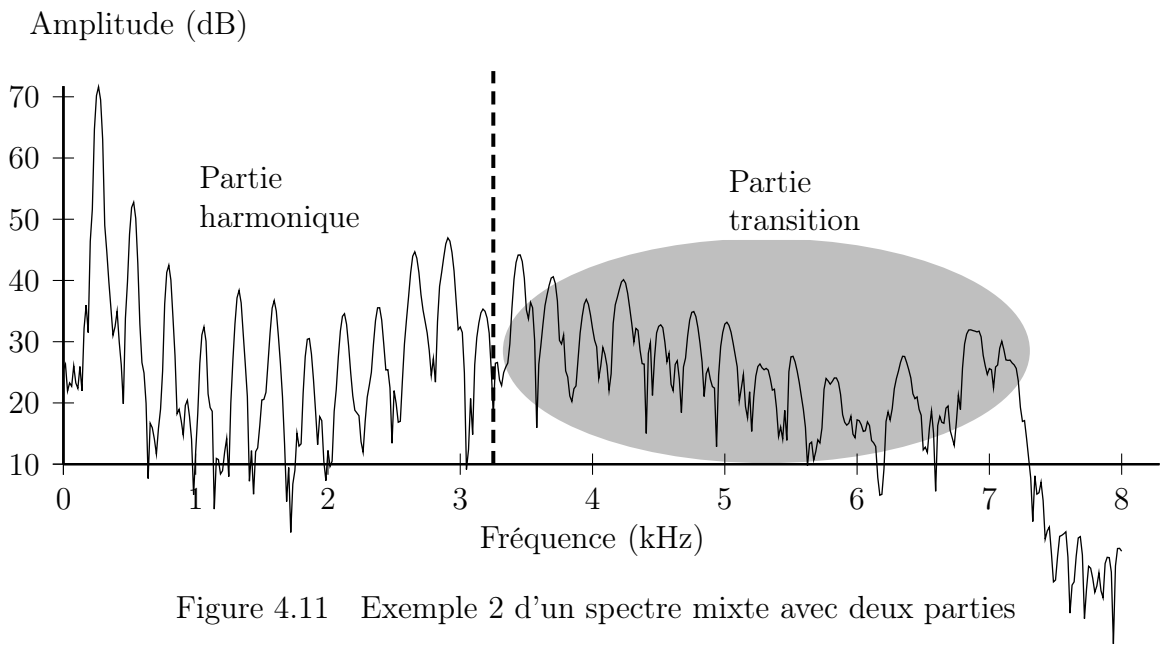


Figure 4.11 Exemple 2 d'un spectre mixte avec deux parties

La figure 4.12 montre la première configuration lorsque les trois parties apparaissent dans un spectre mixte. Le modèle segmente la partie de transition en trois sous-bandes de différentes largeurs qui augmentent graduellement vers les hautes fréquences (cf. figure 4.12). Le modèle offre une plus grande résolution aux premières sous-bandes en diminuant progressivement cette résolution vers les hautes fréquences, ce qui permet d'offrir la meilleure transition possible entre la partie harmonique et la partie bruit.

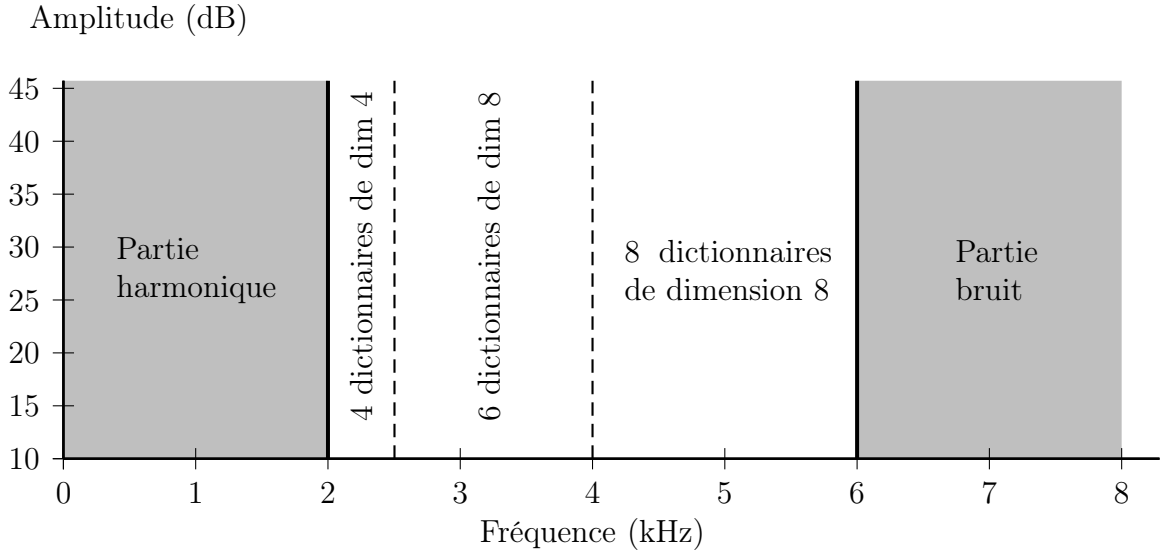


Figure 4.12 Largeur des sous-bandes pour la partie transition (spectre mixte : 3 parties)

La deuxième approche s'utilise lorsque le modèle détecte deux parties (harmonique et de transition) dans les spectres mixtes (cf. figure 4.13). Il faut mentionner que la figure 4.13 représente la longueur maximum que peut posséder la partie de transition, mais qu'elle varie d'une analyse à l'autre.

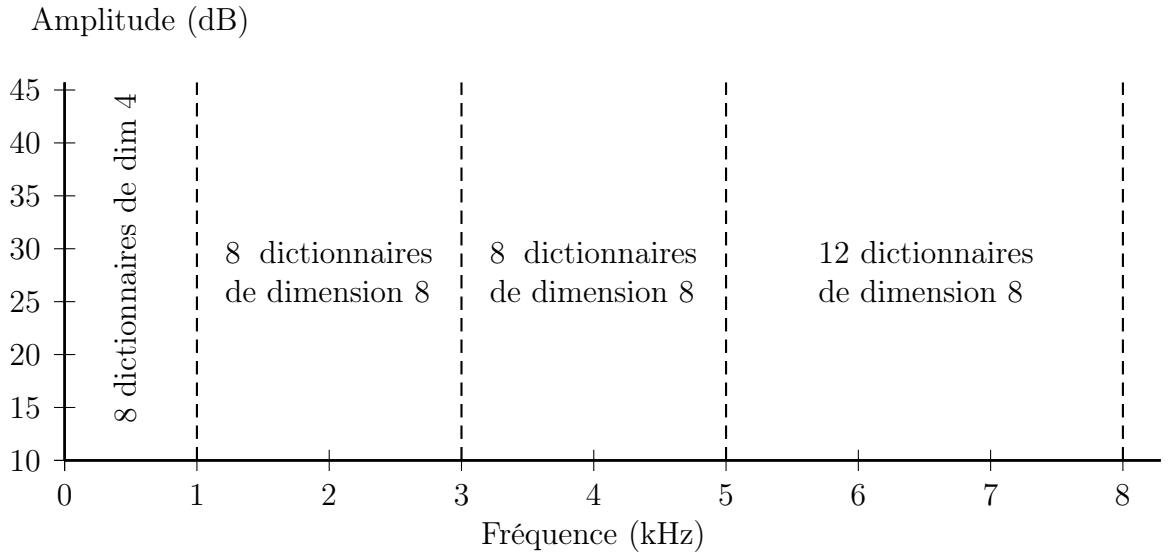


Figure 4.13 Largeur des sous-bandes pour la partie transition (spectre mixte : 2 parties)

Les deux configurations de la partie transition utilisent une quantification vectorielle gain-forme. Les vecteurs de quantification v contiennent les coefficients des canaux de la transformée de Fourier. Le modèle calcule un gain de normalisation $\bar{G}_{\mathrm{norm}_t}$ avec l'équation 4.18 pour chaque vecteur v de quantification de dimension M.

$$\bar{G}_{\mathrm{norm}_t} = \frac{1}{M}\sqrt{\sum_{i=0}^{M-1} v_i^2} \tag{4.18}$$

Le tableau 4.6 indique les dimensions des différents dictionnaires pour les coefficients du spectre de la partie de transition avec la première configuration, lorsque le spectre mixte possède trois parties : harmonique, de transition et bruit.

Le tableau 4.7 montre les dimensions des différents dictionnaires de la partie de transition pour la deuxième configuration, lorsque le spectre contient deux parties : harmonique et de transition. Comme pour la première configuration, le modèle utilise un gain de normalisation $C_{\mathrm{norm}_{t_2}}$ sur les vecteurs v de quantification (cf. équation 4.18).

Tableau 4.6 Description des dictionnaires pour la partie transition (configuration 1)

Dictionnaire des gains de normalisation pour la partie de transition :
$Q : \mathbb{R}^M \rightarrow C_{\text{norm}_{t_1}} \quad C_{\text{norm}_{t_1}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
dimension $M = 3 \quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la première sous-bande (largeur de 500 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{11}} \quad \underline{C}_{t_{11}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 4 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la deuxième sous-bande (largeur de 1500 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{12}} \quad \underline{C}_{t_{12}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la troisième sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{13}} \quad \underline{C}_{t_{13}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Tableau 4.7 Description des dictionnaires pour la partie transition (configuration 2)

Dictionnaire des gains de normalisation pour la partie de transition :
$Q : \mathbb{R}^M \rightarrow C_{\text{norm}_{t_2}} \quad C_{\text{norm}_{t_2}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
dimension $M = 4 \quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la première sous-bande (largeur de 1000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{21}} \quad \underline{C}_{t_{21}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 4 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la deuxième sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{22}} \quad \underline{C}_{t_{22}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la troisième sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{23}} \quad \underline{C}_{t_{23}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la quatrième sous-bande (largeur de 3000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{t_{24}} \quad \underline{C}_{t_{24}} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Cette section a décrit la partie de transition que le modèle propose d'introduire avec les trames mixtes afin de mieux représenter les segments de bruit ayant des caractéristiques

plus tonales. La prochaine section présente la modélisation par générateur de bruit des segments avec des caractéristiques de distribution plus stochastique. La section commence par une étude des différentes distributions de bruit possibles pour le générateur.

4.4.4 Étude de générateurs de bruit avec différentes distributions

Des articles [Richards, 1964][Gazor et Zhang, 2003] démontrent que le signal résiduel x de parole possède les caractéristiques d'une distribution de Laplace. Les études de distributions s'effectuaient avec le signal résiduel entier tandis que dans le cas du projet actuel, l'étude s'effectue uniquement sur des segments résiduels non-harmoniques x_{nv}.

Cette partie décrit les expérimentations effectuées afin de déterminer la distribution la plus adéquate dans le cadre de ce projet parmi les trois distributions les plus plausibles et les plus utilisées : uniforme, normale (gaussienne) et de Laplace. Les prochaines sections décrivent brièvement les distributions et les tests effectués afin de déterminer la meilleure distribution à utiliser dans ce projet.

Forme d'une distribution uniforme

Une distribution uniforme provient d'une variable aléatoire (v.a.) X qui suit une loi de probabilité uniforme et où toutes les valeurs prises par la v.a. sont équiprobables. Ainsi, s'il existe n éventualités d'une v.a. la probabilité d'un évènement est de $1/n$.

L'équation 4.19 [Dodge, 2007] montre qu'une loi uniforme possède toujours la même valeur dans l'intervalle [a ;b] et une valeur nulle ailleurs. La figure 4.14 montre la création de 200 valeurs aléatoires avec un générateur de bruit ayant une distribution uniforme.

$$f(x) = \begin{cases} \dfrac{1}{b-a} & a \leqslant x \leqslant b \\ \\ 0 & \text{ailleurs} \end{cases} \tag{4.19}$$

Forme d'une distribution normale

La loi normale représente le modèle probabiliste le plus utilisé, elle résume de nombreuses distributions statistiques observées. L'équation 4.20 [Dodge, 2007] montre la fonction de distribution d'une variable aléatoire avec la loi normale.

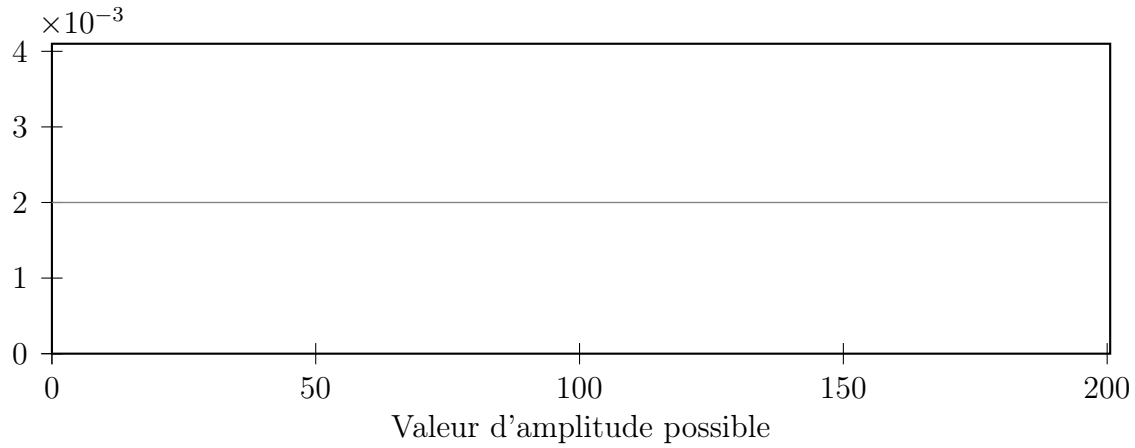


Figure 4.14 Densité de probabilité d'une loi uniforme

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \qquad \begin{aligned} \sigma &> 0 \\ \sigma^2 &= Var(X) \Rightarrow \text{variance} \\ \mu &= E[X] \Rightarrow \text{espérance} \end{aligned} \tag{4.20}$$

La loi normale se caractérise par deux paramètres importants : son espérance μ et son écart-type σ. La figure 4.15 [Dodge, 2007] montre la forme d'une distribution normale appelée centrée réduite, car la valeur de l'espérance est nulle $\mu = 0$ et que la valeur de l'écart-type vaut $\sigma = 1$.

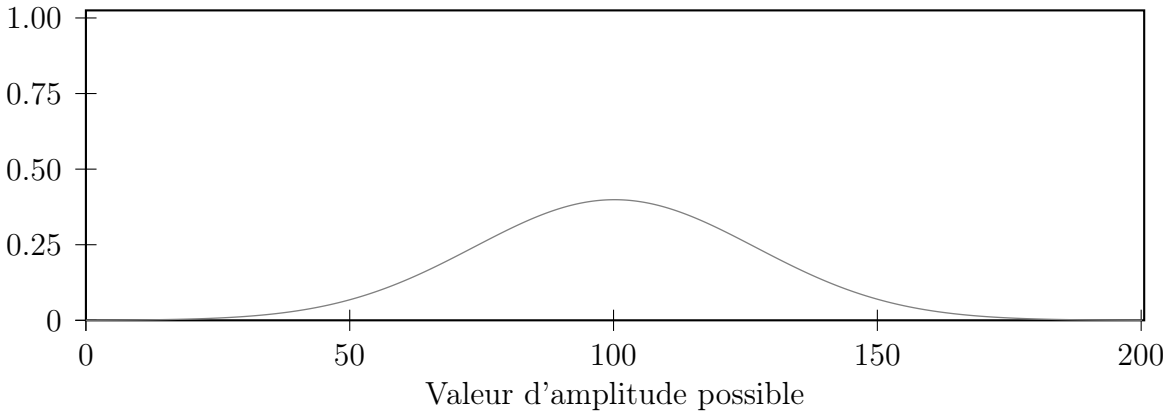


Figure 4.15 Densité de probabilité d'une loi normale

Forme d'une distribution de Laplace

L'équation 4.21 montre la fonction de distribution de Laplace [Dodge, 2007]. Les articles [Richards, 1964][Gazor et Zhang, 2003] mentionnent que cette distribution représente le mieux celle du signal de parole résiduel.

$$f(x) = \frac{1}{2\Phi} \exp\left(\frac{-(x-\mu)}{\Phi}\right) \qquad \begin{aligned} \mu &= E[X] \Rightarrow \text{espérance} \\ 2\Phi^2 &= Var(X) \Rightarrow \text{variance} \end{aligned} \tag{4.21}$$

La figure 4.16 montre la forme d'une distribution de Laplace [Dodge, 2007]. La courbe de distribution de Laplace de la figure 4.16 possède une courbe centrée plus étroite comparativement à la fonction de distribution normale de la figure 4.15

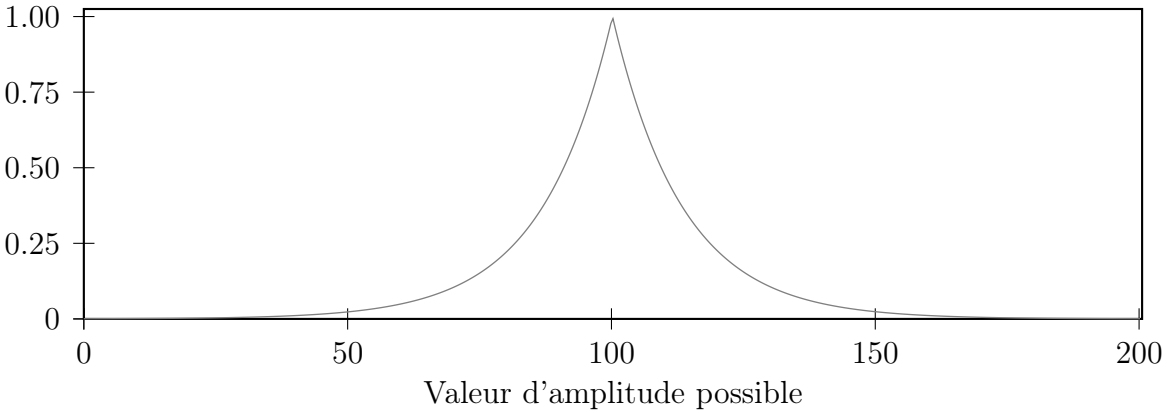


Figure 4.16 Densité de probabilité d'une loi de Laplace

Afin de déterminer la meilleure distribution pour le générateur de bruit du modèle, deux types de tests ont été effectués : un premier test statistique sur la distribution des signaux de bruit résiduel et un second test subjectif sur les différentes distributions.

Statistiques sur la distribution de la partie bruit du modèle

Comme mentionné précédemment, des articles [Richards, 1964][Gazor et Zhang, 2003] démontraient que le signal de parole résiduel possédait les caractéristiques d'une distribution de Laplace. Cependant, dans le cadre de ce projet c'est uniquement la partie bruit x_b du signal de parole résiduel que le modèle tente de modéliser avec du bruit. Ainsi, cette partie propose une étude statistique du bruit résiduel x_b dans le modèle afin de déterminer sa distribution.

Pour connaître la distribution du bruit dans le modèle, une base de données importante a été créée afin d'obtenir uniquement la partie bruit des signaux. Une normalisation a été appliquée sur les amplitudes afin qu'elles se situent entre -1 et 1. Les figures 4.17 et 4.18 montrent la distribution des amplitudes à l'intérieur de 30 classes dont les valeurs d'amplitudes varient entre -1 et 1.

La figure 4.17 présente la distribution des amplitudes pour la partie bruit des spectres mixtes et la figure 4.18 montre la distribution des amplitudes pour la partie bruit des spectres non-harmoniques. Les courbes sur les histogrammes représentent les fonctions de distribution des lois normales théoriques calculées en fonction des données de chaque histogramme.

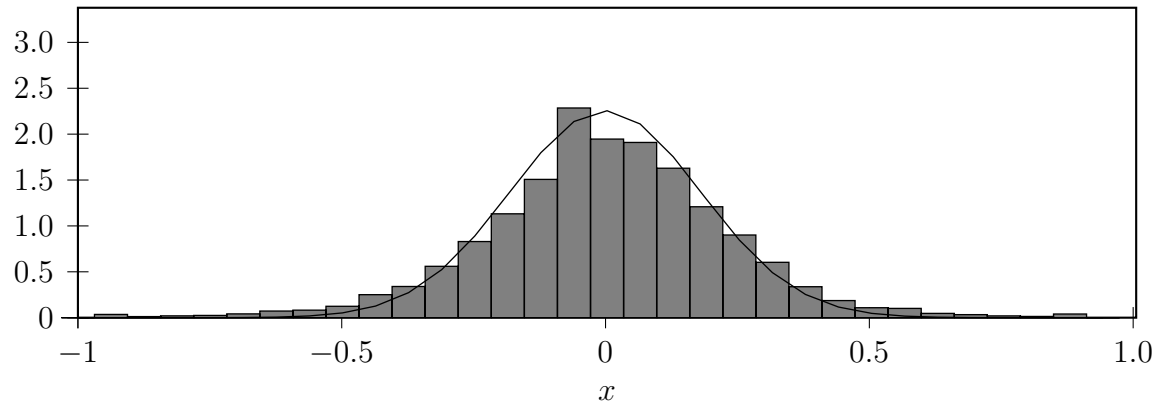


Figure 4.17 Distribution des amplitudes pour la partie bruit des spectres mixtes

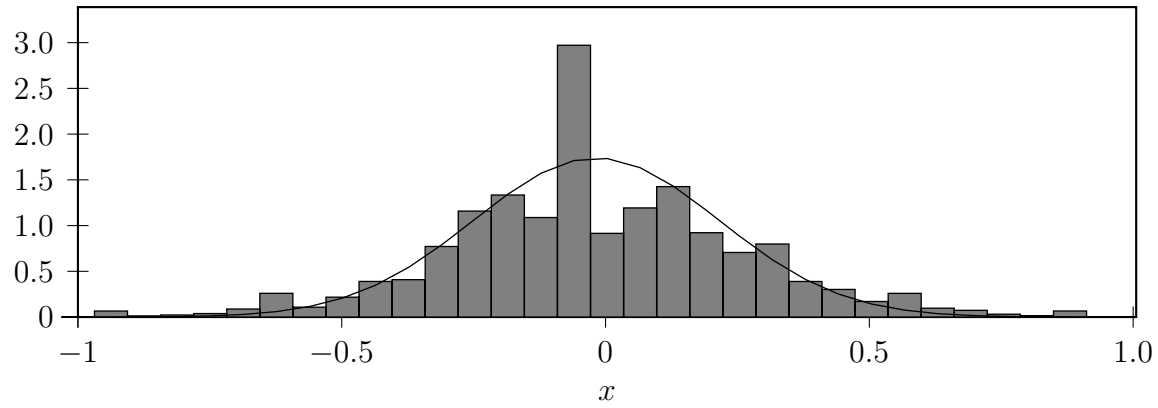


Figure 4.18 Distribution des amplitudes pour la partie bruit des spectres non-harmoniques

L'histogramme de distribution du bruit des spectres mixtes de la figure 4.17 montre une similitude avec la distribution normale théorique calculée. La figure 4.18 possède une classe

plus élevée que les autres, mais ne permet pas d'affirmer que l'histogramme ressemble à une distribution de Laplace.

Cette partie a donné les résultats de tests objectifs sur la distribution de la partie bruit x_b du signal résiduel et qui démontre une certaine similitude avec une distribution normale. La prochaine partie donne les détails des tests subjectifs effectués sur les différentes distributions (uniforme, normale et de Laplace) afin de déterminer s'il existe vraiment des différences lorsqu'elles sont appliquées dans le modèle développé.

Tests subjectifs sur les générateurs de bruit avec différentes distributions

Afin de comparer les différentes distributions, le modèle effectue deux tests subjectifs : un premier test avec des signaux de synthèse qui contient uniquement du bruit et un second test avec le modèle utilisant les différents générateurs.

Le premier test crée des signaux de synthèse contenant uniquement du bruit provenant des différents générateurs. Ce premier test vérifie l'existence de différences lors de l'écoute pour chacune des distributions. Les résultats des écoutes démontrent qu'il existe une différence entre chacun des signaux de synthèse et que chaque distribution possède des signatures auditives bien distinctes.

Le second test subjectif intègre chacune des distributions dans le modèle afin de générer des signaux de parole. Les résultats des tests d'écoute démontrent que l'utilisation de différentes distributions devient indiscernable entre elles lors de l'écoute pour des signaux de parole. Les résultats sont identiques que le modèle utilise n'importe laquelle des distributions de bruit. Pour les signaux de parole, les générateurs ne créent pas de segments de bruit suffisamment long afin que l'oreille puisse distinguer les différentes distributions.

Les résultats des écoutes de fichiers démontrent qu'il est impossible de distinguer les différentes distributions lors de la synthèse de signaux de parole. Ainsi, le modèle pourrait utiliser n'importe laquelle des distributions sans affecter la qualité audio des signaux de synthèse. Cependant, le modèle a choisi d'utiliser un générateur de bruit avec une distribution normale, car les tests statistiques de la figure 4.17 semblaient démontrer que les signaux de bruit possédaient un distribution semblable à la loi normale. La prochaine section donne les détails de la quantification de la partie bruit et en particulier les gains d'énergie utilisés avec les générateurs de bruit.

4.4.5 Quantification vectorielle des gains de la partie bruit

La partie bruit de cette section appartient à un spectre mixte pouvant contenir trois parties distinctes : harmonique, de transition et bruit. Le codec développé modélise la partie bruit des spectres avec un générateur possédant une distribution normale en raison des résultats de la section précédente.

Le générateur de bruit crée un spectre $\hat{\underline{\mathsf{X}}}_{\mathsf{b}}$ avec des valeurs normalisées qui se situent entre 0 et 1. Le modèle applique des gains d'énergie sur le bruit normalisé $\hat{\underline{\mathsf{X}}}_{\mathsf{b}}$ afin de suivre l'enveloppe du spectre original $\underline{\mathsf{X}}_{\mathsf{b}}$. Pour bien suivre l'évolution de l'enveloppe, le modèle calcule deux gains G_{bm1} et G_{bm2} à l'aide de l'équation 4.22. Le modèle sépare la longueur du spectre de bruit en deux longueurs identiques pour le calcul des gains avec l'équation 4.22.

$$G_{bm} = \sqrt{\frac{\underline{\mathsf{X}}_{\mathsf{b}}^{\,2}}{\hat{\underline{\mathsf{X}}}_{\mathsf{b}}^{\,2}}} \tag{4.22}$$

Le tableau 4.8 montre la quantification vectorielle de dimension 2 utilisée pour les gains d'énergie G_{bm1} et G_{bm2}.

Tableau 4.8 Description du dictionnaire de gains du générateur de bruit mixte

Dictionnaire des gains pour le générateur de bruit	
$Q : \mathbb{R}^M \rightarrow C_{\mathrm{G}_{bm}}$	$C_{\mathrm{G}_{bm}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
dimension $M = 2$	$\hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Cette dernière section termine la description des techniques de compression développées pour la partie bruit des spectres mixtes du modèle. La prochaine section propose la conclusion en effectuant un survol des techniques de compression développées pour la partie bruit mixte.

4.4.6 Conclusion sur la partie bruit

Ces dernières sections décrivaient les techniques de compression et de quantification développées pour la partie bruit d'un spectre mixte. Le modèle définit un spectre mixte lorsque celui-ci possède une partie harmonique et une partie bruit. Au début de cette section, le modèle propose l'ajout d'une troisième partie qui se nomme de transition, qui se situe après la fréquence de coupure ω_c, entre les parties harmonique et bruit.

Cette nouvelle partie permet de réduire les transitions trop abruptes entre les parties harmonique et bruit. Le modèle utilise une quantification vectorielle gain-forme sur cette partie du spectre, car elle possède des caractéristiques plus tonales que la partie bruit. Le modèle utilise plusieurs dictionnaires de différentes dimensions afin d'offrir une meilleure résolution au début de cette partie et en diminuant graduellement cette résolution vers la fin de la bande de transition.

Cette section du document propose également la description des techniques de modélisation et de quantification développées pour la partie bruit. Le modèle utilise un générateur de bruit avec une distribution normale pour la modélisation de cette partie. Le modèle a choisi cette distribution d'après les expérimentations effectuées sur différentes distributions : uniforme, normale et de Laplace.

La prochaine section propose également des techniques de compression et de quantification pour la partie bruit, mais spécifiquement pour les spectres non-harmoniques. Les concepts utilisés dans la prochaine section ressemblent à ceux décrits pour les spectres de bruit mixte de cette section, mais avec des configurations différentes.

4.5 Compression des spectres non-harmoniques

Comme pour la section précédente avec la partie bruit mixte (cf. section 4.4.1), le modèle a tenté de modéliser les spectres de bruit non-harmoniques avec uniquement des générateurs, mais les résultats ne furent pas concluants.

Ainsi, le modèle utilise la même technique que le bruit mixte de la section précédente en utilisant de la quantification vectorielle gain-forme sur des bandes du spectre de bruit avec des caractéristiques plus tonales. La prochaine section présente un signal particulier appelé signal transitoire que le modèle catégorise comme un spectre non-harmonique et qui possède des caractéristiques tonales dans les basses fréquences.

4.5.1 Gestions des signaux transitoires dans le modèle

L'une des causes importantes du besoin de la quantification vectorielle pour les spectres non-harmoniques provient de la représentation des signaux transitoires. Les signaux transitoires possèdent des variations d'énergie rapides tant dans le domaine fréquentiel que dans le domaine temporel et c'est ce qui complique leurs détections. La figure 4.19 [Bello *et al.*, 2005] montre les composantes d'un signal transitoire.

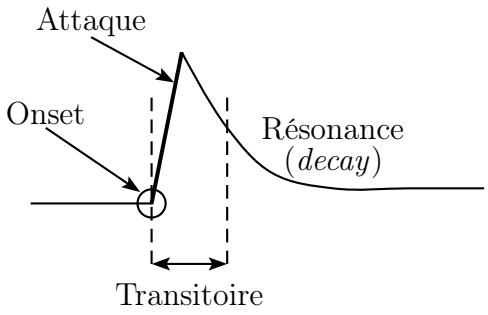


Figure 4.19 Composantes d'un signal transitoire

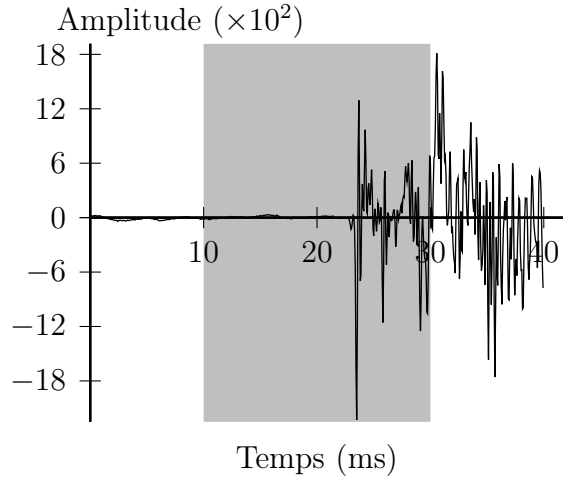
Une mauvaise modélisation des segments transitoires implique souvent une dégradation du signal de synthèse. La figure 4.20 montre un premier exemple d'un signal transitoire qui varie rapidement dans le domaine temporel et fréquentiel.

La figure 4.20(a) montre le signal transitoire original qui possède une variation d'énergie rapide dans le domaine temporel. L'utilisation de la transformée de Fourier avec des signaux qui varient rapidement comme la figure 4.20(a) crée un phénomène d'étalement spectral (cf. figure 4.20(d)). Ce phénomène distribue l'énergie concentrée du domaine temporel sur tous les canaux de la transformée de Fourier. Les modèles par transformée ne peuvent créer un signal temporel qui varie rapidement en amplitude en raison du phénomène d'étalement spectral (cf. figure 4.20(c)).

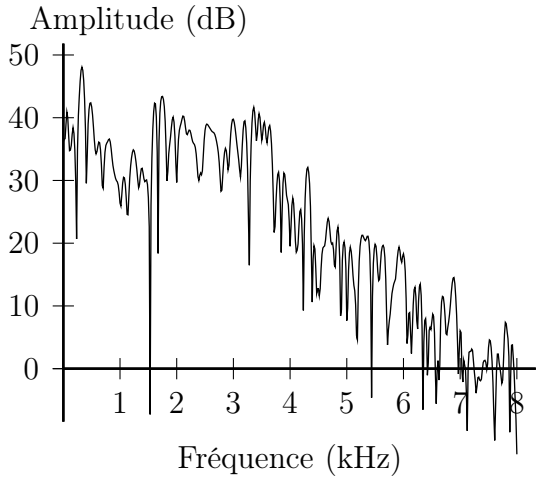
La figure 4.21(b) montre un second exemple de spectre qui contient une énergie concentrée en basse fréquence que le modèle recrée difficilement, avec uniquement un générateur de bruit sur le spectre de synthèse (cf. figure 4.21(d)). L'exemple de la figure 4.21(b) démontre un certain niveau harmonique dans le bas du spectre original, mais pas suffisamment pour les critères du modèle, ainsi il considère le spectre non-harmonique. La figure 4.21(c) montre le signal de synthèse dans le domaine temporel qui possède une certaine difficulté à reproduire la transition entre les parties non-harmonique et harmonique du signal original.

La figure 4.22(b) montre un dernier exemple de signal qui ne représente pas un signal transitoire, mais qui possède une énergie concentrée en basse fréquence sur le spectre d'amplitudes original. L'énergie concentrée de la figure 4.22(b) qui se situe sur un seul partiel ne se catégorise pas comme un signal harmonique durant l'analyse puisqu'il faut un minimum de trois partiels. Cependant, la figure 4.22(a) semble indiquer que le signal original possède une périodicité.

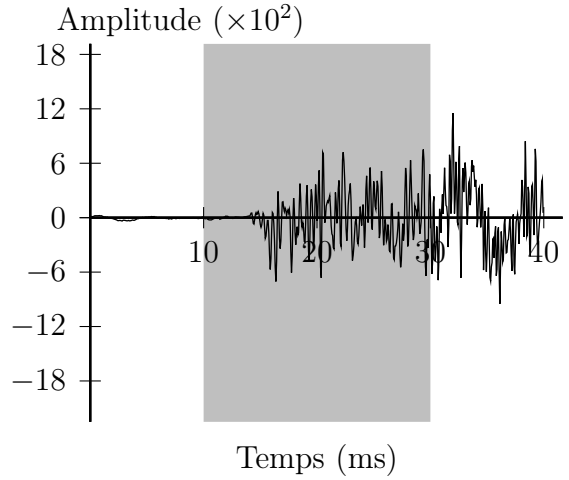
La figure 4.22(d) montre le spectre de synthèse créé avec un générateur de bruit. La figure 4.22(c) montre que le signal de synthèse dans le domaine temporel possède très peu



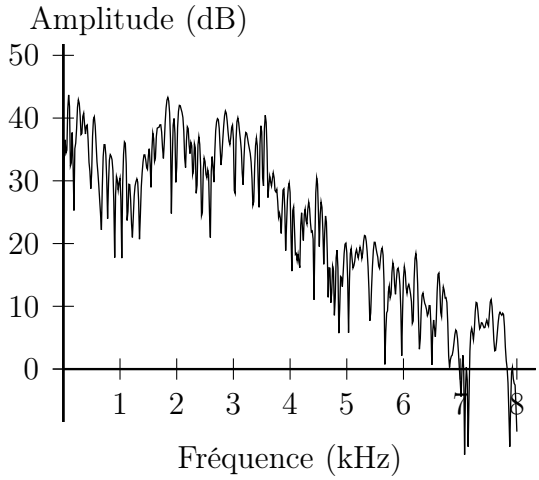
(a) Signal original dans le domaine temporel



(b) Signal original dans le domaine spectral



(c) Signal de synthèse dans le domaine temporel

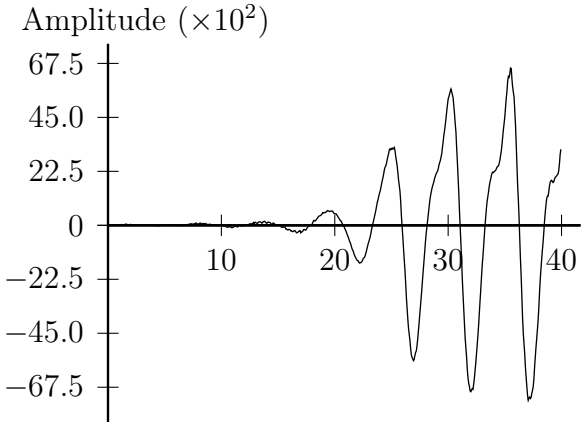


(d) Signal de synthèse dans le domaine spectral

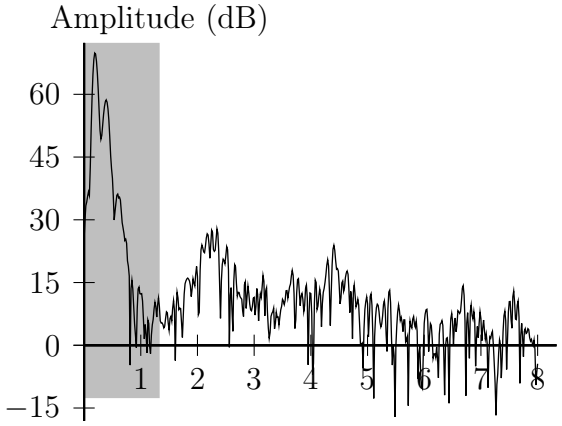
Figure 4.20 Exemple 1 d'un signal transitoire et sa modélisation par le modèle

de caractéristiques harmoniques comparativement au signal original de la figure 4.22(a). Ce dernier exemple montre également la difficulté de modéliser ce type de signal avec uniquement des générateurs de bruit.

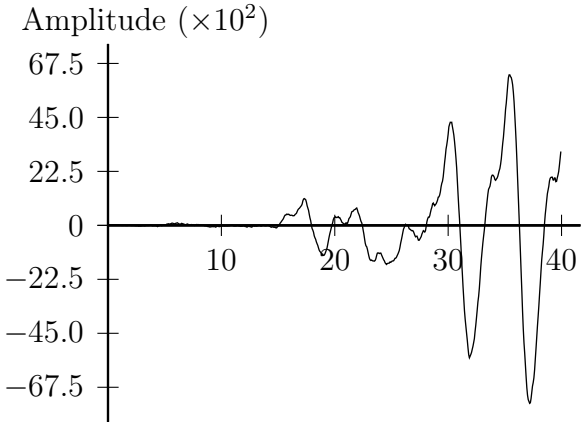
Les exemples de signaux précédents démontrent qu'il existe des spectres avec des niveaux d'énergie importants en basse fréquence. Afin de détecter ces types de spectres, le modèle a expérimenté le calcul du coefficient SFM (***S**pectral **F**latness **M**easure*) sur ces sous-bandes. La prochaine partie explique brièvement cette expérimentation.



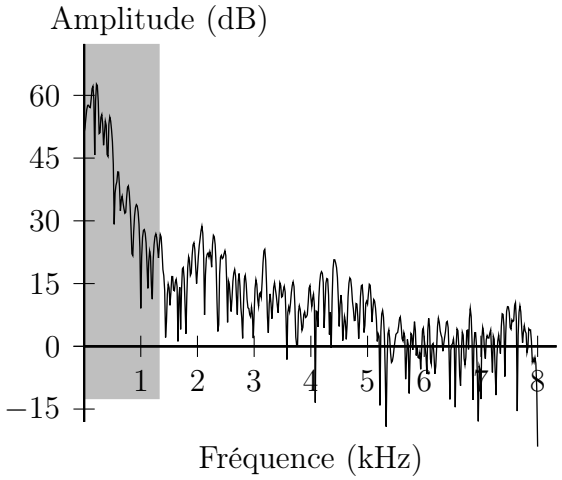
(a) Signal original dans le domaine temporel



(b) Signal original dans le domaine spectral



(c) Signal de synthèse dans le domaine temporel

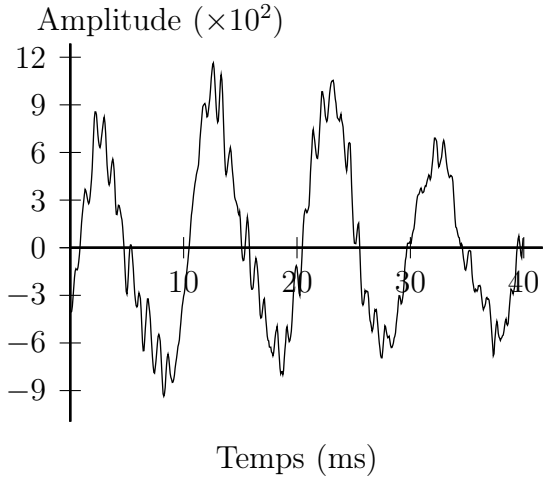


(d) Signal de synthèse dans le domaine spectral

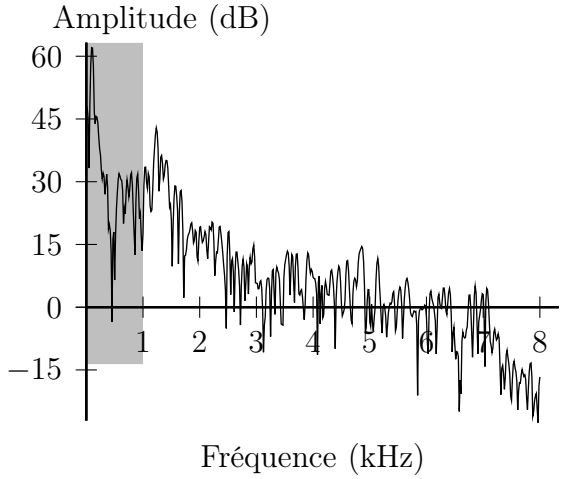
Figure 4.21 Exemple 2 d'un signal transitoire et sa modélisation par le modèle

Détection avec le coefficient SFM (*Spectral Flatness Measure*)

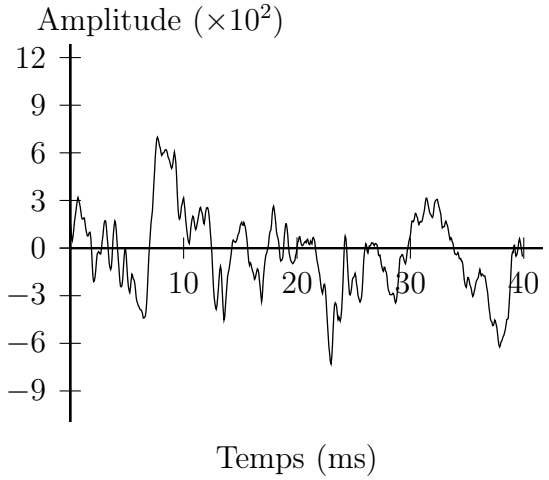
Afin de détecter les signaux avec une énergie importante dans les basses fréquences, le modèle a expérimenté l'utilisation du coefficient SFM (***S**pectral **F**latness **M**easure*) décrit auparavant dans la section 4.4.2. Le modèle applique le calcul du coefficient SFM de l'équation 4.23 au début du spectre d'amplitudes original (de 0 à 2 kHz) afin de détecter les signaux avec des caractéristiques plus tonales.



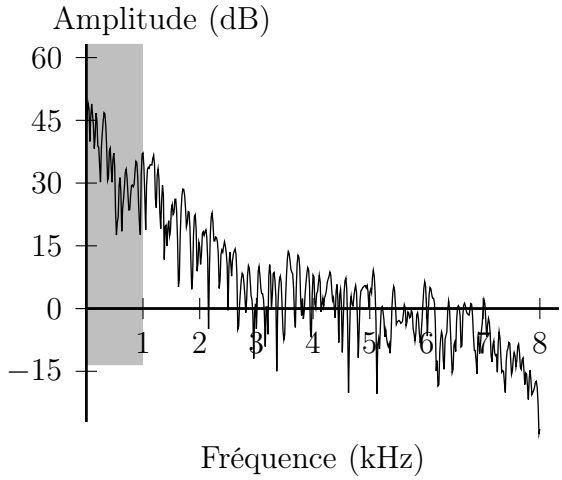
(a) Signal original dans le domaine temporel



(b) Signal original dans le domaine spectral



(c) Signal de synthèse dans le domaine temporel



(d) Signal de synthèse dans le domaine spectral

Figure 4.22 Exemple 3 d'un signal transitoire et sa modélisation par le modèle

$$\text{SFM} = \frac{\sqrt[n]{\prod_{k=0}^{n-1} \mathsf{X}_\mathsf{b}[k]}}{\frac{1}{n}\sum_{k=0}^{n-1} \mathsf{X}_\mathsf{b}[k]} \qquad k = [0, ..., n[,\ k \in \mathbb{C} \quad \text{et} \quad n = 32\ \ (2\ \text{kHz}) \tag{4.23}$$

L'équation 4.23 compare les moyennes géométrique et arithmétique du spectre original X_b dans la largeur de la bande sélectionnée (de 0 Hz à 2 kHz). La valeur du coefficient SFM varie entre 0 et 1. Lorsque la valeur du SFM s'approche de 1, cela signifie que le spectre

possède une distribution d'amplitude à tendance uniforme, tandis qu'une valeur de SFM s'approchant de 0 signifie que le spectre d'amplitudes possède des caractéristiques plus tonales.

Cependant, le modèle n'obtient pas les résultats escomptés avec le coefficient SFM de l'équation 4.23. L'une des raisons provient du fait que le spectre d'amplitudes en basse fréquence ne contient pas nécessairement des caractéristiques tonales, mais plutôt un niveau énergie élevé et condensé comparativement au reste du spectre d'amplitudes.

Ainsi, le modèle propose une solution qui consiste à utiliser une fréquence de coupure minimum ω_{min} afin de séparer le spectre en deux parties : la première partie utilise une quantification vectorielle gains-forme et la seconde partie se modélise avec un générateur de bruit normalisé ainsi que des gains d'énergie. De plus, si la seconde partie possède des caractéristiques plus tonales, détectées à l'aide du calcul du coefficient SFM, le modèle utilisera également la quantification vectorielle gain-forme pour cette seconde partie.

Les nombreuses écoutes de fichiers démontrent qu'une fréquence de coupure minimum de $\omega_{min} = 4$ kHz permet d'obtenir un signal de synthèse perceptuel de bonne qualité. La prochaine section explique la quantification vectorielle que le modèle applique sur les spectres non-harmoniques.

4.5.2 Quantification vectorielle de la partie bruit

Cette section décrit la quantification vectorielle que le modèle applique sur la première partie des spectres non-harmoniques. De plus, si la seconde partie du spectre possède également des caractéristiques plus tonales avec le calcul du coefficient SFM, le modèle appliquera une quantification vectorielle sur tout le spectre.

Le modèle utilise une fréquence de coupure minimum ω_{min} qui sépare en deux parties le spectre non-harmonique. Après de nombreuses écoutes de fichiers audio, la fréquence de coupure minimum a été établie à $\omega_{min} = 4$ kHz afin d'obtenir une bonne qualité perceptuelle.

La figure 4.23 indique la première configuration de quantification possible lorsque le modèle ne quantifie que la première partie du spectre. Pour ce premier schéma de quantification, le modèle utilise quatre dictionnaires de différentes dimensions. Le modèle offre plus de résolution pour les dictionnaires situés en basse fréquence et diminue cette résolution vers les hautes fréquences.

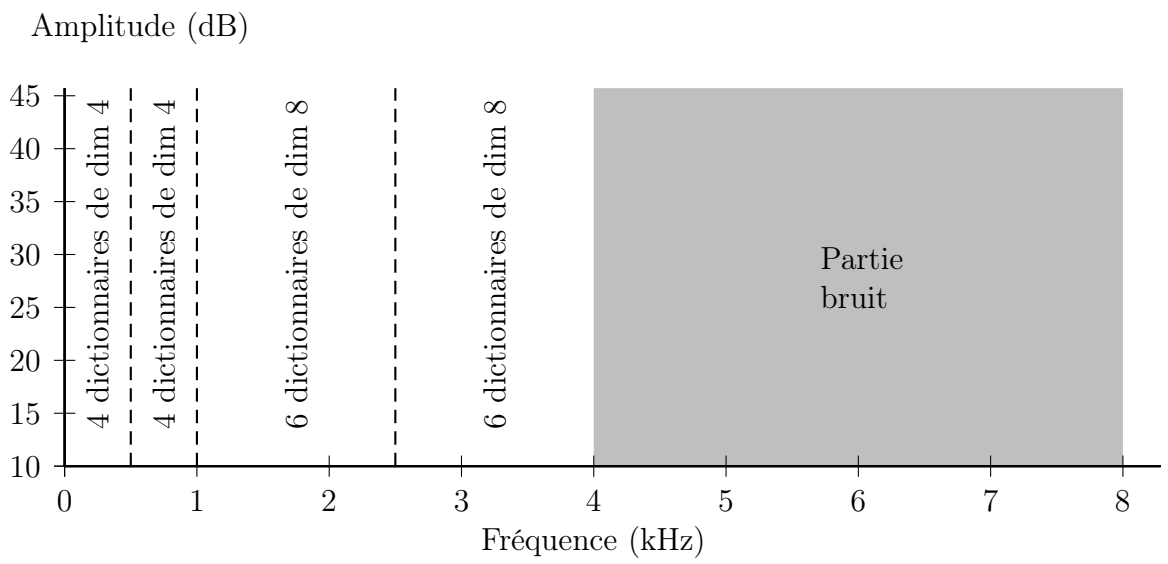


Figure 4.23 Largeur des sous-bandes pour la partie bruit (configuration 1)

Le tableau 4.9 donne les détails des dictionnaires normalisés pour la quantification de la première partie avec la configuration 1. Le modèle utilise l'équation 4.24 afin de calculer les quatre gains de normalisation pour les vecteurs v de quantification de dimension M.

$$G_b = \frac{1}{M}\sqrt{\sum_{i=0}^{M-1} v_i^2} \tag{4.24}$$

La figure 4.24 montre la seconde configuration possible lorsque le modèle quantifie tout le spectre non-harmonique. Cette situation survient lorsque le modèle déclare que la seconde partie du spectre possède également des caractéristiques plus tonales à l'aide du coefficient SFM (***S**pectral **F**latness **M**easure*) de l'équation 4.25.

$$\text{SFM} = \frac{\sqrt[n]{\prod_{k=0}^{n-1} |\mathsf{X}_{\mathsf{pb}}[k]|}}{\frac{1}{n}\sum_{k=0}^{n-1} |\mathsf{X}_{\mathsf{pb}}[k]|} \qquad k = [0, ..., n[,\ k \in \mathbb{N} \text{ et } n = \text{nb. de canaux de la FFT} \tag{4.25}$$

Tableau 4.9 Description des dictionnaires pour la partie bruit (configuration 1)

Dictionnaire des gains de normalisation pour la partie bruit :
$Q : \mathbb{R}^M \rightarrow C_{\text{norm}_{b1}}$ $C_{\text{norm}_{b1}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
dimension $M = 4$ $\hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Dictionnaire pour la première sous-bande (largeur de 500 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b11}$ $\underline{C}_{b11} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 4$ $\underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la deuxième sous-bande (largeur de 500 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b12}$ $\underline{C}_{b12} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 4$ $\underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la troisième sous-bande (largeur de 1500 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b13}$ $\underline{C}_{b13} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8$ $\underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la quatrième sous-bande (largeur de 1500 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b14}$ $\underline{C}_{b14} = (\underline{\hat{v}}_1, \underline{\hat{v}}_2, ..., \underline{\hat{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8$ $\underline{\hat{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Le modèle déclare la seconde partie tonale lorsque le coefficient SFM se situe sous un seuil préétabli. La figure 4.24 montre que la seconde configuration sépare le spectre en quatre sous-bandes de largeurs équivalentes, mais avec des résolutions différentes afin d'offrir plus de résolution vers les basses fréquences tout en diminuant cette résolution graduellement vers les hautes fréquences.

Le tableau 4.10 donne les détails des dictionnaires normalisés pour chacune des sous-bandes de la figure 4.24. Comme pour la première situation, le modèle utilise l'équation 4.25 afin d'obtenir les quatre gains de normalisation.

Cette section a donné les détails de la quantification vectorielle des coefficients de la transformée de Fourier pour les parties des spectres non-harmoniques avec des caractéristiques plus tonales. La prochaine section explique comment le modèle utilise des générateurs de bruit afin de modéliser la seconde partie qui se situe après la fréquence de coupure minimum ω_{min} lorsque survient la configuration 1 (cf. figure 4.23).

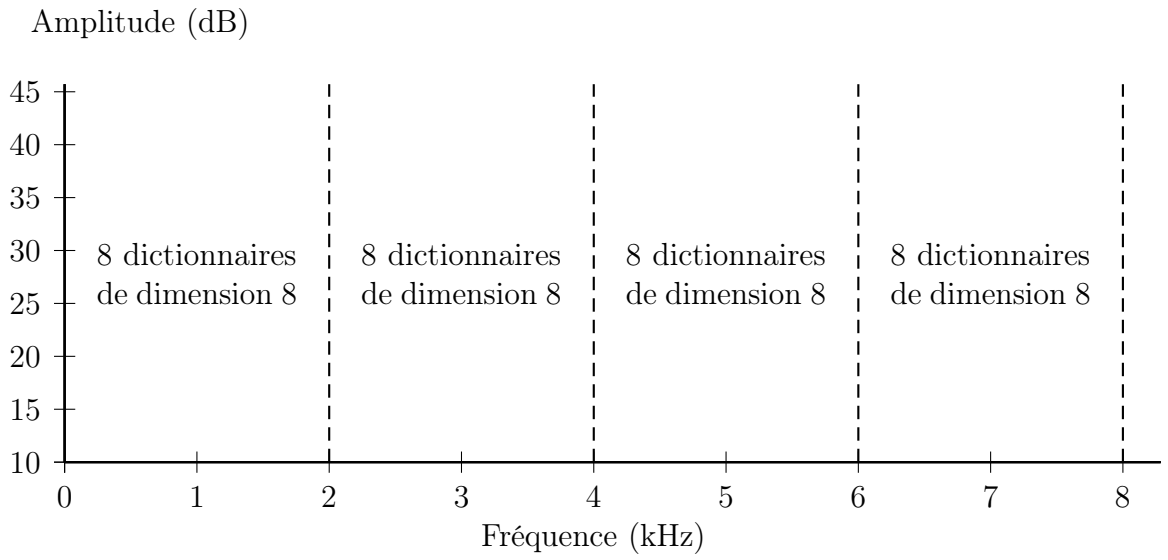


Figure 4.24 Largeur des sous-bandes pour la partie bruit (configuration 2)

Tableau 4.10 Description des dictionnaires pour la partie bruit (configuration 2)

Dictionnaire des gains de normalisation pour la partie bruit :
$Q : \mathbb{R}^M \rightarrow C_{\text{norm}_{b2}} \quad C_{\text{norm}_{b2}} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^{nbits}$
dimension $M = 4 \quad \hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la première sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b21} \quad \underline{C}_{b21} = (\hat{\underline{v}}_1, \hat{\underline{v}}_2, ..., \hat{\underline{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \hat{\underline{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la deuxième sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b22} \quad \underline{C}_{b22} = (\hat{\underline{v}}_1, \hat{\underline{v}}_2, ..., \hat{\underline{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \hat{\underline{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la troisième sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b23} \quad \underline{C}_{b23} = (\hat{\underline{v}}_1, \hat{\underline{v}}_2, ..., \hat{\underline{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \hat{\underline{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$
Dictionnaire pour la quatrième sous-bande (largeur de 2000 Hz) :
$Q : \mathbb{R}^M \rightarrow \underline{C}_{b24} \quad \underline{C}_{b24} = (\hat{\underline{v}}_1, \hat{\underline{v}}_2, ..., \hat{\underline{v}}_L)$ où $L = 2^{nbits}$
dimension $M = 8 \quad \hat{\underline{v}}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

4.5.3 Générateur de bruit

Cette section présente la modélisation par un générateur de bruit de la seconde partie des spectres non-harmonique de la figure 4.23. Comme pour la partie bruit des spectres mixtes de la section 4.4.4, le modèle utilise un générateur de bruit avec une distribution normale afin de créer un spectre de bruit normalisé $\hat{\underline{\mathsf{X}}}_\mathsf{b}$ qui possède des valeurs entre 0 et 1.

Afin de suivre l'enveloppe du spectre original X_b, le modèle calcule deux gains avec l'équation 4.26. Le modèle sépare la partie du spectre en deux sous-bandes d'égales largeurs pour le calcul de l'équation 4.26.

$$G_b = \sqrt{\frac{\mathsf{X}_\mathsf{b}{}^2}{\hat{\underline{\mathsf{X}}}_\mathsf{b}{}^2}} \tag{4.26}$$

Le modèle utilise une quantification vectorielle sur ces deux gains comme le montre le tableau 4.11.

Tableau 4.11 Description du dictionnaire de gains du générateur de bruit

Dictionnaire des gains pour le générateur de bruit	
$Q : \mathbb{R}^M \rightarrow C_{G_b}$	$C_{G_b} = (\hat{v}_1, \hat{v}_2, ..., \hat{v}_L)$ où $L = 2^x$
dimension $M = 2$	$\hat{v}_i \in \mathbb{R}^M$ pour chaque $i \in \mathcal{J} \equiv \{1, 2, ..., L\}$

Cette section complète la description des techniques utilisées pour la modélisation et la quantification des spectres non-harmoniques. La prochaine section effectue un bref retour sur les différentes techniques utilisées sur les spectres non-harmoniques.

4.5.4 Conclusion sur les spectres non-harmoniques

Ces dernières sections ont décrit les techniques de compression et quantification appliquées sur les spectres non-harmoniques. Comme pour les spectres de bruit mixtes de la section 4.4, le modèle utilise la quantification vectorielle gain-forme sur des parties de spectre avec des caractéristiques plus tonales.

Le modèle propose une fréquence de coupure minimum ω_{min} pour séparer le spectre non-harmonique en deux parties : une première partie qui utilise une quantification vectorielle et une seconde partie qui utilise un générateur de bruit ou une quantification vectorielle selon le coefficient SFM. Le modèle vérifie le niveau de tonalité de la seconde partie, avec

le coefficient SFM, afin de déterminer si elle se modélise par un générateur ou bien par une quantification vectorielle.

Le modèle utilise deux configurations de quantification selon qu'elle s'effectue uniquement sur la première partie du spectre (cf. figure 4.23) ou sur le spectre entier (cf. figure 4.24).

4.6 Conclusion du chapitre

Ce chapitre a présenté une version quantifiée du modèle d'analyse-synthèse du chapitre précédent. Le modèle quantifié compresse les signaux de parole et fonctionne entièrement dans le domaine fréquentiel. Les tests subjectifs du chapitre 5 démontrent qu'il obtient une bonne qualité pour des débits de 24 kbit/s et de 30 kbit/s.

L'encodeur catégorise les paramètres à transmettre en trois groupes de paramètres distincts : toujours transmis, pour la synthèse de la partie harmonique et pour la synthèse de la partie bruit. Le décodeur effectue une synthèse distincte avec des modules différents pour les parties harmonique et bruit. Le modèle utilise ce schéma de fonctionnement afin de rendre le codec le plus modulable possible et de simplifier les modifications futures.

Pour la synthèse de la partie harmonique, le modèle propose une méthode afin de diminuer le nombre de phases à transmettre au décodeur, mais sans affecter le nombre de phases dans le spectre de synthèse harmonique. Le modèle transmet uniquement les valeurs des six premières phases et le décodeur crée des valeurs aléatoires ou extrapole les valeurs des autres phases. De plus, avec la quantification vectorielle des phases, le modèle introduit également la prédiction long-terme dans le domaine fréquentiel. Une méthode souvent utilisée pour les modèles de parole temporel, mais pas pour les modèles de codage fréquentiel.

Pour la partie bruit, le modèle utilise des configurations de quantification différentes selon que le bruit provient d'un spectre mixte ou bien d'un spectre non-harmonique. Cependant, le principe de fonctionnement reste le même. Ainsi, le modèle utilise la quantification vectorielle sur les sous-bandes de bruit qui possèdent un certain niveau de tonalité, déterminé par le coefficient SFM (***S****pectral* ***F****latness* ***M****easure*). Pour les autres sous-bandes qui possèdent des caractéristiques d'un signal plus uniforme, le modèle utilise un générateur de bruit avec une distribution normale et des gains d'énergie.

Le modèle quantifié dans ce chapitre fonctionne entièrement dans le domaine de la transformée de Fourier. Il possède un niveau de complexité peu élevé. La complexité provient principalement de la quantification vectorielle dont la dimension maximum pour un dic-

tionnaire est de 16. Le prochain chapitre présente les résultats de tests subjectifs effectués sur différents débits du modèle quantifié de ce chapitre. Les résultats des tests subjectifs démontrent que le modèle quantifié possède une bonne qualité audio autour de 24 kbit/s et de 30 kbit/s.

CHAPITRE 5

ÉVALUATIONS ET ANALYSES DU MODÈLE DÉVELOPPÉ

Ce chapitre présente les détails et les analyses des tests effectués sur le modèle développé du chapitre 3 et de sa version quantifiée du chapitre 4. Ce chapitre évalue quatre éléments importants proposés par le modèle soit : la précision du générateur d'impulsions de sinusoïdes, le modèle d'analyse-synthèse, la méthode de réduction du nombre de phases et le modèle quantifié avec différents débits.

L'évaluation de la précision du générateur de sinusoïdes représente le seul test objectif effectué dans ce chapitre. Les autres évaluations de ce chapitre utilisent des évaluations subjectives qui nécessitent beaucoup plus de ressources (temps, sujets pour l'expérience, etc.) que les tests objectifs. Un test subjectif de type MUSHRA (***MU****ltiple* ***S****timuli with* ***H****idden* ***R****eference and* ***A****nchor*) a été utilisé pour évaluer le modèle d'analyse-synthèse et la méthode de réduction du nombre de phases. Pour la version quantifiée du modèle, c'est un test subjectif de type MOS (***M****ean* ***O****pinion* ***S****core*) qui a été choisi afin de comparer le modèle avec la norme G.722.2 (AMR-WB, ***A****daptive* ***M****ulti* ***R****ate - **W****ide**B**and*) [UIT-T-G.722.2, 2003] de l'institut UIT (**U**nion **I**nternationale des **T**élécommunications).

Une grande particularité du modèle développé provient du fait que le signal de synthèse ne suit pas nécessairement la forme d'onde du signal original, tant dans le domaine temporel que dans le domaine fréquentiel. Ce manque de ressemblances empêche l'utilisation d'évaluations quantitatives sur le modèle comme le calcul RSB (**R**apport **S**ignal sur **B**ruit) et même les algorithmes PESQ (***P****erceptual* ***E****valuation of* ***S****peech* ***Q****uality*) et PEAQ (***P****erceptual* ***E****valuation of* ***A****udio* ***Q****uality*). Ainsi, pour toutes les étapes de développement, la majorité des évaluations ont été effectuées avec des tests subjectifs qui demandent beaucoup plus de ressources que les tests objectifs.

De plus, lors des évaluations subjectives du modèle des difficultés sont également survenues sur le choix des tests à effectuer. Par exemple, l'évaluation du modèle quantifié a nécessité deux itérations complètes de tests différents avant d'obtenir la meilleure comparaison possible entre les différents codecs. Un premier test subjectif MUSHRA avec des sujets experts n'a pas donné une comparaison impartiale des différents codecs.

L'une des raisons importantes provient de la signature auditive différente des signaux de synthèse provenant des codecs temporels et des codecs par transformée. Les sujets experts de ce premier test possèdent une expertise en codage de parole temporel et ont ainsi acquis une familiarité d'écoute pour ces types de codecs. Ainsi, lors du test subjectif MUSHRA, ils reconnaissent immédiatement les différents codecs et cela augmente la difficulté d'une évaluation impartiale.

Afin d'obtenir une meilleure comparaison des différents codecs, un second test subjectif de type MOS a été effectué avec des sujets non-experts qui ne possèdent aucune connaissance dans le domaine du codage. Le déploiement d'un test subjectif MOS nécessite plus de ressources qu'un test subjectif MUSHRA tant au niveau du nombre de sujets, du temps de conception du test et du niveau d'analyse des résultats. Habituellement, des firmes externes effectuent la conception et l'analyse des résultats des tests MOS, toutefois pour ce projet, toutes les étapes du test MOS ont été effectuées afin de bien comprendre tout le processus d'évaluation.

Déroulement du chapitre

La prochaine section montre le peu de ressemblances qui existe entre le signal de synthèse et le signal original, tant dans le domaine temporel que dans le domaine fréquentiel. Cette section explique également l'impossibilité d'utiliser des tests objectifs comme les algorithmes PESQ et PEAQ qui tentent de prédire les scores des tests subjectifs MOS.

Par la suite, ce chapitre présente l'analyse des résultats du test objectif effectué sur le générateur de sinusoïdes. Ensuite, ce chapitre présente les résultats et les analyses du test subjectif MUSHRA qui se décrit en deux parties. La première partie propose une description de l'analyse des résultats du modèle d'analyse-synthèse développé. La seconde partie du test MUSHRA décrit l'analyse des résultats pour la méthode de réduction du nombre de phases. Finalement, ce chapitre donne les résultats et les analyses du test MOS de différentes versions du modèle quantifié en le comparant avec la norme G.722.2 de l'institut UIT.

5.1 Raisons des tests subjectifs sur le modèle développé

Une grande particularité du modèle développé est qu'il ne suit pas nécessairement la forme d'onde du signal original. Cette particularité empêche l'utilisation de tests objectifs

de comparaison avec le signal original tel que le calcul RSB (**R**apport **S**ignal sur **B**ruit) de l'équation 5.1.

$$RSB_{dB} = 10 \cdot log_{10} \left(\frac{A^2_{\text{original}}}{A^2_{\text{bruit}}} \right) \tag{5.1}$$

Puisque le signal de synthèse du modèle possède peu de ressemblances avec le signal original, cela implique une faible valeur RSB dans l'équation 5.1, mais qui ne correspond pas nécessairement à la qualité perceptuelle audio du signal de synthèse.

Afin de bien visualiser le manque de similitudes entre les signaux de synthèse et les signaux originaux, la prochaine section montre des exemples de signaux créés par le modèle d'analyse-synthèse développé.

5.1.1 Peu de ressemblances entre les signaux original et de synthèse

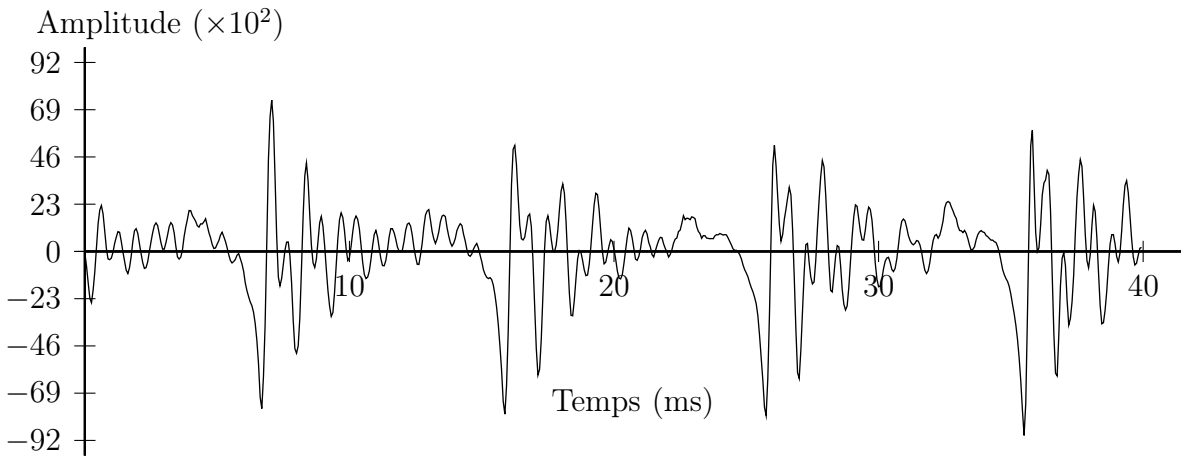
Un premier exemple montre le manque de ressemblances entre le signal original et le signal de synthèse dans le domaine temporel. La figure 5.1(a) représente le signal original et la figure 5.1(b) montre le signal de synthèse qui possède toutes les bonnes valeurs de phases et d'amplitudes trouvées lors de l'analyse.

Le signal original et le signal de synthèse de la figure 5.1 possèdent quelques ressemblances, mais ne suffisent pas afin d'obtenir de bons résultats avec un calcul RSB de l'équation 5.1. La figure 5.2 compare les mêmes signaux de la figure 5.1, mais dans le domaine fréquentiel.

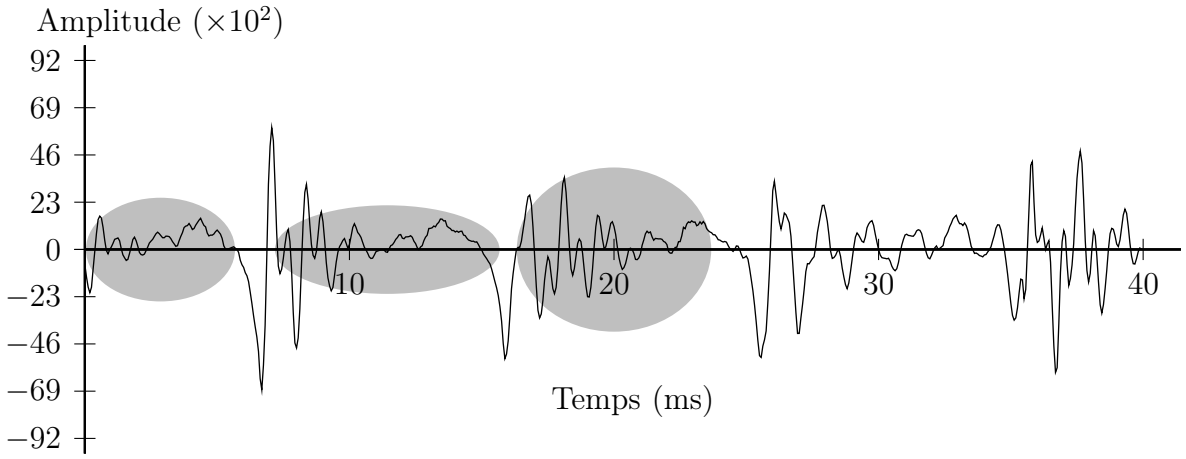
La figure 5.3 montre un second exemple du manque de similitudes dans le domaine fréquentiel entre le signal original et le signal de synthèse. La figure 5.3(a) montre le spectre du signal original et la figure 5.3(b) représente le spectre du signal de synthèse qui possède toutes les valeurs des phases et des amplitudes trouvées lors de l'analyse.

Les deux spectres d'amplitudes des figures 5.3(a) et 5.3(b) possèdent très peu de ressemblances entre eux, mais ce manque de ressemblances n'empêche pas que le signal de synthèse possède une qualité audio perceptuelle transparente.

La figure 5.4 présente les mêmes signaux de la figure 5.3, mais dans le domaine temporel. La figure 5.4 montre que les signaux possèdent un certain niveau de similitudes, mais qui n'existe pas dans le domaine des fréquences comme l'indique la figure 5.3.



(a) Signal original dans le domaine temporel

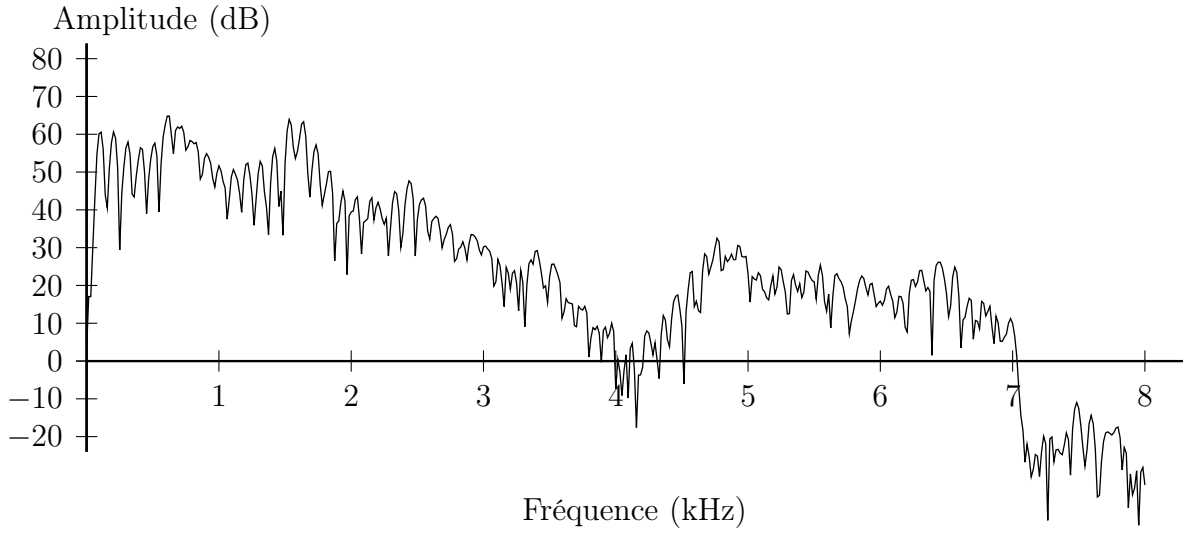


(b) Signal de synthèse dans le domaine temporel

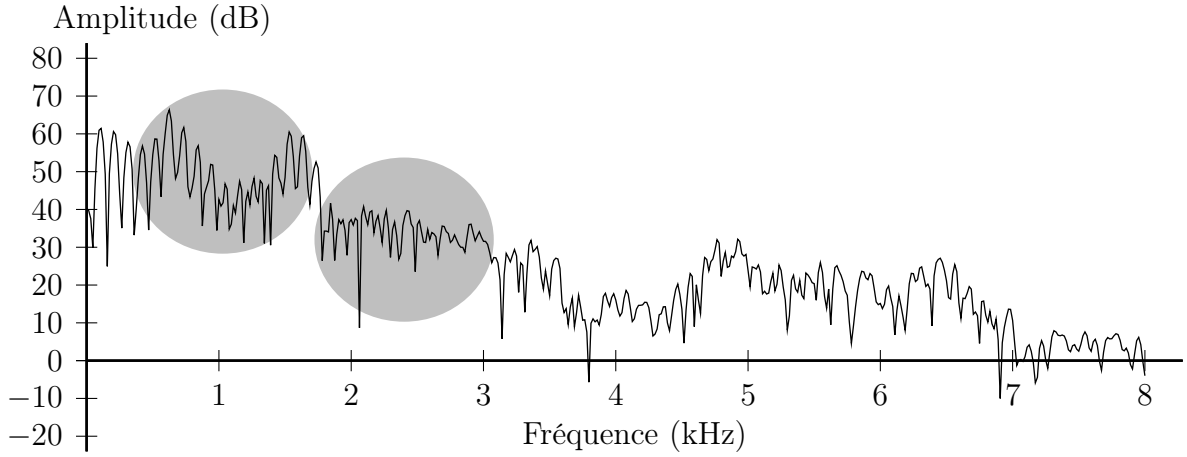
Figure 5.1 Ex. 1 du manque de ressemblances des signaux (domaine temporel)

Cette section 5.1.1 montrait le peu de ressemblances entre les signaux original et de synthèse tant dans le domaine temporel que dans le domaine fréquentiel. Ces manques de similitudes démontrent que des évaluations objectives de comparaison avec le signal original, comme le calcul RSB, ne peuvent s'appliquer afin de démontrer la qualité perceptuelle du signal de synthèse.

La prochaine section explique également pourquoi le modèle n'utilise pas des algorithmes tels que PESQ (***P****erceptual* ***E****valuation of* ***S****peech* ***Q****uality*) et PEAQ (***P****erceptual* ***E****valuation of* ***A****udio* ***Q****uality*) qui tentent de prédire les scores de tests subjectifs MOS (***M****ean* ***O****pinion* ***S****core*). L'institut UIT (**U**nion **I**nternationale des **T**élécommunications) a proposé ces algorithmes afin de réduire les coûts et les délais qu'entraînent le déploiement de tests subjectifs MOS.



(a) Spectre d'amplitudes du signal original

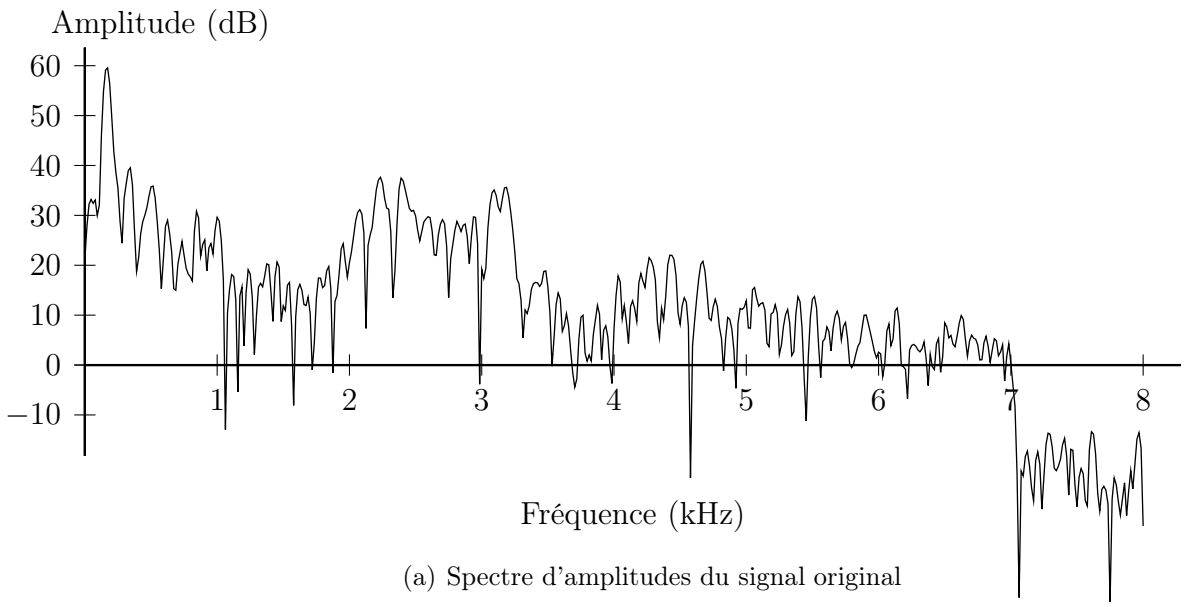


(b) Spectre d'amplitudes du signal de synthèse

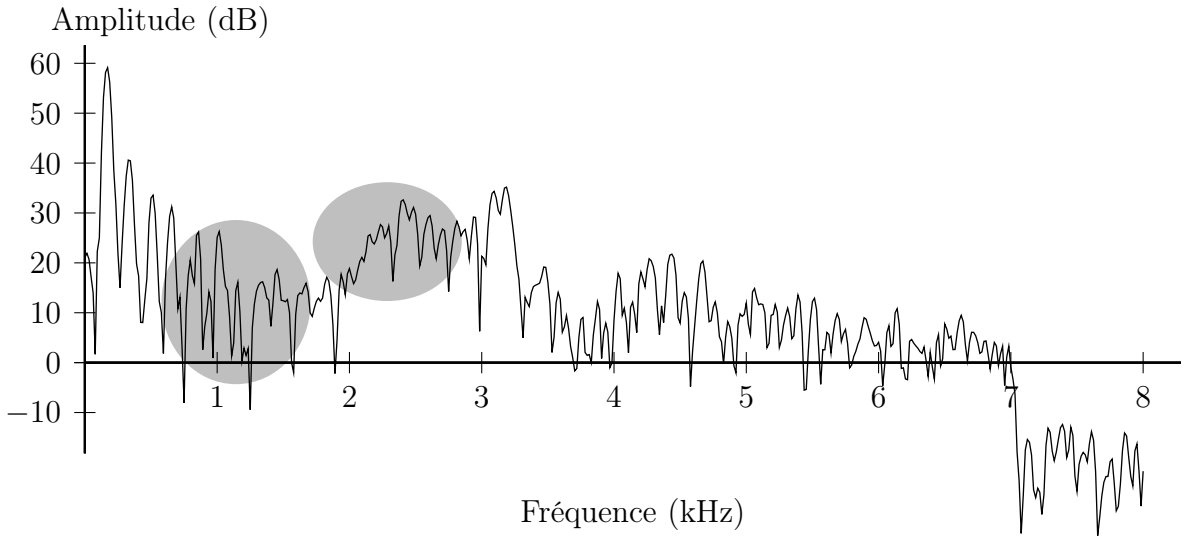
Figure 5.2 Ex. 1 du manque de ressemblances des signaux (domaine fréquentiel)

5.1.2 Impossibilité d'utiliser les algorithmes PESQ et PEAQ

L'évaluation subjective MOS (***M**ean **O**pinion **S**core*) [UIT-T-P.800, 1996] représente le test qui demande le plus de ressources pour tous les types de tests (subjectif et objectif). Afin de réduire toutes les ressources que nécessitent les tests subjectifs MOS, l'institut UIT (**U**nion **I**nternationale des **T**élécommunications) propose au début des années 2000 des algorithmes mathématiques qui simulent rapidement les résultats d'un test MOS. Ces algorithmes ne remplacent pas complètement les tests subjectifs MOS, mais permettent



(a) Spectre d'amplitudes du signal original

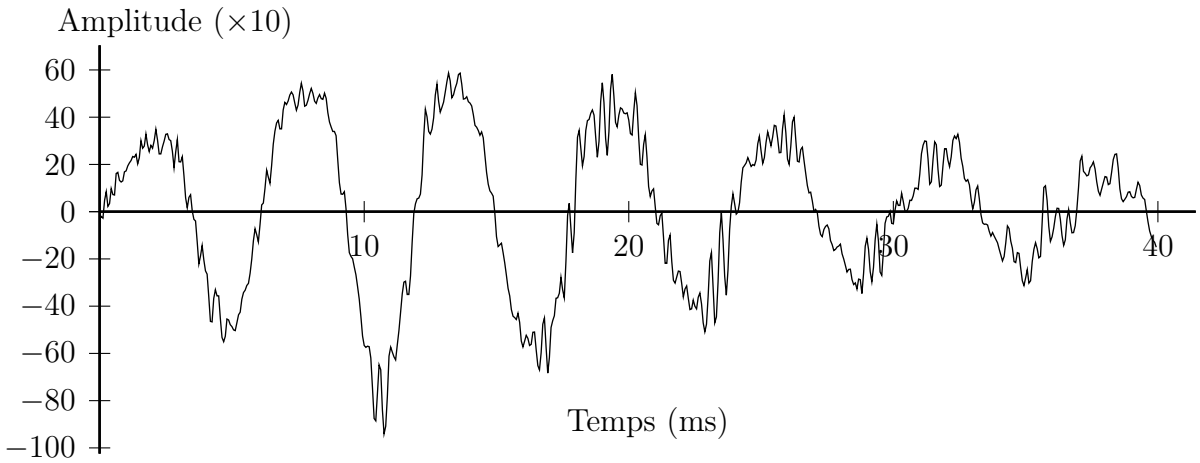


(b) Spectre d'amplitudes du signal de synthèse

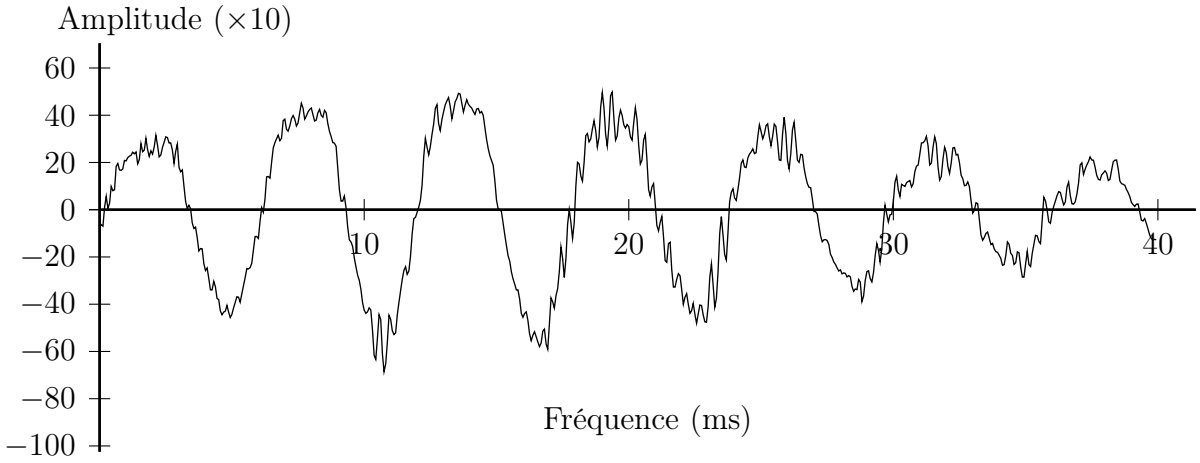
Figure 5.3 Ex. 2 du manque de ressemblances des signaux (domaine fréquentiel)

rapidement d'anticiper certains résultats qui sont par la suite confirmés avec de vrais tests d'écoute MOS.

L'institut UIT propose l'algorithme PESQ (***P**erceptual **E**valuation of **S**peech **Q**uality*) [UIT-T-P.862, 2001][UIT-T-P.862.2, 2005] qui évalue la qualité de signaux de parole et l'algorithme PEAQ (***P**erceptual **E**valuation of **A**udio **Q**uality*) [UIT-R-BS.1387.1, 2001] qui mesure la qualité des signaux audio.



(a) Signal original dans le domaine temporel



(b) Signal de synthèse dans le domaine temporel

Figure 5.4 Ex. 2 du manque de ressemblances des signaux (domaine temporel)

Lors d'un test subjectif MOS les sujets évaluent leur niveau d'appréciation du signal entendu selon l'échelle du tableau 5.1. L'institut possède une norme [UIT-T-P.800, 1996] qui contient tous les détails afin d'effectuer les tests subjectifs de type MOS.

Tableau 5.1 Échelle d'appréciation du test subjectif MOS

Qualité du signal entendu	Note
Excellente	5
Bonne	4
Passable	3
Médiocre	2
Mauvaise	1

Les algorithmes PESQ et PEAQ permettent de diminuer les coûts et les délais qu'entraînent le déploiement de tests subjectifs MOS. L'algorithme PESQ effectue les analyses de signaux de parole dans le domaine temporel alors que l'algorithme PEAQ évalue les signaux audio dans le domaine perceptuel.

Malgré le fait que les deux algorithmes utilisent des analyses différentes et des domaines de traitement différents, il est impossible de les employer avec les signaux de synthèse du modèle développé. La raison provient du fait que ces deux algorithmes commencent par un alignement des signaux à comparer afin de calculer le niveau de corrélation.

L'alignement des signaux pour le calcul du niveau de corrélation empêche l'utilisation de ces algorithmes avec les signaux de synthèse du modèle puisque ceux-ci ne suivent pas nécessairement la forme d'onde du signal original. Afin de confirmer cette hypothèse, les algorithmes PESQ et PEAQ ont été utilisés sur des signaux de synthèse qui possèdent une qualité perceptuelle transparente. Les signaux de synthèse ont obtenu de mauvais scores MOS avec les algorithmes PESQ et PEAQ malgré leur qualité perpétuelle transparente. Ces résultats confirment que le modèle doit absolument effectuer des tests subjectifs traditionnels pour la validation des résultats.

La section 5.3 donne les détails du test subjectif MUSHRA (***MU****ltiple* ***S****timuli with* ***H****idden* ***R****eference and* ***A****nchor*) utilisé pour le modèle d'analyse-synthèse du chapitre 3 et pour la méthode de réduction du nombre de phases du chapitre 4. De plus, la section 5.4 donne les détails d'un test subjectif MOS effectué sur le modèle quantifié du chapitre 4 afin de le comparer avec la norme G.722.2 de l'institut UIT [UIT-T-G.722.2, 2003].

5.2 Évaluations objectives du générateur d'impulsions

Cette section donne les détails du test objectif effectué sur le générateur d'impulsions de sinusoïdes précalculées. Le modèle d'analyse-synthèse fonctionne entièrement dans le domaine fréquentiel et offre un certain degré de liberté entre l'analyse et la synthèse en permettant différentes longueurs pour les transformées de Fourier ($N = 1024$) et de Fourier inverse ($K = 512$).

Ainsi, le spectre d'analyse possède un plus grand nombre de points afin d'obtenir une grande précision pour l'extraction des paramètres importants, tandis que le spectre de synthèse possède un nombre de points plus faibles pour bien suivre l'évolution du pitch. Malgré le fait que le spectre de synthèse possède un nombre de points plus faibles, celui-ci réussit à obtenir une grande précision grâce à un générateur d'impulsions qui possède

une table précalculée. Cette table augmente la résolution du spectre sans accroître la complexité de calcul puisqu'elle est précalculée.

Cette section évalue la précision du générateur d'impulsions du modèle. Le test compare la valeur du pitch trouvée durant l'analyse et la valeur du pitch calculée pour la synthèse du spectre. Pour l'évaluation du modèle, un signal harmonique pur a été créé avec une fréquence fondamentale qui augmente linéairement de 76 Hz à 470 Hz. Les valeurs des fréquences fondamentales posées représentent les limites minimum et maximum pour la voix humaine [Calliope, 1989].

La figure 5.5 montre l'erreur de différence entre la fréquence fondamentale trouvée durant l'analyse et la fréquence fondamentale utilisée pour la synthèse du spectre. Pour obtenir la valeur de la fréquence fondamentale pour la synthèse, le modèle effectue une seconde analyse avec le signal de synthèse.

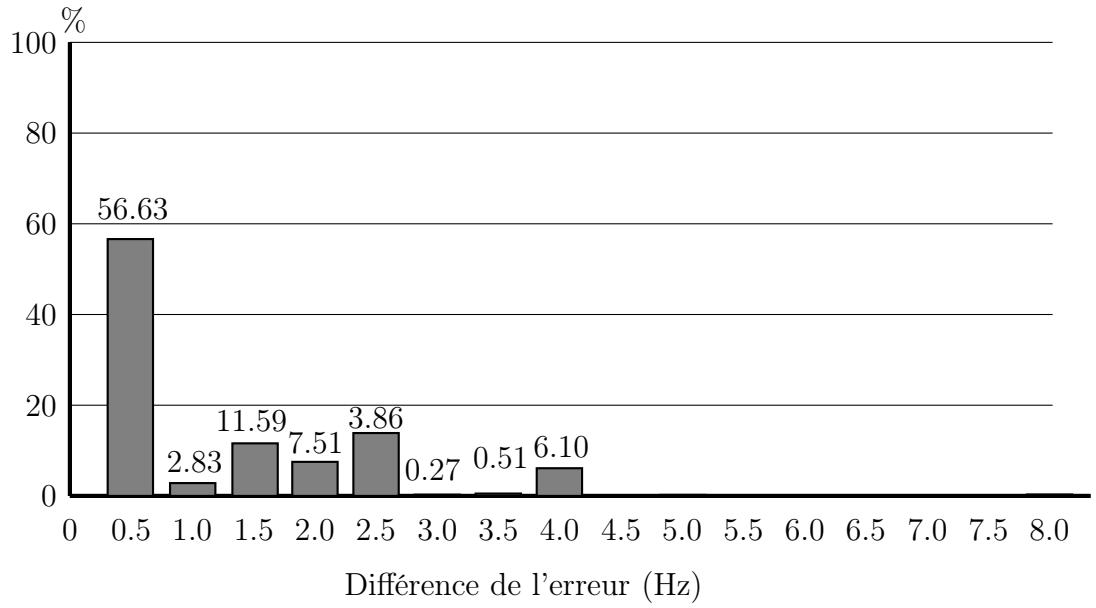


Figure 5.5 Distribution de l'erreur du pitch ω_0 entre l'analyse et la synthèse

La figure 5.5 montre que 56% des erreurs pour la différence entre le pitch durant l'analyse et le pitch du spectre de synthèse possède une différence de 0.5 Hz. La figure 5.6 montre l'erreur cumulative des différences des erreurs de pitch de la figure 5.5.

La figure 5.6 montre que plus de 92% des erreurs sur la précision de la valeur du pitch se situent en bas de 2.5 Hz. Cette erreur est négligeable si l'on considère que le spectre de synthèse possède quatre fois moins de points que le spectre d'analyse. Le générateur harmonique est plus précis que la résolution naturelle du spectre de synthèse sans augmenter la complexité de calcul.

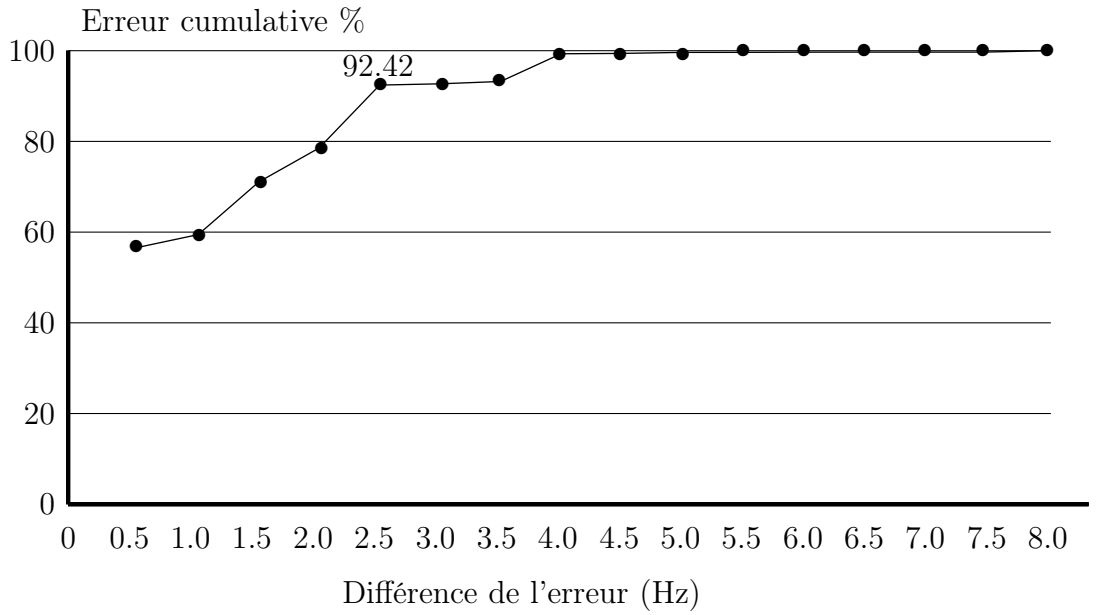


Figure 5.6 Erreur cumulative de la valeur du pitch ω_0 entre l'analyse et la synthèse

Les résultats démontrent qu'il est possible de bien suivre l'évolution du pitch du signal synthèse dans le domaine des fréquences. Avec ces différentes longueurs de transformée pour l'analyse (encodeur) et la synthèse (décodeur), le modèle réussit à bien suivre l'évolution du pitch sans augmenter la complexité du modèle. Le modèle d'analyse-synthèse démontre également qu'il est possible de coder un signal de parole sans la contrainte de la reconstruction parfaite comme les approches fréquentielles existantes. La prochaine section décrit la première partie du test MUSHRA appliqué sur le modèle d'analyse-synthèse afin de vérifier la qualité du signal de sortie.

5.3 Tests MUSHRA et RSB effectués sur le modèle d'analyse-synthèse

Le modèle d'analyse-synthèse utilise le test subjectif de type MUSHRA (***MU**ltiple **S**timuli with **H**idden **R**eference and **A**nchor*) afin d'évaluer les signaux de sortie du modèle. Un test subjectif MUSHRA s'effectue avec des sujets experts, car il nécessite un niveau élevé de concentration sur une longue période de temps. Un test MUSHRA possède plusieurs stimuli à comparer dans une seule séquence du test. Les sujets écoutent autant de fois qu'ils le désirent les stimuli et peuvent également sélectionner des segments précis à écouter sur les stimuli.

Ainsi, une séquence permet de vérifier la qualité des stimuli par rapport à la référence ainsi que la qualité intermédiaire qui représente la comparaison entre les stimuli. Un test MUSHRA offre une grande précision des résultats grâce à son échelle d'évaluation graduée continue de 1 à 100 comme l'indique la figure 5.8. L'institut UIT possède une norme pour le test MUSHRA afin de rendre ce test uniforme [UIT-R-BS.1534.1, 2003].

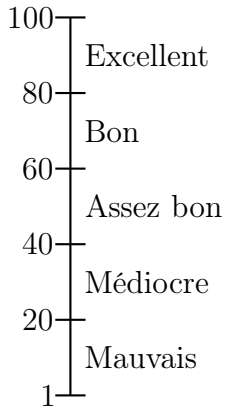


Figure 5.7 Échelle d'évaluation d'un test MUSHRA

La figure 5.8 montre une capture d'écran d'une séquence d'un test MUSHRA effectué durant les évaluations du modèle développé. Dans la séquence de la figure 5.8, les sujets doivent dans un premier temps, trouver le signal de référence dans les stimuli «A» à «G» et la noter à 100. La référence cachée est identique au signal avec le stimulus indiqué «REF». Ensuite, les sujets évaluent les autres stimuli par rapport à la référence et par rapport aux autres stimuli afin d'obtenir des évaluations intermédiaires.

Pour le test MUSHRA présenté dans la prochaine section, il y a eu quatre sujets experts qui ont effectué chacun 20 séquences et ce qui représente un total de 80 scores pour chacun des stimuli. La prochaine partie donne les détails des banques de sons utilisées pour l'évaluation de ce projet.

5.3.1 Description des banques de sons utilisées

Les tests subjectifs de ce chapitre utilisent des fichiers de parole qui proviennent de différentes banques de sons de langues française et anglaise contenant des voix de femmes et d'hommes. Les trois banques de sons choisies pour les évaluations du modèle sont reconnues dans le domaine du traitement du signal de la parole et de l'audio : BDSON [Haton et Lamotte, 1971], *Harvard sentences* [Rothauser *et al.*, 1969] et NTT-AT [NTT, 1994].

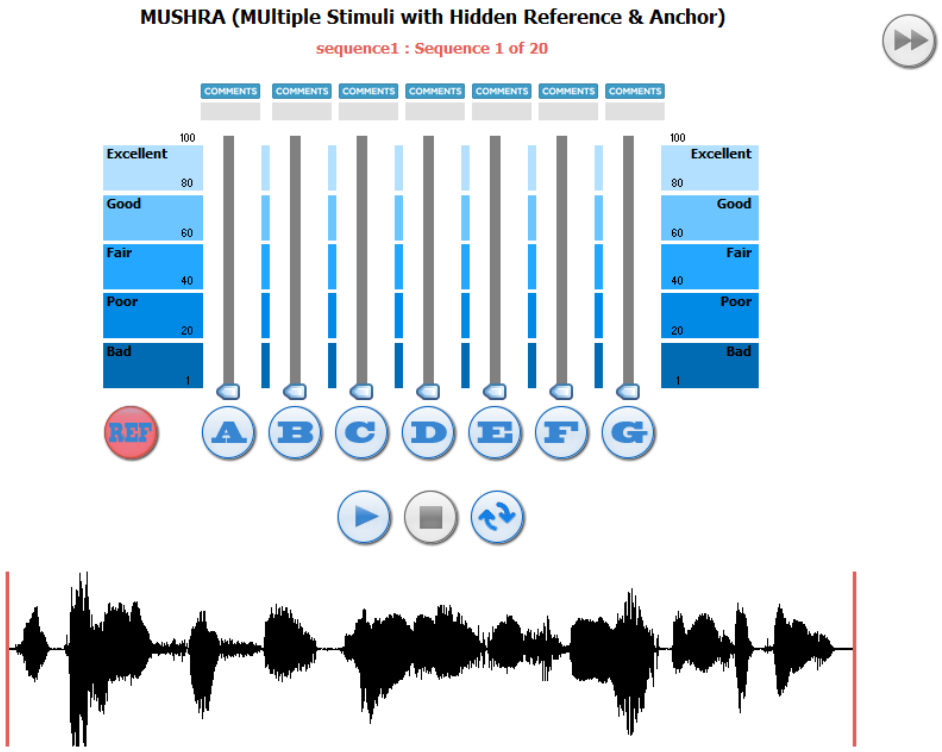


Figure 5.8 Capture d'écran de l'interface d'un test MUSHRA effectué

Le modèle utilise des phrases de la banque de sons BDSON qui contient 25 phrases de voix de femmes et 25 phrases de voix d'hommes en français. La banque de données BDSON possède 20 listes de phrases dont chacune de ces listes possède une phonétique équilibrée, c'est-à-dire qu'elle contient un même nombre de types de phonème et dans la même proportion représentative de la langue utilisée, dans ce cas-ci le français [Haton et Lamotte, 1971].

Le modèle utilise également des phrases qui proviennent de la banque de sons *Harvard sentences*. La banque de sons se compose de 128 phrases en langue anglaise, dont 64 phrases de voix de femmes et 64 phrases de voix d'hommes. Comme avec la première banque de données BDSON, la banque de données *Harvard sentences* se compose de 72 listes et chacune des listes contient 10 phrases [Rothauser *et al.*, 1969]. Ces listes de phrases possèdent également une phonétique équilibrée dans la même proportion représentative de la langue anglaise.

Finalement, le modèle utilise des phrases qui proviennent de la banque de sons NTT-AT (***N****ippon* ***T****elegraph and* ***T****elephone corporation -* ***A****dvanced* ***T****echnology*) [NTT, 1994]. La

banque de sons contient 192 phrases en français avec un nombre égal de phrases dictées par différentes voix d'hommes (96 phrases) et par différentes voix de femmes (96 phrases). Les phrases dictées respectent les normes d'enregistrement de la recommandation P.800 de l'institut UIT [UIT-T-P.800, 1996]. La prochaine section donne les détails du test MUSHRA effectué ainsi que les stimuli utilisés.

5.3.2 Conditions du test

Le test MUSHRA contient 20 séquences à évaluer contenant chacun 6 stimuli. La figure 5.8 montre une capture d'écran du test MUSHRA effectué dans le cadre de ce projet. Comme l'indique la figure 5.8, les sujets évaluent les 6 stimuli sur une échelle continue de 1 à 100.

Pour la création des stimuli, le test utilise des fichiers de parole échantillonnés à 16 kHz avec un format PCM (***P**ulse **C**ode **M**odulation*) encodé sur 16 bits. Les fichiers utilisés pour le test contiennent des signaux de parole dictés par différentes voix de femmes (10 phrases) et de voix d'hommes (10 phrases), en langue française et anglaise.

Le test MUSHRA évalue deux éléments importants du projet : le modèle d'analyse-synthèse du chapitre 3 et la méthode de réduction du nombre de phases à transmettre du chapitre 4.

Le tableau 5.2 décrit les stimuli que contient chaque séquence du test MUSHRA. Selon les critères de la norme MUSHRA [UIT-R-BS.1534.1, 2003], chaque séquence du test doit contenir le signal original caché ainsi qu'un signal original caché filtré passe-bas à 3.5 kHz appelé le signal repère (cf. tableau 5.2).

Tableau 5.2 Descriptions des stimuli du test MUSHRA

Original	Signal original
Repère	Signal original filtré passe-bas à 3.5 kHz
Modèle AS	Signal du modèle d'analyse-synthèse (AS)
Modèle AS-2φ	Signal du modèle AS avec 2 valeurs de phases originales
Modèle AS-0φ	Signal du modèle AS avec 0 phase originale
Modèle AS-Algo	Signal du modèle AS avec 0 phase originale + algo. phases

Le tableau 5.2 contient trois stimuli avec un nombre de phases différentes afin d'évaluer la méthode de réduction du nombre de phases à transmettre au décodeur. Pour ce test, le modèle utilise un minimum de phases afin d'évaluer la qualité du signal de synthèse avec un modèle qui suit très peu la forme d'onde du signal original. De plus, ce test montre également une situation où le modèle ne transmet aucune phase au décodeur avec le stimulus modèle AS-0φ. Les résultats du test MUSHRA se présentent en deux parties :

la première partie présente les résultats pour le modèle d'analyse-synthèse et la seconde partie propose les résultats pour la méthode de réduction de phases.

5.3.3 Résultats des tests du modèle d'analyse-synthèse

Cette section propose une analyse de la première partie du test MUSHRA et RSB avec les signaux de sortie du modèle d'analyse-synthèse. Pour la création du signal de synthèse, le modèle dispose de toutes les valeurs de phases et d'amplitudes trouvées durant l'analyse du spectre. Afin d'évaluer uniquement la performance du modèle d'analyse-synthèse, le signal de sortie possède la modélisation proposée par le modèle pour la partie harmonique et contient la partie bruit du signal originale.

De plus, l'évaluation du modèle propose également une comparaison objective avec le calcul RSB segmentaire afin de comparer ces résultats avec le test subjectif MUSHRA.

Calcul RSB segmentaire

Pour obtenir les valeurs RSB segmentaires, le modèle calcule la valeur RSB de chaque trame en comparant le signal original s et le signal de synthèse $\hat{s}$ dans l'équation 5.2.

$$RSB = 10 \cdot \log_{10} \frac{\sum_{n=0}^{K-1} s(n)^2}{\sum_{n=0}^{K-1} (s(n) - \hat{s}(n))^2} \qquad K = \text{Longueur d'une trame} \tag{5.2}$$

Le calcul du RSB segmentaire s'obtient en effectuant la somme de toutes les valeurs RSB par trame calculées de l'équation 5.2 et en divisant cette somme par le nombre de trames analysées (cf. équation 5.3).

$$RSB_{seg} = \frac{\sum_{j=0}^{N-1} RSB_j}{N} \qquad N = \text{Nb. de trames} \tag{5.3}$$

La figure 5.9 montre des valeurs normalisées en pourcentage pour les valeurs RSB segmentaires. La normalisation s'effectue en fonction du signal original et qui lui possède la valeur de 100%. Une valeur RSB élevée indique que le signal de synthèse est peu bruité tandis

qu'une valeur RSB faible signifie que le signal de synthèse possède un niveau de bruit plus élevé que le signal original. Les résultats du test MUSHRA de la figure 5.9 représentent le score moyen obtenu de chaque stimulus sur un total de 80 scores.

Analyse des résultats

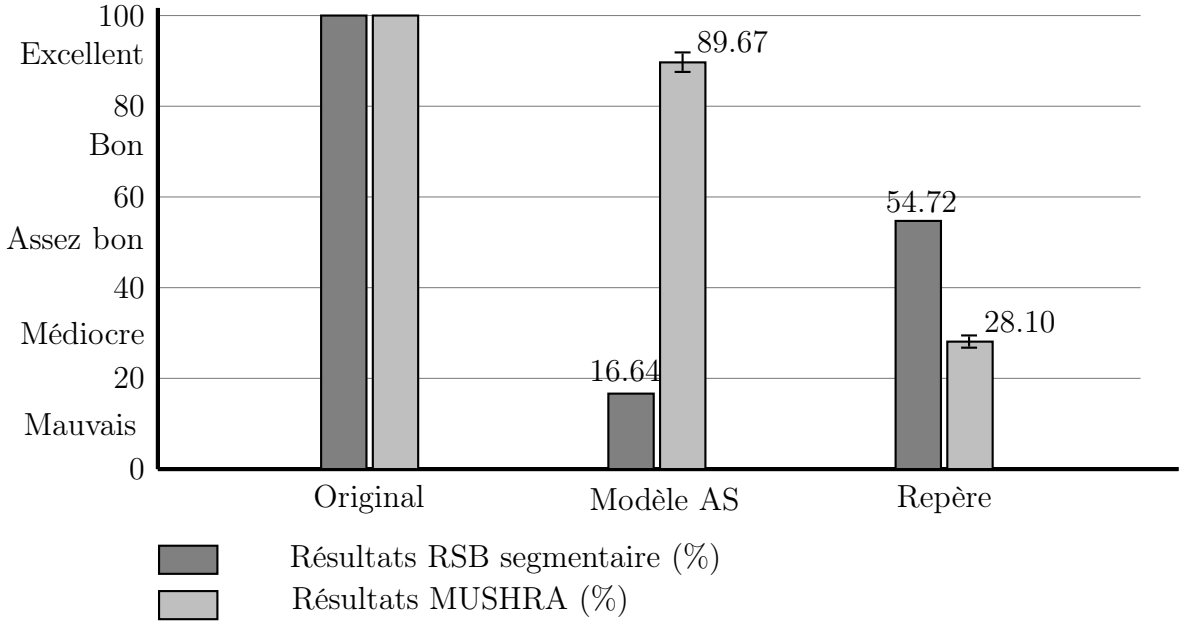


Figure 5.9 Résultats des tests MUSHRA et RSB pour le modèle développé

La figure 5.9 montre que le modèle d'analyse-synthèse proposé possède une valeur RSB segmentaire de 16.64% par rapport au signal original, ce qui signifie que le signal de synthèse possède une mauvaise qualité. Cependant, le résultat subjectif MUSHRA montre un résultat contraire au test objectif RSB. Ainsi, le signal de synthèse du modèle possède une qualité dans la catégorie excellent avec un résultat d'environ 90%. Les sujets de l'expérience ont également mentionné qu'il était souvent nécessaire d'effectuer plusieurs écoutes des stimuli afin de distinguer le signal original et le signal du modèle d'analyse-synthèse durant le test.

Les résultats de la figure 5.9 démontrent qu'il est possible d'obtenir un signal de synthèse de qualité dans la catégorie excellent avec un signal qui ne suit pas nécessairement la forme d'onde du signal original. De plus, les résultats RSB segmentaires de la figure 5.9 confirment que l'évaluation du modèle ne peut s'effectuer avec un test objectif. La prochaine section décrit la deuxième partie du test MUSHRA avec l'étude des phases.

5.3.4 Résultats des tests sur les phases du modèle d'analyse-synthèse

Cette section donne les détails de la deuxième partie du test MUSHRA qui évalue la méthode de réduction des phases du modèle d'analyse-synthèse. La méthode pour la réduction du nombre de phases transmis n'influence pas sur le nombre de phases dans le spectre de synthèse. Pour les phases non transmises, le modèle applique une extrapolation des valeurs des phases par rotation pour les anciens partiels ou leur attribue des valeurs aléatoires pour les phases provenant de nouveaux partiels. La section 4.3.1 du chapitre 4 donne tous les détails pour la méthode de réduction de phases transmises.

Afin d'approfondir les connaissances des phases dans le modèle, des études ont été effectuées sur l'importance des phases sur la qualité du signal de synthèse. La prochaine section donne les détails des observations sur le comportement des phases dans le modèle.

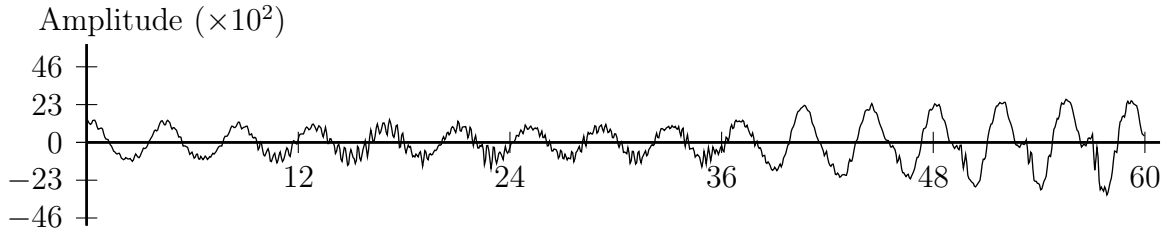
Analyse sur la cohérence des phases entre les trames

L'objectif de l'étude des phases consiste à déterminer leur importance sur la qualité du signal de synthèse dans le modèle. Pour déterminer ce niveau d'importance, le modèle diminue au maximum le nombre de phases transmises au décodeur tout en tentant de conserver un signal de synthèse de qualité.

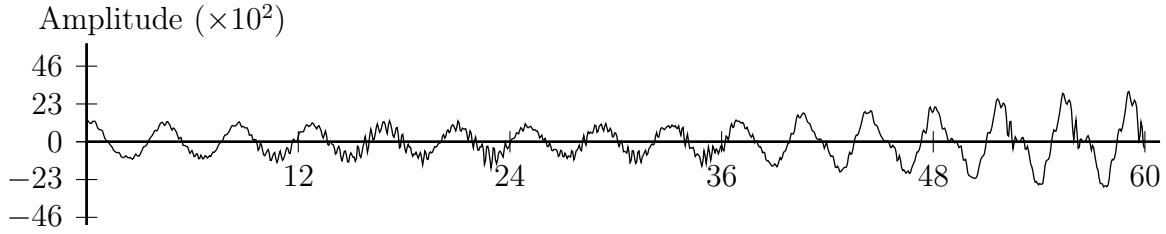
Après quelques tests non formels, les résultats démontrent que le modèle nécessite peu de phases originales pour obtenir un signal de synthèse de qualité. Pour démontrer ce concept, le modèle utilise un nombre restreint de phases afin de mieux comprendre les limites du modèle. Ainsi, le tableau 5.2 des stimuli montre que le modèle utilise un stimulus avec 2 phases originales et deux stimuli avec 0 phase originale afin de créer les spectres de synthèse.

Avec le stimulus modèle AS-0φ, le modèle tente l'expérience plus loin en ne transmettant aucune valeur de phase au décodeur. Pour la création du spectre de synthèse, le décodeur crée des valeurs de phases aléatoires pour les nouveaux partiels et applique une extrapolation par rotation des phases pour les anciens partiels.

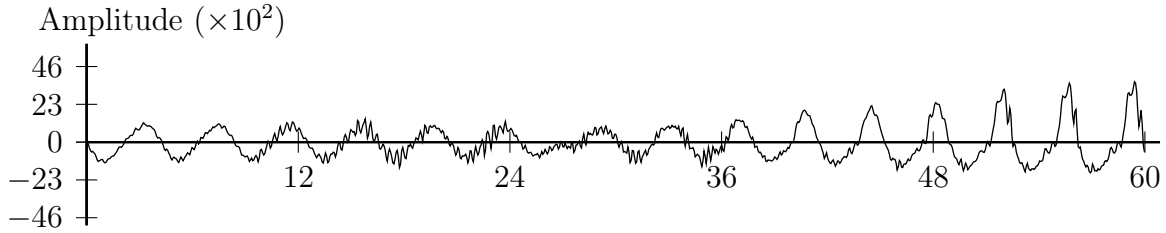
Cependant, les nombreuses écoutes de fichiers démontrent la présence d'artéfacts audibles à des endroits précis dans le stimulus modèle AS-0φ et qui n'existent pas dans le signal du stimulus modèle AS-2φ avec 2 phases originales. La figure 5.10 montre l'exemple d'un segment de synthèse avec un artéfact audible qui s'entend uniquement dans le stimulus modèle AS-0φ avec aucune phase.



(a) Signal original dans le domaine temporel



(b) Signal du stimulus modèle AS-2φ avec 2 phases originales



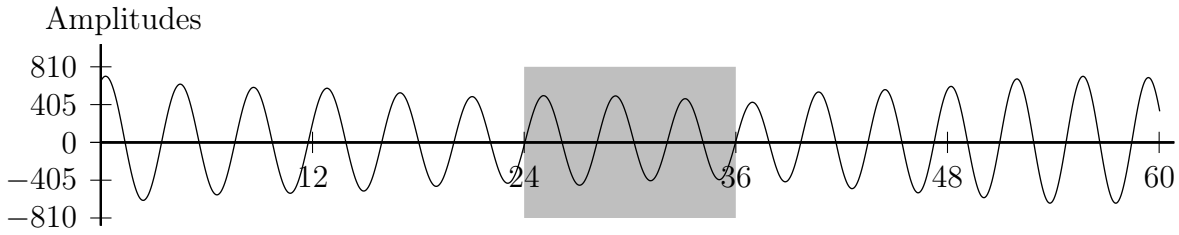
(c) Signal du stimulus modèle AS-0φ avec aucune phase originale

Figure 5.10 Signaux de synthèse avec un nombre différent de phases originales

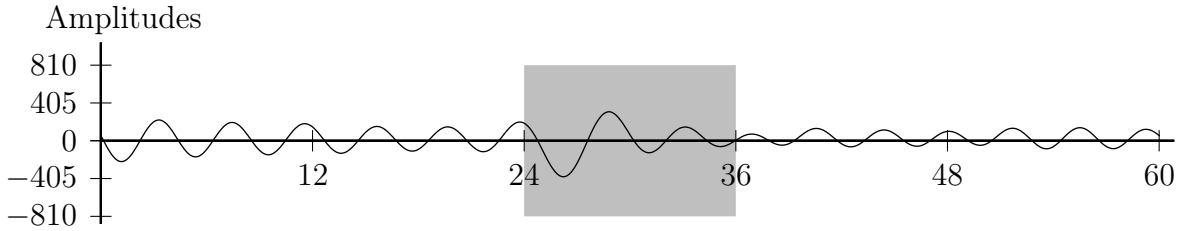
L'observation des différents signaux de la figure 5.10 ne démontre pas l'évidence flagrante d'un artéfact audible. Toutefois, l'utilisation d'un filtre passe-bas à 200 Hz sur les mêmes signaux de la figure 5.10 démontre la présence d'une discontinuité visible et audible dans le stimulus modèle AS-0φ avec aucune phase (cf. figure 5.11).

Après de nombreuses observations et d'écoute de fichiers, il a été constaté que ces discontinuités surviennent lors d'un changement d'état bien précis du classificateur. La figure 5.12 montre que les discontinuités se produisent lorsque le classificateur déclare une trame non-harmonique entre deux trames mixtes. Il est important de mentionner que la déclaration de cette trame non-harmonique ne provient pas d'une erreur de classification, car selon les spécifications actuelles, le classificateur devait déclarer cette analyse non-harmonique.

La situation de la figure 5.12 entraîne une discontinuité de phases, car la trame mixte à $t+1$ qui suit la trame non-harmonique t possède des phases aléatoires qui ne sont plus cohérentes avec les anciennes phases des trames $t-1$ et $t-2$. Par conséquent, ce manque de cohérence des valeurs des phases entraîne des discontinuités audibles. Afin de valider



(a) Signal du stimulus modèle AS-2φ filtré passe-bas (200 Hz) avec 2 phases originales



(b) Signal du stimulus modèle AS-0φ filtré passe-bas (200 Hz) avec aucune phase originale

Figure 5.11 Signaux de la figure 5.10 filtrés passe-bas à 200 Hz

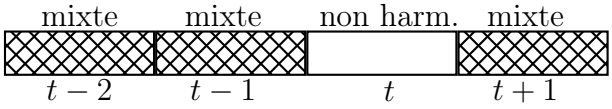


Figure 5.12 Schéma des endroits où se produisent les discontinuités

l'importance de la cohérence des phases entre les trames, un algorithme a été développé pour la gestion des états de discontinuités de la figure 5.12.

L'algorithme proposé prolonge les segments mixtes d'une longueur de trame de 12 ms. La figure 5.13 montre que durant une transition entre une trame mixte et une trame non-harmonique, l'algorithme recopie les informations harmoniques de la trame mixte précédente $t-1$ à la trame non-harmonique t afin de s'assurer une meilleure continuité entre les trames. Ainsi, la trame non-harmonique t devient un trame mixte t comme le montre la figure 5.13 .

L'algorithme s'utilise uniquement pour la première trame déclarée non-harmonique suivant une trame mixte, ce qui assure que le signal de synthèse ne possède pas un son trop synthétique. La figure 5.14(c) montre l'usage de l'algorithme sur le stimulus modèle AS-0φ filtré passe-bas à 200 Hz.

La figure 5.14(c) montre que l'algorithme élimine les discontinuités entre les trames. Lors de l'écoute des signaux de synthèse, les artéfacts audibles auparavant n'apparaissent plus lors de l'écoute. La prochaine partie montre les résultats obtenus du test MUSHRA avec le concept des phases.

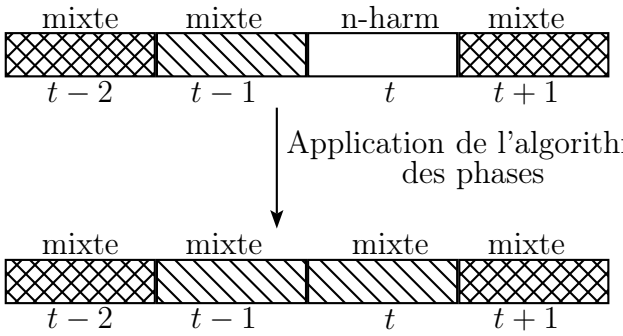
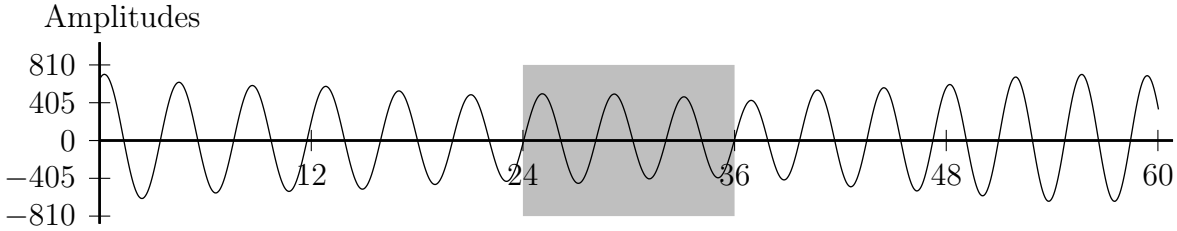
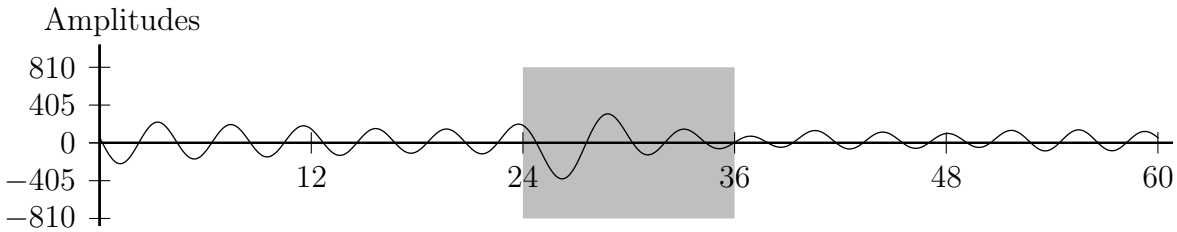


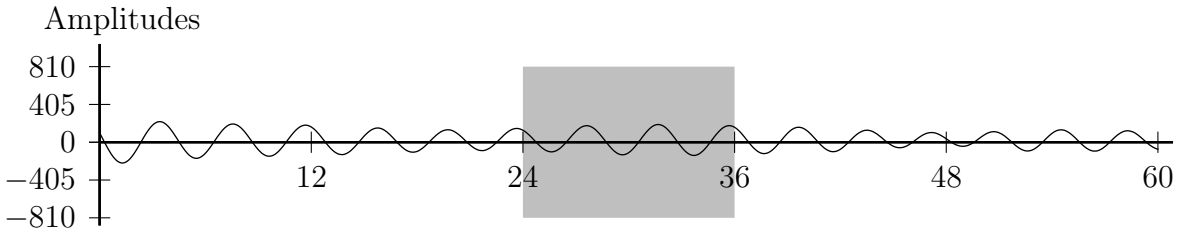
Figure 5.13 Schéma de fonctionnement de l'algorithme développé pour les phases



(a) Signal du stimulus modèle AS-2φ filtré passe-bas (200 Hz) avec 2 phases originales



(b) Signal du stimulus modèle AS-0φ filtré passe-bas (200 Hz) avec aucune phase originale



(c) Signal du stimulus modèle AS-Algo filtré passe-bas (200 Hz) avec aucune phase originale et ajout de l'algorithme

Figure 5.14 Résultat avec l'ajout de l'algorithme pour le signal 5.14(c)

Résultats des tests

La figure 5.15 montre les résultats MUSHRA pour la méthode des phases ainsi que les valeurs RSB segmentaires de chaque stimulus. Le calcul des valeurs RSB segmentaires ont

été effectués avec les mêmes équations 5.2 et 5.3 de la section 5.3.3. Ainsi, une valeur RSB élevée indique que le signal de synthèse est peu bruité tandis qu'une valeur de RSB faible signifie un signal de synthèse avec un niveau de bruit élevé comparativement au signal original.

La figure 5.15 montre les résultats obtenus pour les valeurs RSB segmentaires et les scores moyens MUSHRA de chacun des stimuli sur un total de 80 scores. Les valeurs RSB segmentaires de la figure 5.15 ont été normalisées de 0% à 100% en fonction du signal original qui lui possède une valeur RSB de 100%.

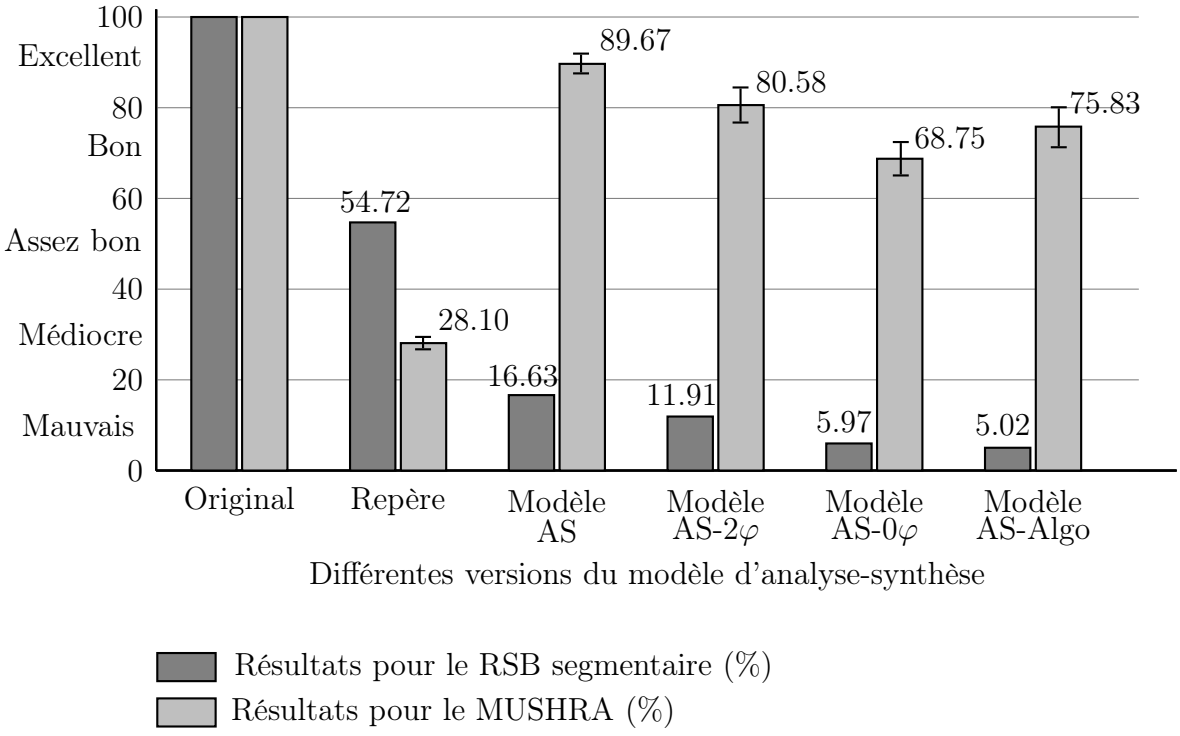


Figure 5.15 Résultats des tests RSB et MUSHRA avec différentes versions du modèle

Analyse des résultats

La figure 5.15 montre qu'il existe de grands écarts entre les résultats de tests objectifs (valeurs RSB segmentaires) et les résultats de tests subjectifs (valeurs MUSHRA) des stimuli. Cette première observation démontre que le modèle de codage par transformée réussit à obtenir un signal de qualité de bon à excellent sans nécessairement suivre la forme d'onde du signal original.

De plus, la figure 5.15 montre également que le stimulus avec uniquement 2 phases transmises réussit également à obtenir un signal de synthèse avec une bonne qualité tout en

allant plus loin dans l'abstraction de la forme d'onde du signal original. Les résultats du stimulus avec 2 phases et celui avec 0 phase transmise démontrent l'importance qu'il faut transmettre un minimum de phases pour obtenir un signal de synthèse de qualité. L'appareil auditif humain possède une grande sensibilité en basse fréquence, mais semble par la suite peu sensible sur l'exactitude des valeurs pour les phases.

Cette hypothèse de l'importance de la cohérence et non de la valeur absolue pour les phases se confirme par le stimulus qui ne contient aucune phase transmise, mais qui intègre l'algorithme de recouvrement des discontinuités de la section 5.3.4. Les résultats de la figure 5.15 montrent que le stimulus avec l'algorithme réussit à améliorer le signal de synthèse comparativement au stimulus qui n'utilise aucune phase.

5.3.5 Conclusion sur les tests MUSHRA et RSB

Cette partie du document a présenté les résultats du test subjectif MUSHRA et du test objectif RSB effectués sur les signaux de sortie du modèle d'analyse-synthèse et de la méthode de réduction des phases. Les résultats des tests objectif et subjectif proposent des résultats contradictoires. Les mauvais résultats des valeurs RSB démontrent que le signal de synthèse possède peu de similitudes avec le signal original tandis que les bons résultats MUSHRA démontrent que le signal de synthèse possède une bonne qualité perceptuelle audio de bon à excellent.

Ainsi, les résultats démontrent que le modèle réussit à obtenir un signal de synthèse de qualité dans le domaine fréquentiel sans nécessairement suivre la forme d'onde du signal original. De plus, la méthode de réduction des phases démontre également l'importance de la cohérence des phases entre les trames et de la moindre importance de leur valeur absolue. Cette méthode pousse encore plus loin le niveau d'abstraction possible avec la forme d'onde du signal original dans le modèle développé.

5.4 Test subjectif MOS sur le modèle quantifié

Un test subjectif MOS a été développé, afin de comparer le modèle quantifié du chapitre 4 avec la norme audio G.722.2 (AMR-WB, ***A**daptive **M**ulti **R**ate - **W**ide**B**and*) [UIT-T-G.722.2, 2003] de l'institut UIT (**U**nion **I**nternationale des **T**élécommunications). L'évaluation subjective MOS (***M**ean **O**pinion **S**core*) représente le test qui demande le plus de ressources pour tous les types de tests (subjectif et objectif).

Cette section présente toutes les étapes du test MOS de la création du plan d'expérience, de l'exécution du test avec les sujets et de l'analyse des résultats avec des tests de variances. Le test MOS suit les recommandations de la norme P.800 [UIT-T-P.800, 1996]. La prochaine section donne les détails du test MOS effectué.

5.4.1 Description du test subjectif MOS

Il existe plusieurs types de tests subjectifs MOS et pour le projet actuel, c'est l'évaluation par catégorie absolue (***A****bsolute* ***C****ategory* ***R****ating*, ACR) qui a été choisie. Ce test appartient au groupe de tests appelé essai d'opinion d'écoute. Ce test appartient à la norme P.800 qui décrit les différentes évaluations subjectives MOS [UIT-T-P.800, 1996]. Le test ACR représente la méthode d'évaluation la plus recommandée dans la catégorie des essais d'opinion d'écoute.

Avec la méthode ACR, les sujets de l'expérience entendent une seule fois la séquence, qui contient deux phrases dictées par le même locuteur. Après cette écoute, le sujet donne une note de qualité sur une échelle discrète de 1 à 5 comme l'indique le tableau 5.3.

Tableau 5.3 Échelle d'appréciation du test subjectif MOS

Qualité du signal entendu	Note
Excellente	5
Bonne	4
Passable	3
Médiocre	2
Mauvaise	1

Les résultats de cette méthode ont permis, à de nombreuses reprises, à l'aboutissement de recommandations telles que G.722.2 , G.726, G.728 et G.729 [UIT-T-P.800, 1996]. Avec la méthode ACR, il est possible d'obtenir des résultats homogènes avec différents laboratoires internationaux, en effectuant le même test ACR dans les mêmes conditions.

Choix des sujets pour l'expérience

Pour l'expérience, 30 sujets ont été choisis au hasard parmi une population normale. Les sujets non-experts ne possèdent aucune connaissance en codage audio ou autres domaines connexes. Le nombre de sujets féminins et masculins n'a pas été équilibré, car cela n'était pas requis pour le bon fonctionnement de l'expérience.

Création des séquences

L'expérience utilise la banque de sons de NTT-AT (***N**ippon **T**elegraph and **T**elephone corporation - **A**dvanced **T**echnology*) [NTT, 1994] reconnu dans le domaine du traitement du signal audio afin que cette expérience soit reproductible par d'autres laboratoires. La banque de sons contient 192 phrases dictées en français, dont 96 phrases par 4 voix différentes d'hommes et 96 phrases par 4 voix différentes de femmes. Les phrases dictées respectent les normes d'enregistrement de la recommandation P.800 [UIT-T-P.800, 1996] de l'institut UIT.

Les phrases possèdent une courte durée de 2 à 3 secondes et le texte qu'il contient est simple et clair à comprendre. Le texte des phrases provient d'un vocabulaire courant (texte d'un journal par exemple) et ne possède aucun terme technique. Les phrases ont un ordre aléatoire et ne possèdent pas de sens entre elles.

Pour l'expérience, une séquence contient deux phrases et un intervalle entre ces deux phrases. L'intervalle entre les deux phrases possède toujours la même longueur de 300 ms. Il est important de posséder toujours la même longueur d'intervalle et que les sujets puissent entendre le processus d'adaptation du codec entre les deux phrases. Le tableau donne des exemples de séquences entendues durant l'expérience.

Tableau 5.4 Exemples de séquences provenant du test MOS

Et que devient Paul dans tout cela ?	**Intervalle**	Baudelaire donna un nouveau souffle à la poésie.
Les poubelles passent le mardi et le jeudi.	**Intervalle**	Téléphoner est dangereux pour les oreilles
Il n'y a plus de beurre dans le frigidaire.	**Intervalle**	En cas d'incendie ne pas paniquer.

Description des conditions à évaluer

Le tableau 5.5 montre les conditions à vérifier dans le test MOS. Chaque condition du tableau applique un traitement à toute la banque de sons, ce qui fait un total de 1152 séquences à tester (12 conditions x 96 séquences). Dans les conditions de test, il y a le signal original qui représente le signal de référence pour le niveau de qualité du test MOS. Le modèle contient également quatre conditions de référence appelée MNRU (***M**odulated **N**oise **R**eference **U**nit*) [UIT-T-P.810, 1996].

Les conditions de référence MNRU représentent des signaux audio auxquels du bruit modulé a été ajouté au signal selon le niveau Q désiré en dB (cf. équation 5.4) [UIT-T-P.810,

Tableau 5.5 Description des conditions contenues dans le test MOS

Conditions	Codec	Débits (kbit/s)
01	Original	-
02	MNRU Q=15 dB	-
03	MNRU Q=25 dB	-
04	MNRU Q=35 dB	-
05	MNRU Q=45 dB	-
06	AMR-WB (G.722.2)	12.65
07	AMR-WB (G.722.2)	15.85
08	AMR-WB (G.722.2)	19.85
09	AMR-WB (G.722.2)	23.85
10	Modèle développé	20.14
11	Modèle développé	24.35
12	Modèle développé	30.71

1996]. La variable Q représente le rapport entre la puissance des signaux et la puissance du bruit modulé.

$$y(n) = x(n)[1 + 10^{-Q/20}N(n)] \tag{5.4}$$

$x(n)$: Signal d'entrée

$N(n)$: Bruit aléatoire

Les conditions de référence MNRU permettent d'obtenir un étalonnage de l'échelle d'appréciation montré dans le tableau 5.4. Un test MOS contient des conditions de références MNRU afin que les différentes expériences qui s'effectuent ailleurs ou à un autre moment puissent se comparer. Ces conditions de référence permettent également de rejeter les résultats de certains sujets, lorsque ceux-ci ne possèdent pas l'étalonnage prévu pour les conditions de référence MNRU.

De plus, le test s'assure que tous les signaux possèdent la même énergie d'amplitudes en leur appliquant tous un seuil de 26 dB. La prochaine section donne les détails sur le plan d'expérience effectué ainsi que les critères de randomisation.

5.4.2 Plan d'expérience et randomisation

Le tableau 5.5 montre que l'expérimentation contient 12 conditions à vérifier et que le test contient 12 séquences par locuteur, cela fait 1152 séquences pour le test complet (12

conditions x 12 séquences x 8 locuteurs). Le tableau 5.6 présente le plan de l'expérience complet qui contient les différents facteurs (I à IV). Le plan d'expérience du tableau 5.6 se sépare en six blocs où chaque bloc contient deux séquences ainsi que toutes les conditions de ces séquences.

Tableau 5.6 Plan d'expérience pour le test MOS

I-Type de locuteur		Femme				Homme			
III-Séquences	IV-Conditions II-Locuteurs	f1	f2	f3	f4	h1	h2	h3	h4
séquence 1	condition 01								
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	condition 12								
séquence 2	condition 01								
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	condition 12								
séquence 3	condition 01								
	⋮								
	condition 12								
séquence 4	condition 01								
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	condition 12								
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
séquence 11	condition 01								
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	condition 12								
séquence 12	condition 01								
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	condition 12								

Pour chaque bloc, cinq sujets effectuent le test d'écoute MOS deux fois. Ainsi, chaque bloc recueille 160 scores par condition (2 x [2 séquences x 8 locuteurs x 5 sujets]) et ce qui donne pour l'expérience totale, 960 scores pour chaque condition (160 scores x 6 blocs). Les sujets d'un groupe évaluent un bloc soit deux séquences du tableau de la figure 5.6. Le test d'écoute dure environ 60 minutes.

Chaque condition du tableau 5.5 crée des signaux de synthèse avec des niveaux de qualité différents qu'il faut considérer lors de la création de l'ordre des séquences pour l'écoute. L'ordre des séquences doit s'assurer de ne privilégier aucune des conditions du tableau 5.5. Un exemple possible est de toujours valoriser une condition en attribuant précédemment une condition avec une qualité inférieure afin d'obtenir systématiquement de bons scores.

Afin de ne valoriser aucune condition, l'expérience utilise la permutation aléatoire de Fisher-Yates [Knuth, 1997] pour générer la liste et de s'assurer que les permutations possèdent toute la même probabilité d'apparaître. Le modèle utilise l'algorithme de Fisher-Yates afin de déterminer l'ordre des conditions, l'ordre des locuteurs et l'ordre des séquences. Cependant, comme les facteurs conditions, locuteurs et séquences ne possèdent pas le même nombre de niveaux (conditions = 12, locuteurs = 8 et séquences = 2) (cf. tableau 5.6), il est nécessaire d'effecteur une gestion de l'ordre des facteurs lorsque l'ordre de Fisher-Yates ne peut être respecté.

La gestion de l'ordre survient avec les facteurs locuteurs et séquences, car l'ordre du facteur conditions ne varie jamais puisque c'est le critère le plus important. Ainsi, lorsque le locuteur de la séquence choisi par l'algorithme a déjà été mis dans la liste, l'algorithme applique une itération afin de trouver une autre séquence du même locuteur ou sinon un autre locuteur du même sexe. Cependant, lorsque cela n'est pas possible l'algorithme choisit les autres locuteurs disponibles. Le plus important dans la randomisation est que l'ordre ne varie jamais pour le facteur conditions. Le test applique la liste finale de randomisation aux 6 blocs de l'expérimentation. La prochaine section montre les résultats obtenus des tests MOS.

5.4.3 Résultats du test subjectif MOS

Cette section propose les résultats du test MOS effectué par 30 sujets non-experts choisis aléatoirement et qui a permis d'obtenir 960 scores pour chaque condition. La figure 5.16 compare les moyennes des scores MOS du modèle quantifié avec la norme G.722.2 pour différents débits. La ligne horizontale sur le graphique représente la note moyenne de 4.53 obtenue par le signal original.

Sur la figure 5.16, il se distingue deux courbes de qualité selon les différents modèles. La courbe du modèle quantifié (score = 3.86) possède une qualité légèrement inférieure (différence de 0.41) à la norme G.722.2 (score = 4.27) au débit autour de 24 kbit/s. Sur une échelle discrète comme le test MOS, les deux modèles à 24 kbit/s se situent dans la même catégorie moyen avec un score autour de 4. Les modèles ne peuvent se comparer à des débits plus élevés, car la norme G.722.2 ne possède pas une version au-delà de 24 kbit/s.

Les trois débits choisis pour le modèle quantifié permettent de visualiser le point d'inflexion de la courbe et montrent que le modèle possède une perte de qualité importante à partir de 20 kbit/s. Cependant, cette perte de qualité du modèle est prévisible en raison du manque de débit disponible afin de quantifier la partie bruit des spectres.

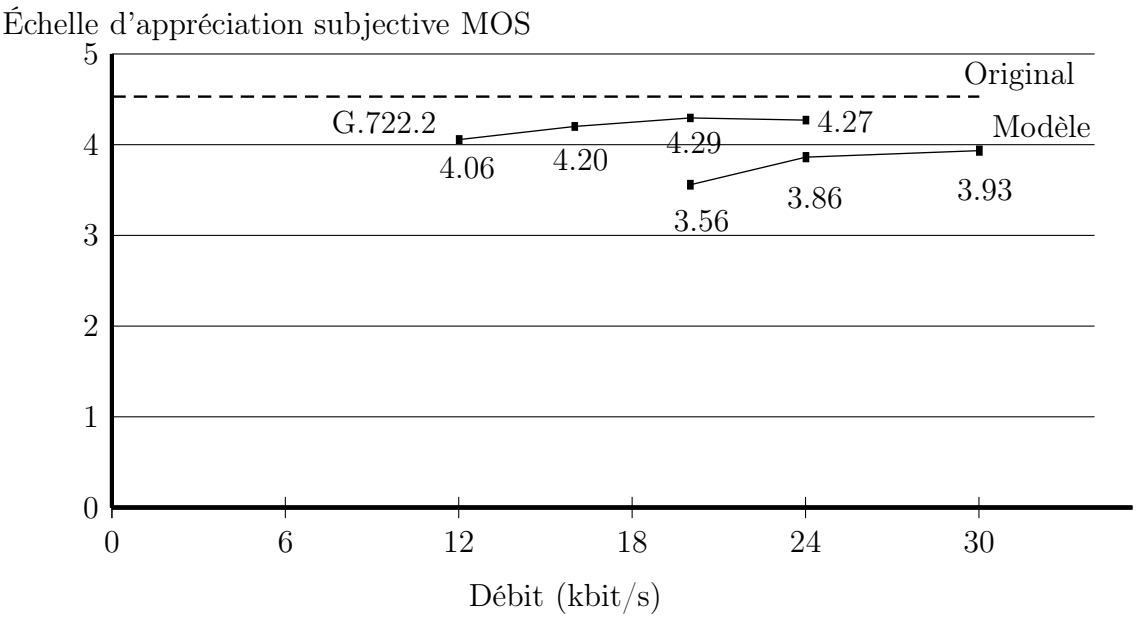


Figure 5.16 Résultats MOS des différentes versions du modèle quantifié

Les figures 5.17 et 5.18 montrent les résultats obtenus selon le type de locuteur entendu durant l'expérience. En comparant les figures 5.17 et 5.18, il ne semble pas exister de différence majeure de qualité entre les résultats obtenus pour les différents types de locuteurs selon les différentes conditions. Selon la figure 5.18, le modèle quantifié semble obtenir une performance légèrement meilleure avec des locuteurs masculins. Les prochaines sections proposent une analyse des variances (ANOVA, ***AN****alysis* ***Of*** ***VA****riance*) des résultats MOS afin de déterminer les moyennes ayant réellement des différences significatives.

5.4.4 Description générale d'une ANOVA à 1 facteur

L'analyse de variance (ANOVA, ***AN****alysis* ***Of*** ***VA****riance*) compare les moyennes en analysant les sources de variations entre les données. Les résultats de l'ANOVA déterminent l'existence de variation significative entre les moyennes, mais ne mentionnent pas lesquelles des moyennes possèdent des différences. Par la suite, c'est les tests post-hoc qui déterminent les moyennes qui diffèrent.

L'ANOVA représente une généralisation du test de Student et s'utilise lorsqu'il y a plus de deux comparaisons à effectuer. L'utilisation du test de Student sur plusieurs comparaisons augmente les risques d'effectuer une erreur puisque les calculs utilisent les mêmes données sur plus d'une comparaison.

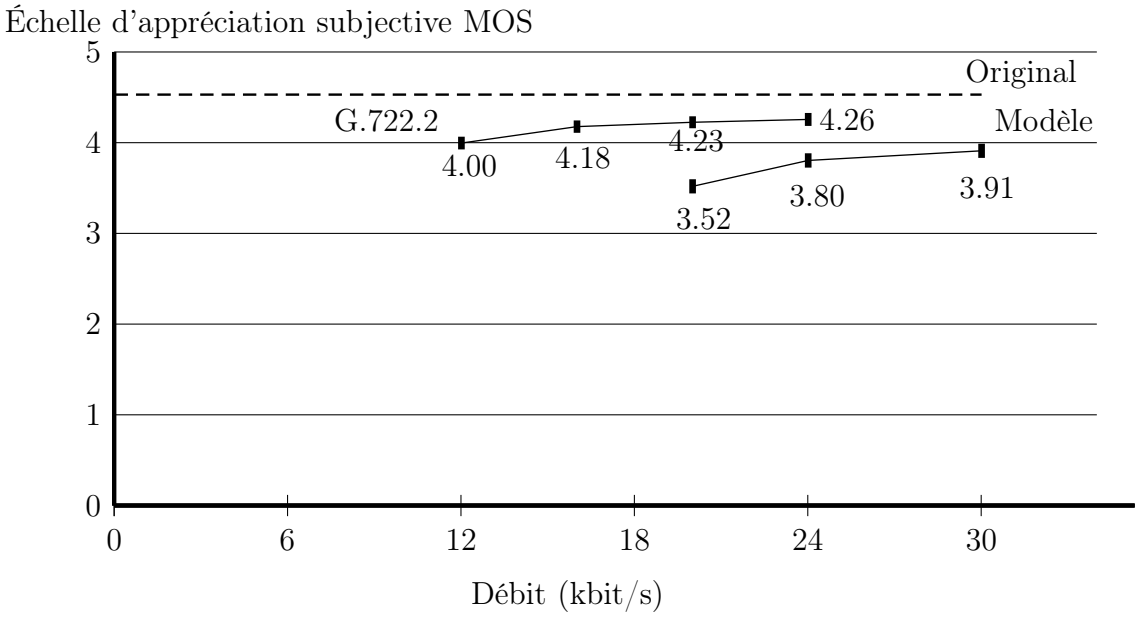


Figure 5.17 Résultats MOS avec des locuteurs féminins

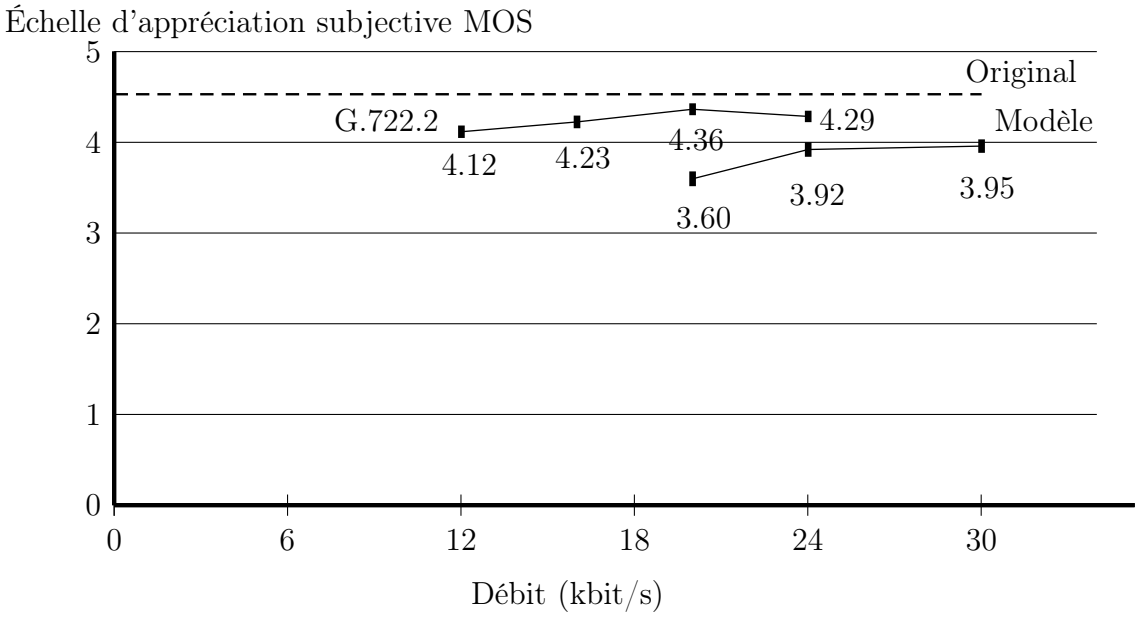


Figure 5.18 Résultats MOS avec des locuteurs masculins

Par exemple, si la probabilité de faire une erreur pour chaque comparaison est de 5% (P(erreur $\alpha = 5\%$)) et que la probabilité d'effectuer au moins une erreur α lorsque nous effectuons N comparaisons en paire, l'erreur devient $1 - (1 - \alpha)^N$. Ainsi, pour $N = 10$ comparaisons en paire, cela donne une probabilité d'erreur de 40%. L'analyse de variance contourne le problème de l'augmentation de l'erreur α en effectuant une seule comparaison.

Lorsque l'ANOVA s'utilise pour deux comparaisons, l'analyse retourne la même conclusion que le test de Student.

Cependant, avant d'utiliser une analyse de variance les données expérimentales doivent respecter les critères présentés dans le tableau 5.7.

Tableau 5.7 Conditions pour l'utilisation d'une ANOVA

- Indépendance des observations
- Normalité de la population
- Homoscédasticité des variances (variances uniformes)

Vérifications des conditions pour le test ANOVA

Les sujets ont été choisis aléatoirement et ne possèdent aucune expertise dans le domaine du traitement du signal. De plus, chaque phrase du test ne possède également aucune cohérence entre elles. Ainsi, le choix aléatoire des sujets et des phrases démontrent l'indépendance des observations.

De plus, afin de s'assurer de respecter les conditions du tableau 5.7, une transformation Box-Cox [Box et Cox, 1964] est appliquée sur les données brutes des tests MOS. Elle représente la transformation non linéaire la plus répandue en statistique et se définit selon l'équation 5.5. Dans l'équation 5.5, la variable x et qui représente les données brutes, possède toujours une valeur positive.

$$B(x,\lambda) = \begin{cases} \dfrac{x^{\lambda}-1}{\lambda} & \text{si } \lambda \neq 0 \\ \\ \log(x) & \text{si } \lambda = 0 \end{cases} \tag{5.5}$$

La valeur λ se calcule à l'aide du graphique de la vraisemblance en fonction de λ. Pour le test MOS de cette section, une valeur $\lambda = 2.1$ a été appliquée sur les données brutes de l'expérience avec l'équation 5.5 afin de s'assurer de respecter les critères ANOVA du tableau 5.7.

Concept de l'ANOVA

L'idée d'une ANOVA consiste à comprendre la source de la variabilité totale. L'analyse de variance détermine si la variabilité observée avec les moyennes provient du hasard ou s'il existe effectivement des différences significatives. La variabilité peut provenir des différentes conditions dues à la différence de traitement (intergroupe) et une autre partie

de la variabilité peut également provenir de l'intérieur de chaque groupe (intragroupe). Le tableau 5.8 montre les causes de variabilité entre les groupes et à l'intérieur d'un groupe.

Tableau 5.8 Les causes de variabilité des moyennes

Intergroupe	Intragroupe
- Les différences individuelles	- Les différences individuelles
- Les erreurs expérimentales	- Les erreurs expérimentales
- L'effet du traitement	

L'équation 5.6 montre la décomposition effectuée par l'ANOVA avec les différentes causes de variabilité du tableau 5.8.

$$\mathrm{F} = \frac{\text{Intergroupe}}{\text{Intragroupe}} = \frac{\text{effet. trait.} + \text{diff. ind.} + \text{erreur}}{\text{diff. ind.} + \text{erreur}} \tag{5.6}$$

L'équation 5.6 montre que lorsque le rapport F possède une valeur près de 1, il n'existe pas d'effet de traitement. Ainsi, si le rapport F est vraiment supérieur à 1 cela signifie la présence d'un effet de traitement important. Les prochaines parties expliquent comment l'ANOVA décompose l'équation 5.6.

Répartition de la somme des carrés moyens

Dans la situation des ANOVA, les variances ne peuvent s'additionner, car les données ne s'additionnent pas, mais se comparent sur deux angles différents : intergroupe et intragroupe. Cependant, la somme des carrés (SC) possède cette propriété d'additivité, car elle représente le résultat de l'addition entre la $\mathrm{SC}_{\mathrm{S|A}}$ intergroupe et la SC_{A} intragroupe (cf. figure 5.8) [Montgomery, 2009].

$$\sum_i \sum_j (y_{ij} - \bar{y}_{..})^2 = n \cdot \sum_i (\bar{y}_{i.} - \bar{y}_{..})^2 + \sum_i \sum_j (y_{ij} - \bar{y}_{i.})^2 \tag{5.7}$$

$$\mathrm{SC_T} = \mathrm{SC_{S|A}} + \mathrm{SC_A} \tag{5.8}$$

Répartition des degrés de liberté

Chaque SC calculée précédemment y est associée un degré de liberté (dl) qui possède également la propriété d'additivité (cf. équation 5.10) [Montgomery, 2009].

$$pn - 1 = p(n-1) + (p-1) \tag{5.9}$$
$$\mathrm{dl_T} = \mathrm{dl_{S|A}} + \mathrm{dl_A} \tag{5.10}$$

Non-répartition des carrés moyens

Les valeurs des carrés moyens (CM) s'obtiennent en divisant les valeurs SC par leur degré de liberté respectif. Contrairement aux variables SC et dl, les valeurs CM ne possèdent pas la propriété d'additivité (cf. équation 5.11).

$$\mathrm{CM_T} \neq \mathrm{CM_{S|A}} + \mathrm{CM_A} \tag{5.11}$$

Les hypothèses

Ainsi, les ANOVA possèdent deux hypothèses dont la première hypothèse H_0 prédit qu'il n'existe aucune différence entre les moyennes (cf. équation 5.12) tandis que la seconde hypothèse H_1 indique qu'il existe au moins une différence de moyenne (cf. équation 5.13).

$$H_0 = \mu_1 = \mu_2 = ... = \mu_p \tag{5.12}$$
$$H_1 = \text{il existe au moins une } \mu_i \neq \mu_j \tag{5.13}$$

Ainsi, le rapport F de l'équation 5.6 devient l'équation 5.14 avec les calculs des carrés moyens.

$$\mathrm{F} = \frac{\text{Intergroupe}}{\text{Intragroupe}} = \frac{\text{effet. trait.} + \text{diff. ind.} + \text{erreur}}{\text{diff. ind.} + \text{erreur}} = \frac{\mathrm{CM_A}}{\mathrm{CM_{S|A}}} \tag{5.14}$$

5.4.5 ANOVA à 2 facteurs du test MOS (conditions et type de locuteur)

La première ANOVA s'est effectuée avec deux facteurs : les conditions et le type de locuteur (femme et homme). Une ANOVA à deux facteurs ressemble à une analyse à un facteur et

possède l'avantage de déterminer également les interactions possibles entre ces facteurs. Ainsi, cette analyse de variance à deux facteurs vérifiera s'il existe une relation entre les conditions (A) et le type de locuteur (B) sur le score MOS obtenu.

Les observations précédentes des courbes de résultats des tests des figures 5.17 et 5.18 semblaient démontrer peu de différence entre les résultats des scores pour les voix de femmes et les voix d'hommes en fonction des différentes conditions. L'analyse de variance confirmera ou infirmera les observations effectuées sur les figures 5.17 et 5.18. Le tableau 5.9 montre le cumulatif des scores MOS obtenu par les différentes conditions (A) selon le type de locuteur (B).

Tableau 5.9 Cumulatif des scores MOS par condition et type de locuteur

Conditions (A)	**Cumulatif des scores selon le locuteur (B)** Femme (b1)	Homme (b2)	**Somme**
Référence (a1)	11698.534	11793.775	23492.310
AMR-WB 12.65 (a2)	9191.237	9743.800	18935.037
AMR-WB 15.85 (a3)	10030.909	10283.003	20313.913
AMR-WB 19.85 (a4)	10241.616	10937.276	21178.892
AMR-WB 23.85 (a5)	10456.117	10512.144	20968.261
Modèle 20.14 (a6)	7217.752	7582.731	14800.483
Modèle 24.35 (a7)	8430.726	8966.211	17396.937
Modèle 30.71 (a8)	8893.686	9064.561	17958.247
Somme	76160.578	78883.501	155044.079

Le tableau 5.10 montre le résultat de l'analyse de variance à deux facteurs obtenue à l'aide des données du tableau 5.9. Afin de déterminer les égalités de variance du tableau ANOVA 5.10, des tests de Fisher s'appliquent sur ces données.

Tableau 5.10 Analyse ANOVA pour les différentes conditions et locuteurs

Sources de variation	Somme des carrés SC	Degré de liberté dl	Carré moyen CM	Ratio F_{obs}
Intergroupe	54145.768	15		
Conditions (A)	52776.829	7	7539.547	138.128
Locuteurs (B)	965.405	1	52776.829	966.898
Interaction (AB)	403.533	7	57.648	1.056
Intra groupe (S|AB) (erreur)	418329.295	7664	54.584	
Total	472475.063	7679		

Le premier test de Fisher vérifie s'il existe une interaction entre les facteurs A et B du tableau 5.9 sur le score MOS. La première hypothèse se pose comme suit :

$$H_0 \quad : \quad \text{Absence d'interaction entre A x B (conditions x types de locuteurs)} \tag{5.15}$$

Ainsi, si $\mathrm{F_{obs}}$ du tableau 5.10 possède une valeur supérieure à $\mathrm{F_{crit}}$ de H_0 (cf. équation 5.15) cela démontre une présence d'interaction. La valeur critique $\mathrm{F_{crit}}$ de 2.011 pour l'hypothèse H_0 se calcule avec la table de Fisher (seuil $\alpha = 5\%$), ainsi qu'avec les degrés de liberté $\mathrm{dl_{S|AB}}$ et $\mathrm{dl_e}$ du tableau ANOVA 5.10.

Le tableau ANOVA 5.10 démontre que $\mathrm{F_{obs}} = 1.056 < \mathrm{F_{crit}} = 2.011$, ce qui signifie qu'il n'existe aucune relation entre les différentes conditions et le type de locuteurs sur les résultats des scores MOS. Ainsi, les conditions possèdent une qualité uniforme pour tous les types de locuteurs.

La prochaine étape vérifie avec des tests de Fisher les effets principaux des facteurs (cf. équations 5.16 et 5.17). La vérification s'effectue indépendamment puisqu'il n'existe pas de relation entre ces deux facteurs.

$$H_{0A} \quad : \quad \text{Absence d'effet du facteur A (conditions)} \tag{5.16}$$

$$H_{0B} \quad : \quad \text{Absence d'effet du facteur B (types de locuteurs)} \tag{5.17}$$

Le tableau 5.11 montre les résultats des calculs pour $\mathrm{F_{obs}}$ et $\mathrm{F_{crit}}$ pour les deux facteurs. Les résultats démontrent l'existence d'effet significatif pour les deux facteurs puisque les $\mathrm{F_{obs}}$ possèdent des valeurs plus élevées que les $\mathrm{F_{crit}}$. Ainsi, les différentes conditions possèdent un impact sur les résultats des tests. La section 5.4.6 effectuera une ANOVA à un facteur sur les conditions afin de déterminer l'existence de différences significatives dans les scores selon les différentes conditions.

Tableau 5.11 Résultats des tests de Fisher

	$\mathrm{F_{obs}}$	$\mathrm{F_{crit}}$	dl pour table F
H_{0A}	138.128	2.011	$(\mathrm{dl_A}, \mathrm{dl_e})$
H_{0B}	966.898	3.843	$(\mathrm{dl_B}, \mathrm{dl_e})$

De plus, le tableau 5.11 montre que le type de locuteur possède un impact sur les scores du test MOS. Selon les observations des figures 5.17 et 5.18 ainsi que les conclusions du test de Fisher, ces résultats démontrent que lorsque le locuteur est un homme, la probabilité d'obtenir un score plus élevé qu'une voix de femme est à 95%.

Ainsi, les facteurs possèdent des impacts sur les scores du test MOS, mais ne possèdent pas d'interaction entre eux. Le résultat de l'ANOVA démontre que le modèle quantifié possède une qualité uniforme pour la synthèse des voix d'hommes et de femmes (cf. figures 5.17 et 5.18). Ainsi, malgré l'augmentation du niveau de difficulté avec une fréquence fondamentale faible (voix d'homme) lors de l'analyse du spectre harmonique, le modèle réussit à obtenir un signal de synthèse de qualité comparable à des voix de femmes. Les fréquences fondamentales avec une faible valeur possèdent des partiels ayant des distances plus petite entre eux et ce qui rend leur détection plus difficile dans le spectre.

La prochaine section effectue une ANOVA à un facteur sur les conditions afin de déterminer l'existence de différences significatives de scores, et ainsi confirmer les résultats obtenus par l'ANOVA du tableau 5.10.

5.4.6 ANOVA à 1 facteur du test MOS (conditions)

Cette section propose une analyse ANOVA à un facteur sur les résultats obtenus du test pour tous les types de locuteurs de la figure 5.16. L'ANOVA déterminera s'il existe des différences significatives des résultats MOS en fonction des différentes conditions. L'observation de la figure 5.16 et du tableau 5.12 semble démontrer qu'il existe des différences entre les moyennes, toutefois l'ANOVA déterminera si les différences sont significatives.

Tableau 5.12 Résultats du test MOS selon les conditions

Conditions (A)	**Score MOS** Moyenne
Référence (a1)	24.471
AMR-WB 12.65 (a2)	19.724
AMR-WB 15.85 (a3)	21.160
AMR-WB 19.85 (a4)	22.061
AMR-WB 23.85 (a5)	21.842
Modèle 20.14 (a6)	15.417
Modèle 24.35 (a7)	18.122
Modèle 30.71 (a8)	18.707
Moyenne totale	20.188
Écart-type total	7.844

Après les calculs effectués à l'aide des équations de la section 5.4.4 cela donne les résultats du tableau ANOVA 5.13.

Par la suite, c'est le test de Fisher qui vérifie s'il existe une réelle différence entre les moyennes des différentes conditions. Le test de Fisher propose les hypothèses suivantes :

Tableau 5.13 Analyse ANOVA pour les différentes conditions

Sources de variation	Somme des carrés SC	Degré de liberté dl	Carré moyen CM	Ratio F_{obs}
Intergroupe (A)	52776.829	7	7539.547	137.821
Intra groupe (S|A) (erreur)	419698.234	7672	54.705	
Total	472475.063	7679		

H_0 : Il n'existe pas de différence entre les moyennes dans le tableau 5.12 (5.18)

H_1 : Il existe au moins une différence de moyenne dans le tableau 5.12 (5.19)

Puisque $F_{obs} = 137.821 > F_{crit} = 2.011$, cela signifie qu'il existe au moins une différence significative de moyenne entre les conditions. L'analyse de variance ne permet pas de déterminer laquelle ou lesquelles des moyennes diffèrent. Il faut effectuer un test de type post-hoc afin de déterminer les moyennes qui varient significativement.

Test-post hoc de Tukey (HSD)

Le test HSD (***H**onestly **S**ignificant **D**ifference*) aussi appelé de Tukey représente l'un des nombreux tests post-hoc possibles afin de comparer les moyennes et déterminer celles qui sont significativement différentes. Il existe plusieurs tests post-hoc comme le test de Fisher (LSD, ***L**east **S**ignificant **D**ifference*), SNK (***S**tudent-**N**ewman-**K**euls*), de Tukey (HSD, ***H**onestly **S**ignificant **D**ifference*) et de Scheffé qui représentent quelques exemples.

Il existe un grand nombre de tests post-hoc et ce qui les différencies est la puissance statistique qui diffère d'un test à l'autre. Certains tests plus laxistes donnent facilement une valeur significative tandis que d'autres tests plus conservateurs donnent difficilement un résultat significatif.

Le test de Fisher représente l'un des tests le plus laxiste et le test de Scheffé représente l'un des tests le plus conservateur. Il n'existe pas de règles dans le choix d'un test post-hoc. C'est le test de Tukey qui a été choisi comme test post-hoc pour vérifier les moyennes de l'expérience. Le test de Tukey est plus conservateur que le test de Fisher et plus laxiste que le test de Scheffé.

Le test HSD compare les moyennes par paires et l'hypothèse du test s'écrit comme suit :

$$H_0 \ : \ \text{Il n'existe pas de différence entre les deux moyennes comparées} \tag{5.20}$$

Ainsi, si $\mathrm{Q_{obs}}$ possède une valeur supérieure à $\mathrm{Q_{crit}}$ (cf. équation 5.20) cela signifie une différence significative de moyenne. La valeur critique $\mathrm{Q_{crit}}$ de 4.31 pour l'hypothèse H_0 se calcule avec la table de Tukey (seuil de $\alpha = 5\%$), ainsi qu'avec le nombre de conditions ($p = 8$) et le carré moyen de l'erreur ($\mathrm{CM_e} = 7672$) du tableau ANOVA 5.13.

$$Q_{\mathrm{obs}} = \frac{|\bar{X}_i - \bar{X}_j|}{\sqrt{\frac{\mathrm{CM}_e}{p}}} \tag{5.21}$$

L'équation 5.22 montre que si la différence de moyenne se situe au-delà de 1.029 cela signifie que les moyennes sont significativement différentes au seuil de 5%.

$$|\bar{X}_i - \bar{X}_j| > Q_{\mathrm{crit}} \cdot \sqrt{\frac{\mathrm{CM}_e}{p}} = 4.31 \cdot 0.239 = 1.029 \tag{5.22}$$

Le tableau 5.14 montre toutes les comparaisons des moyennes du tableau 5.12 des scores MOS. Les cellules grises du tableau 5.14 indiquent les moyennes qui sont équivalentes selon le critère de l'équation 5.22.

Les résultats de comparaison des moyennes du tableau 5.14 démontrent que trois versions du G.722.2 (15.85 kbit/s, 19.85 kbit/s et 23.85 kbit/s) ne possèdent pas des moyennes significativement différentes. De plus, le tableau montre que les modèles quantifiés avec les débits autour de 24 kbit/s et de 30 kbit/s ne possèdent pas des moyennes significativement différentes.

De plus, aucun codec ne possède une moyenne semblable au signal original. Les résultats du test de comparaison de Tukey confirment les observations déjà observées avec le graphique des scores MOS de la figure 5.16.

Tableau 5.14 Comparaisons multiples par paires avec le test de Tukey

Différences	$\bar{X}_{a1}$	$\bar{X}_{a2}$	$\bar{X}_{a3}$	$\bar{X}_{a4}$	$\bar{X}_{a5}$	$\bar{X}_{a6}$	$\bar{X}_{a7}$	$\bar{X}_{a8}$
$\bar{X}_{a1}$		4.747	3.311	2.410	2.629	9.054	6.349	5.765
$\bar{X}_{a2}$			1.436	2.337	2.118	4.307	1.602	1.037
$\bar{X}_{a3}$				0.901	0.682	5.743	3.039	2.454
$\bar{X}_{a4}$					0.219	6.644	3.940	3.355
$\bar{X}_{a5}$						6.425	3.720	3.135
$\bar{X}_{a6}$							2.705	3.289
$\bar{X}_{a7}$								0.585

Légende du tableau 5.14

$\bar{X}_{a1}$: Moyenne de la référence
$\bar{X}_{a2}$: Moyenne du G.722.2 (12.65 kbit/s)
$\bar{X}_{a3}$: Moyenne du G.722.2 (15.85 kbit/s)
$\bar{X}_{a4}$: Moyenne du G.722.2 (19.85 kbit/s)
$\bar{X}_{a5}$: Moyenne du G.722.2 (23.85 kbit/s)
$\bar{X}_{a6}$: Moyenne du modèle (20.14 kbit/s)
$\bar{X}_{a7}$: Moyenne du modèle (24.35 kbit/s)
$\bar{X}_{a8}$: Moyenne du modèle (30.71 kbit/s)

5.4.7 Conclusion sur le test MOS

Cette section du chapitre présentait les résultats du test subjectif MOS effectué sur le modèle quantifié afin de le comparer avec la norme G.722.2 [UIT-T-G.722.2, 2003] de l'institut UIT. Les résultats du test de la figure 5.16 démontrent qu'il se distingue deux courbes de qualité qui représente chacun des codecs : le modèle quantifié et la norme G.722.2. Le modèle développé (score=3.86) possède une qualité légèrement inférieure (différence de 0.41) à la norme G.722.2 (score=4.27) à un débit autour de 24 kbit/s. Cependant, sur l'échelle discrète du test MOS, les deux codecs se situent dans la même catégorie de qualité bonne avec un score autour de 4 pour un débit autour de 24 kbit/s.

Malgré le score légèrement plus bas du modèle quantifié, celui-ci possède l'avantage de démontrer qu'il est possible de compresser un signal de parole entièrement dans le domaine des fréquences et qu'il permet également d'envisager un modèle de codage universel entièrement dans ce domaine.

Cette section a également présenté les résultats MOS en fonction du type de locuteur (femme et homme) et des conditions. Les figures 5.17 et 5.18 montraient une légère différence de qualité sur les courbes, mais l'analyse de variance (ANOVA) a démontré que cette différence sur les courbes n'était pas significative. Ainsi, le modèle quantifié possède une qualité uniforme pour tous les types de locuteurs du test.

5.5 Conclusion du chapitre

Ce chapitre décrivait les tests et les analyses effectués sur le modèle développé du chapitre 3 et le modèle quantifié du chapitre 4 afin d'évaluer quatre éléments importants proposés : la précision du générateur d'impulsions de sinusoïdes, le modèle d'analyse-synthèse, la méthode de réduction des phases à transmettre et le modèle quantifié avec différents débits.

Ce chapitre contenait un test objectif et deux tests subjectifs différents. Le test objectif évaluait la précision du générateur d'impulsions. Le modèle propose un certain niveau d'indépendance qui permet d'utiliser différentes longueurs de transformées de Fourier lors de l'analyse et de la synthèse. Malgré un nombre de points plus faible pour le spectre de synthèse, quatre fois moins de points dans ce cas-ci, le modèle réussit tout de même à obtenir une grande précision à l'aide d'un générateur d'impulsions qui possède une table précalculée. Comme le montre les résultats des tests de la section 5.2, le générateur augmente la résolution naturelle du spectre de synthèse lors de l'ajout des partiels sans augmenter la complexité de calcul.

Le test MUSHRA de ce chapitre se segmentait en deux parties afin que la première partie évalue le modèle d'analyse-synthèse et que la seconde partie évalue de concept de réduction du nombre de phases à transmettre au décodeur. Le test MUSHRA a montré que le modèle d'analyse-synthèse réussit à obtenir un signal de qualité perceptuelle transparent dans le domaine de la transformée Fourier sans nécessairement suivre la forme d'onde du signal original.

Le test MUSHRA démontrait également que le signal de synthèse obtient également une bonne qualité provenant de différentes versions du modèle contenant un nombre de phases réduit. Ces différentes versions permettent d'aller plus loin dans l'abstraction de la forme d'onde du signal original tout en conservant une bonne qualité du signal synthèse.

Finalement, le chapitre présentait un test MOS afin d'évaluer le modèle quantifié avec la norme G.722.2 [UIT-T-G.722.2, 2003]. Les tests démontraient que le modèle quantifié possède une qualité légèrement inférieure à la norme G.722.2 à un débit autour de 24 kbit/s. Cependant, le modèle possède l'avantage de fonctionner entièrement dans le domaine de la transformée et démontre également la possibilité d'un codage universel et unifié dans ce domaine.

CHAPITRE 6
CONCLUSION

Au cours des trente dernières années, deux grandes familles de modèles de codage audio ont été développées en parallèle, l'une opérant dans le domaine temporel et l'autre dans le domaine fréquentiel. À chacune de ces deux grandes familles est associée un type de contenu audio particulier : les modèles dans le domaine temporel possèdent généralement une plus grande efficacité pour le traitement des signaux de parole, tandis que les modèles dans le domaine fréquentiel sont plus généraux et permettent de traiter les signaux complexes tels que la musique.

Ces deux familles utilisent une modélisation très différente du signal : un modèle de production de la parole pour les codeurs dans le domaine temporel, et un modèle de perception auditive pour les codeurs dans le domaine fréquentiel. Ils mettent également en oeuvre des outils différents, notamment la prédiction linéaire pour les modèles de codage dans le domaine temporel et la transformée MDCT (***M**odified **D**iscrete **C**osine **T**ransform*) pour les modèles de codage dans le domaine fréquentiel.

Ils se basent toutefois sur les mêmes deux grands principes[1] de suivi de la forme d'onde du signal original d'une part, et de la mise en forme du bruit de codage d'autre part. Les modèles temporels suivent la forme d'onde par un filtrage LPC (***L**inear **P**redictive **C**oding*) tandis que les codeurs fréquentiels utilisent un modèle perceptuel pour contrôler l'attribution de bits pour la mise en forme du bruit de codage.

Plus récemment, il y a eu une émergence de codecs hybrides universels qui peuvent traiter tous les types de signaux audio, autant les signaux de parole que les signaux de musique, avec une performance assez uniforme sur ces différents types de contenus. Actuellement, il existe les standards USAC (***U**nified **S**peech and **A**udio **C**oding*) de MPEG (***M**oving **P**icture **E**xperts **G**roup*) [ISO/IEC-23003-3, 2012], EVS (***E**nhanced **V**oice **S**ervices*) de 3GPP (***3**rd **G**eneration **P**artnership **P**roject*) [ETSI/TS-126-445, 2014] et Opus de IETF (***I**nternet **E**ngineering **T**ask **F**orce*) [IETF/RFC-6716, 2012].

Ces trois codecs fonctionnent selon le même principe : ce sont des codecs multi-modes qui alternent entre un mode de codage temporel et un mode de codage fréquentiel selon

[1]Cette remarque, ne s'applique pas aux codeurs à très bas débit tels que les vocodeurs. Pour ces codeurs, l'objectif est souvent de maximiser l'intelligibilité de la parole codée. On est généralement bien loin d'une qualité perceptuelle transparente.

les caractéristiques de la trame à traiter. Les trames de parole voisée sont généralement codées dans le domaine temporel, tandis que les autres trames sont codées dans le domaine fréquentiel. Ces codecs ne sont donc pas véritablement des codecs unifiés. Puisqu'ils intègrent un classificateur, ces codecs possèdent une sensibilité aux erreurs de classification (en particulier pour les signaux difficiles à classer tels que de la parole sur un fond de bruit ou de musique). Ils possèdent également une tendance à être des codecs plus complexes puisqu'il s'agit d'une juxtaposition d'au moins deux modes de codage très différents.

L'objectif de cette thèse de doctorat était d'établir les bases d'un modèle de codage de l'audio véritablement unifié (un modèle unique, générique, indépendant du type de signal à coder). Le modèle proposé dans le chapitre 3, sa version quantifiée dans le chapitre 4 ainsi que les résultats des évaluations subjectives présentées dans le chapitre 5 démontrent les éléments suivants :

- La possibilité d'atteindre (ou de s'approcher) de la transparence sans nécessairement suivre la forme d'onde du signal original. Les résultats du test subjectif MUSHRA (***MU****ltiple* ***S****timuli with* ***H****idden* ***R****eference and* ***A****nchor*) démontrent que le signal de sortie du modèle d'analyse-synthèse développé possède une qualité perceptuelle transparente ;

- La possibilité de coder efficacement la parole dans le domaine fréquentiel, ce qui pave la voie à un codec réellement unifié puisque le domaine fréquentiel est traditionnellement réservé aux signaux audio autres que les signaux de parole. Les résultats du test subjectif MOS (***M****ean* ***O****pinion* ***S****core*) avec le modèle quantifié qui fonctionne entièrement dans le domaine de la transformée, démontrent que le signal de synthèse possède une bonne qualité à des débits autour de 24 kbit/s et de 30 kbit/s. Le modèle quantifié à 24 kbit/s se situe dans la même catégorie que la norme G.722.2 (AMR-WB) [UIT-T-G.722.2, 2003] ;

- La possibilité de coder un signal audio dans le domaine fréquentiel sans être contraint par l'exigence de reconstruction parfaite qui caractérise les approches fréquentielles existantes. En particulier, nous avons montré que l'analyse (à l'encodeur) et la synthèse (au décodeur) des signaux audio peuvent être relativement indépendantes l'une de l'autre.

  Dans l'approche présentée au chapitre 3, l'analyse et la synthèse sont toutes deux effectuées dans le domaine fréquentiel, mais avec des résolutions fréquentielles très différentes. On peut même aisément imaginer des variantes dans lesquelles certains

éléments d'analyse s'effectueraient dans le domaine temporel (par exemple la mesure du pitch d'un signal de parole) ;

- Enfin, il est possible de coder l'audio dans le domaine fréquentiel de façon efficace sans avoir à respecter la contrainte d'échantillonnage critique qui caractérise les approches fréquentielles par MDCT (***M****odified* ***D****iscrete* ***C****osine* ***T****ransform*). L'usage d'une transformée de Fourier pour la synthèse facilite certaines opérations ; elle permet par exemple de faire de la prédiction dans le domaine fréquentiel. Cette démonstration de la prédiction possible dans le domaine de la transformée de Fourier a été effectuée dans le chapitre 4.

Les principaux concepts de base pour un codeur de parole et de musique réellement unifié ont donc été démontrés dans cette thèse. Il reste toutefois certains éléments à développer avant de disposer d'un codec complet et compétitif par rapport aux solutions déjà standardisées. L'optimisation de la performance de certains quantificateurs ainsi que l'amélioration du codage de la partie non-harmoniques des spectres de parole. Les parties non-harmoniques nécessitent une étude plus approfondie afin de bien caractériser ce type de son et ainsi optimiser le codage de ceux-ci. Finalement, étendre le modèle de codage développé pour des signaux polyphoniques et aux signaux de musique en général.

LISTE DES RÉFÉRENCES

Adoul, J.-P., Mabilleau, P., Delprat, M. et Morissette, S. (1987). Fast CELP coding based on algebraic codes. *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '87.*, volume 12, p. 1957 – 1960.

Atal, B. (1986). High-quality speech at low bit rates : Multi-pulse and stochastically excited linear predictive coders. *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86*, volume 11, numéro 1169247, p. 1681–1684.

Atal, B. et Remde, J. (1982). A new model of LPC excitation for producing natural-sounding speech at low bit rates. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '82*, volume vol.7, p. p.614–617.

Bello, J., Daudet, L., Abdallah, S., Duxbury, C., Davies, M. et Sandler, M. B. (2005). A tutorial on onset detection in music signals. *Speech and Audio Processing, IEEE Transactions on*, volume 13, numéro 5, p. p. 1035–1047.

Bosi, M., Brandenburg, K., Quackenbush, S., Fielder, L., Akagiri, K., Fuchs, H. et Dietz, M. (1997). ISO/IEC MPEG-2 Advanced Audio Coding. *J. Audio Eng. Soc*, volume 45, numéro 10, p. p.789–814.

Box, G. E. P. et Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, volume 26, numéro 2, p. 211–252.

Brandenburg, K. (1999). MP3 and AAC explained. *Audio Engineering Society Conference : 17th International Conference : High-Quality Audio Coding*.

Brandenburg, K. et Chiariglione, L. (2003). Projet : Simulation du principe de codage/décodage MP3 sous matlab. Website, http ://tcts.fpms.ac.be/cours/1005-03/projet2003-2004.pdf.

Brown, M. et Forsythe, A. (1974). Robust tests for equality of variances. *Journal of the American statistical association*, volume 69, numéro 346, p. 364–367.

Calliope (1989). *La Parole et son Traitement Automatique*. Dunod.

Dodge, Y. (2007). *Statistique : Dictionnaire encyclopédique*. Springer.

ETSI/TS-126-445 (2014). *Codec for Enhanced Voice Services (EVS) ; Detailed algorithmic description* (Rapport technique). ETSI.

Fant, G. (1960). *Acoustic Theory of Speech Production*. Mouton, The Hague.

FraunhoferIIS (2012). *The AAC Audio Coding Family for Broadcast The AAC Audio Coding Family for Broadcast and Cable TV* (Rapport technique). Fraunhofer IIS.

Gazor, S. et Zhang, W. (2003). Speech probability distribution. *Signal Processing Letters, IEEE*, volume 10, numéro 7, p. p. 204–207.

Gersho, A ; Gray, R. (1992). *Vector Quantization and Signal Compression*. Springer.

Gray, A., J. et Markel, J. (1974). A spectral-flatness measure for studying the autocorrelation method of linear prediction of speech analysis. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, volume 22, numéro 3, p. p. 201–217.

Griffin, D. et Lim, J. (1985). A new model-based speech analysis/synthesis system. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '85*, volume Vol.10, p. p.513–516.

Griffin, D. et Lim, J. (1988). Multiband excitation vocoder. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume Vol.36, p. p.1223–1235.

Hardy, D., Malleus, G. et Mereur, J.-N. (2002). *Réseaux : Internet, téléphonie, multimédia*. De Boeck Université.

Haton, J.-P. et Lamotte, M. (1971). Étude statistique des phonèmes et diphonèmes dans le français parlé dans le français parlé. *Revue d'Acoustique*, volume 16.

Herre, J. et Grill, B. (2000). Overview of MPEG-4 audio and its applications in mobile communications. *Communication Technology Proceedings, 2000. WCC - ICCT 2000. International Conference on*, volume 1, p. p.604–613.

Herre, J. et Johnston, J. (1996). Enhancing the performance of perceptual audio coders by using Temporal Noise Shaping (TNS). *Audio Engineering Society Convention 101.*

Herre, J. et Schultz, D. (1998). Extending the MPEG-4 AAC codec by perceptual noise substitution. *Audio Engineering Society Convention 104.*

IETF/RFC-6716 (2012). *Definition of the Opus Audio Codec* (Rapport technique). IETF.

ISO/IEC-11172-3 (1993). *Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5Mbit/s Part 3 : Audio* (Standard). ISO/IEC.

ISO/IEC-13818-7 (1997). *Information technology — Generic coding of moving pictures and associated audio information — Part 7 : Advanced Audio Coding (AAC)* (Standard). ISO/IEC.

ISO/IEC-14496-3 (2009). *Information technology — Coding of audio-visual objects — Part 3 : Audio* (Standard). ISO/IEC.

ISO/IEC-23003-3 (2012). *MPEG-D (MPEG Audio Technologies), Part 3 : Unified Speech and Audio Coding* (Standard). ISO/IEC.

ISO/IEC14496-3 (2005). Coding of audio-visual objects, part 3 : Audio. *ISO/IEC.*

ISO/IEC/JTC1/SC29/WG11 (1998). *Report on the MPEG-2 AAC Stereo Verification Tests* (Rapport technique). ISO/IEC.

JTC1/SC29/WG11 (2007). *Call for Proposals on Unified Speech and Audio Coding* (Standard). ISO/IEC.

JTC1/SC29/WG11.N6828 (2004). *MPEG-7 Overview (Version 10)* (Standard). ISO/IEC.

Knuth, D. E. (1997). *The Art of Computer Programming, Volume 1 (3rd Ed.) : Fundamental Algorithms*. Addison Wesley Longman Publishing Co., Inc.

Laflamme, C., Adoul, J.-p., Salami, R., Morissette, S. et Mabilleau, P. (1991). 16 kbps wideband speech coding technique based on algebraic CELP. *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, volume 1, p. p.13–16.

Laflamme, C., Adoul, J.-P., Su, H. et Morissette, S. (1990). On reducing computational complexity of codebook search in CELP coder through the use of algebraic codes. *International Conference on Acoustics, Speech, and Signal Processing, 1990. ICASSP-90*, volume Vol. 1, p. p. 177–180.

Makhoul, J., Viswanathan, R., Schwartz, R. et Huggins, A. (1978). A mixed-source model for speech compression and synthesis. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '78*, volume Vol. 3, p. p. 163–166.

McAulay, R. et Quatieri, T. (1985). Mid-rate coding based on a sinusoidal representation of speech. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '85*, volume Vol. 10, p. p. 945–948.

McAulay, R. et Quatieri, T. (1986). Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume Vol.34, p. p. 744–754.

McAulay, R. et Quatieri, T. (1987). Multirate sinusoidal transform coding at rates from 2.4 kbps to 8 kbps. *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '87*, volume 12, p. 1645–1648.

Meltzer, S. et Moser, G. (2006). *MPEG-4 HE-AAC v2 - Audio Coding for Today's Digital Media World* (Rapport technique). EBU (European Broadcasting Union).

Montgomery, D. (2009). *Design and Analysis of Experiments 7th (seventh) edition*. Wiley, John and Sons, Incorporated.

Neuendorf, M., Gournay, P., Multrus, M., Lecomte, J., Bessette, B., Geiger, R., Bayer, S., Fuchs, G., Hilpert, J., Rettelbach, N., Nagel, F., Robilliard, J., Salami, R., Schuller, G., Lefebvre, R. et Grill, B. (2009). A novel scheme for low bitrate unified speech and audio coding – MPEG RM0. *Audio Engineering Society Convention 126*.

Neuendorf, N., Multrus, M., Rettelbach, N., Fuchs, G., Nagel, F., Robilliard, J., Lecomte, J., Wilde, S., Bayer, S., Disch, S., Helmrich, C., Lefebvre, R., Gournay, P., Bessette, B., Lapierre, J., Kjörling, K., Purnhagen, H., Villemoes, L., Oomen, W., Schuijers, E., Kikuiri, K., Chinen, T., Norimatsu, T., Seng, C., Oh, E., Kim, M., Quackenbush, S. et Grill, B. (2012). MPEG unified speech and audio coding – the ISO/MPEG standard for high-efficiency audio coding of all content types. *Audio Engineering Society Convention 132*.

NTT (1994). `http://www.ntt-at.com/product/speech/`.

Pan, D. (1995). A tutorial on MPEG/audio compression. *Multimedia, IEEE*, volume 2, numéro 2, p. p.60–74.

Quatieri, T. et Danisewicz, R. (1990). An approach to co-channel talker interference suppression using a sinusoidal model for speech. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, volume 38, numéro 1, p. p.56–69.

Richards, D. (1964). Statistical properties of speech signals. *Electrical Engineers, Proceedings of the Institution of*, volume 111, numéro 5, p. p. 941–949.

Richharia, M. et Westbrook, L. (2010). *Satellite Systems for Personal Applications : Concepts and Technology*. Wiley.

Rivier, E. (2003). *Communication audiovisuelle*. Paris : Springer.

Rossi, M. (2007). *Audio*. Presses Polytechinques et Universitaires romandes.

Rothauser, E., Chapman, W., Guttman, S., Hecker, M., Sordby, K., Silbiger, H., Urbanek, G. et Weinstock, M. (1969). Ieee recommended practices for speech quality measurements. *IEEE Transactions on Audio and Electroacoustics*, volume 17, numéro 3, p. 225–246.

Salami, R. (1995). *Analysis-by-Synthesis Predictive Speech Coding*. Université de Sherbrooke.

Schroeder, M. et Atal, B. (1985). Code-Excited Linear Prediction(CELP) : High-quality speech at very low bit rates. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '85*, volume Vol.10, p. p.937–940.

Singhal, S. et Atal, B. (1984). Improving performance of multi-pulse lpc coders at low bit rates. *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, volume 9, p. 9–12.

Spanias, A., Painter, T. et Atti, V. (2007). *Audio Signal Processing and Coding*. Wiley-Interscience.

Tremain, T. E. (1982). The govemment standard linear predictive coding algorithm : LPC-10. *Speech Technology*, p. 40–49.

UIT-R-BS.1387.1 (2001). *Méthode de mesure objective de la qualité du son perçu* (Standard). Union International des Télécommunications.

UIT-R-BS.1534.1 (2003). *Méthode d'évaluation subjective du niveau de qualité intermédiaire des systèmes de codage* (Standard). Union International des Télécommunications.

UIT-T-G.722.2 (2003). *Codage Vocal à Large Bande à 16 kbit/s Environ par Codage Adaptatif Multidébit à Large Bande (AMR-WB)* (Standard). Union International des Télécommunications.

UIT-T-G729 (2007). *Codage de la Parole à 8 kbit/s par Prédiction Linéaire avec Excitation par Séquences Codées à Structure Algébrique Conjuguée* (Standard). Union International des Télécommunications.

UIT-T-P.800 (1996). *Méthodes d'évaluation objective et subjective de la qualité* (Standard). Union International des Télécommunications.

UIT-T-P.810 (1996). *Appareil de référence à bruit modulé (MNRU)* (Standard). Union International des Télécommunications.

UIT-T-P.862 (2001). *Evaluation de la qualite vocale percue : méthode objective d'évaluation de la qualité vocale de bout en bout des codecs vocaux et des réseaux téléphoniques à bande Ètroite* (Standard). Union International des Télécommunications.

UIT-T-P.862.2 (2005). *Extension large bande de la Recommandation P.862 pour l'évaluation des codecs vocaux et réseaux téléphoniques à large bande* (Standard). Union International des Télécommunications.

VoiceAge (2011). `http://www.voiceage.com/technologies.php`.