



Faculté de génie
Génie électrique et génie informatique

Codage Audio Hiérarchique à Faibles Débits

Thèse de doctorat
Spécialité: génie électrique

Guillaume FUCHS

RÉSUMÉ

Avec l'essor de la téléphonie mobile et la démocratisation de l'Internet haut débit, les services de communications téléphoniques et de données convergent de plus en plus vers une coopération entre des réseaux multiples et variés. Le codage audio, essentiel pour établir une communication fiable, doit alors s'adapter aux différentes conditions des réseaux traversés ainsi qu'aux différentes applications visées. Le codage hiérarchique de l'information est une solution attrayante qui permet au récepteur de restituer le signal transmis avec une qualité variante selon le débit disponible. Néanmoins, une hiérarchisation complexifie le processus du codage et réduit la performance globale de compression de l'information.

L'objet de cette thèse est de rendre le codage hiérarchique à la fois efficace et très polyvalent. On veut en particulier traiter convenablement de la parole à faible débit ainsi que des signaux plus complexes pour des débits supérieurs. On exige en plus que le décodage puisse raffiner la restitution sonore pour un incrément très fin du débit.

Les schémas de codage hiérarchique développés au cours de cette thèse sont tous construits autour d'un codeur de parole auquel on adjoint un codage d'amélioration dans le domaine fréquentiel. Une telle association doit tirer profit au maximum des deux parties pour qu'elles puissent mutuellement se compléter et se compenser. Pour cette raison nous avons introduit un posttraitement fréquentiel pour les codeurs de parole ainsi qu'un annulateur de préécho pour le codage par transformée. De plus, pour que le décodage soit graduel, nous avons aussi introduit une extension de bande combinant estimation et codage des composantes fréquentielles ainsi qu'une quantification vectorielle algébrique à raffinements successifs.

Les codeurs hiérarchiques obtenus ont des débits allant de 8 kbit/s à 32 kbit/s. Bien qu'étant encore légèrement en retrait à débit égal et pour une source donnée, les performances de nos solutions se rapprochent de celles des codeurs optimisés pour un débit fixe et dédiés à un seul type de signal. Nos solutions hiérarchiques sont de plus les plus polyvalentes et les plus flexibles aux diverses conditions des réseaux de communication.

REMERCIEMENTS

Ce travail a été effectué 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 aussi mon directeur de thèse.

Que la compagnie VoiceAge, basée à Montréal, Québec, ainsi que le Ministère des Affaires étrangères du Canada reçoivent toute l'expression de ma reconnaissance pour avoir permis de mener à bien le projet.

Je remercie vivement les membres de mon jury de thèse en les personnes du Professeur Yves Bérubé-Lauzière, du Professeur Martin Bouchard et du Professeur Chon-Tam Le Dinh, pour s'être intéressés à mon travail et pour l'avoir évalué et corrigé.

Merci aussi à tous mes collègues et amis étudiants de l'Université de Sherbrooke qui se reconnaîtront ici. Je leur exprime ma profonde sympathie et leur souhaite beaucoup de succès dans leur carrière. Je souhaite particulièrement remercier mon collègue de bureau Mohamed, les membres du laboratoire NECOTIS dirigé par le Professeur Jean Rouat et les doctorants du CARTEL. Je remercie aussi la communauté internationale de l'Université de Sherbrooke et les amis du Club Plein Air.

Je remercie du fond du cœur mes parents et mon frère pour leur soutien à distance, mais précieux. Je remercie tous mes amis de Bretagne et surtout les *branleurs* de l'INSA, Damien, Gildas, Hervé, Jimmy, Jojo, les 2 Rom'1, Seb, Stéph et Yann (*Yec'hed mat!*). Pour finir, un très grand merci à Gregory, Kamal, Ramin, Shems, Wajdi et Teresa.

TABLE DES MATIÈRES

1	Introduction	1
1.1	L'évolution des télécommunications	1
1.2	Le contexte normatif en codage audio	3
1.3	Les avantages du codage audio hiérarchique	5
1.4	Positionnement du projet	7
1.5	Plan de la thèse	8
2	Le Codage Audio	11
2.1	Codage de source	11
2.2	Codage de la parole	14
2.2.1	Modélisation de la parole	14
2.2.2	Codage prédictif	15
2.2.3	Prédiction à court terme	17
2.2.4	Prédiction à long terme	19
2.2.5	Les différentes approches en codage de la parole	20
2.3	Codage par transformée	25
2.3.1	Modélisation de l'appareil auditif humain	25
2.3.2	Transformation Temps-Fréquence	29
2.3.3	Schéma de codage par transformée	35
2.4	Codage universel	36
2.5	Conclusion	39
3	Codage Audio Hiérarchique	41
3.1	Amélioration graduelle de la description	42
3.1.1	CELP imbriqué	42
3.1.2	CELP associé au codage par transformée	43
3.1.3	Codage prédictif par transformée	45

3.2	Extension graduelle de la bande	45
3.2.1	Extension par codage prédictif	47
3.2.2	Extension par codage par transformée	47
3.2.3	Extension artificielle ou hautement paramétrique	47
3.3	Conclusion	49
4	Posttraitement Fréquentiel pour Codeurs de Parole	51
4.1	Introduction	51
4.2	Caractéristiques du signal de synthèse de l'AMR-WB	51
4.3	Posttraitement fréquentiel	54
4.3.1	Principe	54
4.3.2	Caractéristiques du posttraitement	56
4.4	Codage du masque M	60
4.4.1	Codage sans perte	60
4.4.2	Codage avec pertes	63
4.4.3	Compromis débit/délai/qualité	70
4.4.4	Codage multimodal	72
4.5	Performances et tests	75
4.6	Conclusions	76
5	Quantification Algébrique à Raffinements Successifs	79
5.1	Introduction	79
5.2	Quantification vectorielle	80
5.2.1	Définitions	80
5.2.2	Les avantages de la quantification vectorielle	81
5.2.3	Quantification vectorielle par contrainte	85
5.3	Quantification par réseau régulier de points	86
5.3.1	Réseaux réguliers de points	86
5.3.2	Quantification par réseau régulier de points	89
5.3.3	LVQ en codage	91

5.4	LVQ à raffinements successifs	93
5.4.1	LVQ multidébit	94
5.4.2	Extension de Voronoï	96
5.4.3	Extension de Voronoï Graduelle	100
5.4.4	Quantification multidébit par extension de Voronoï graduelle	109
5.4.5	Génération du train binaire	116
5.5	Performances des quantifications multidébit	118
5.5.1	Cas d'une source gaussienne	118
5.5.2	Codage d'une cible TCX	122
5.6	Conclusion	124
6	Codage Hiérarchique en Bande Élargie	127
6.1	Introduction	127
6.2	Structure hiérarchique du codeur	128
6.2.1	Principe	128
6.2.2	Couche de base	130
6.2.3	Couche d'amélioration	130
6.2.4	Modèle perceptuel	132
6.2.5	Codage de la cible TCX	134
6.2.6	Performances du codeur hiérarchique	136
6.3	Optimisations perceptuelles	139
6.3.1	Principe	139
6.3.2	Commutation du domaine de codage	141
6.3.3	Posttraitement fréquentiel	147
6.3.4	Performances avec optimisations	148
6.4	Conclusion	149
7	Codage Hiérarchique avec Extension de Bande	151
7.1	Introduction	151
7.2	Structure hiérarchique du codeur	153

7.2.1	Principe	153
7.2.2	Codage paramétrique de la bande haute	158
7.2.3	Codage par transformée	161
7.2.4	Multiplexage codage/estimation	167
7.3	Améliorations et compromis	167
7.3.1	Ordonnancement des vecteurs codés	167
7.3.2	Allocation binaire fixe entre la bande basse et la bande haute	168
7.3.3	Réduction de l'étalement spectral	169
7.3.4	Postfiltrages de la synthèse	171
7.3.5	Autres améliorations potentielles	174
7.4	Tests subjectifs	174
7.5	Conclusion	175
CONCLUSION		179
A	Modélisation par un processus de Markov	185
BIBLIOGRAPHIE		188

LISTE DES FIGURES

1.1	Comparaison de l'approche conventionnelle (a), avec l'approche hiérarchique (b) selon les conditions du réseau.	6
1.2	Illustration d'une transmission multipoint de données codées hiérarchiquement.	6
1.3	Configuration de notre codeur hiérarchique.	8
2.1	Schéma fondamental d'un système de communication.	11
2.2	Séparation codage source-canal.	12
2.3	Mécanisme de production de la parole.	15
2.4	Modèle source-filtre de production de la parole.	16
2.5	Représentations d'un segment voisé de parole.	18
2.6	Système ADPCM.	21
2.7	Principe de codage par analyse par synthèse.	23
2.8	Principe du codage CELP.	23
2.9	Spectres du filtre de synthèse et du filtre perceptuel.	24
2.10	Exemple d'un banc de filtres auditif non uniforme.	27
2.11	Modèle de masquage fréquentiel par une fonction d'étalement.	28
2.12	Schéma de principe d'un codeur par transformée.	30
2.13	Représentation d'une transformation modulée par un banc de filtres.	33
2.14	Chaîne de codage par transformée.	36
2.15	Chaîne de codage par transformée.	37
2.16	Principe du codeur TCX.	38
3.1	Codeur/Décodeur (Codec) à deux couches.	41
3.2	Compromis entre la qualité du codeur de base et celle des couches d'amélioration.	42
3.3	Codage CELP imbriqué avec k dictionnaires innovateurs.	44
3.4	Association d'un codage parole avec un codeur par transformée.	44
3.5	Extension d'un codeur bande étroite en bande élargie par une décomposition en deux sous-bandes.	46

3.6	Extension en pleine bande d'un codeur bande étroite.	46
3.7	Principe de l'extension de bande artificielle ou paramétrique.	48
4.1	Comparaison du spectre d'amplitude d'un signal d'orgue original avec celui de sa synthèse par l'AMR-WB.	53
4.2	Erreurs de codage avec une excitation uniquement adaptative et une excitation adaptative et innovatrice.	54
4.3	Diagramme de haut niveau du système proposé.	55
4.4	Le post-traitement est appliqué à la sortie \hat{x}_{base} du décodeur parole.	55
4.5	Génération du masque M et du sous-flux d'amélioration au codeur.	56
4.6	Signal original d'orgue et sa synthèse \hat{x}_{enh} après le post-traitement fréquentiel. .	57
4.7	Illustration du traitement pour une composante fréquentielle : (a) pas de mise à zéro, (b) mise à zéro.	58
4.8	Répercussions du post-traitement sur l'énergie de la synthèse en fonction du nombre de composantes mise à zéro pour chacune des trames d'un signal audio constitué de paroles et de musiques.	59
4.9	Entropie du signal masque M pour l'alphabet $B = \{0, 1\}$	61
4.10	Projection des statistiques du masque M de différentes sources dans le plan des équi-contours de l'entropie du modèle de Markov.	64
4.11	Remise en forme du masque par élimination des valeurs isolées.	66
4.12	Améliorations des statistiques de M vers M' pour différentes sources dans le plan des isocontours de l'entropie du modèle de Markov.	67
4.13	Remise en forme par plages du masque.	69
4.14	Améliorations des statistiques de M vers M'' pour différentes sources dans le plan des isocontours de l'entropie du modèle de Markov.	69
4.15	Décimation du masque pour une réduction du débit.	71
4.16	Interpolation de facteur 4 du masque.	71
4.17	Codage multi-modal du masque.	73
4.18	Représentation de la discrimination des signaux lors du codage multi-modal par des zones du plan des iso-contours de l'entropie du modèle de Markov.	74
4.19	Résultats du test MUSHRA : notes moyennes avec l'intervalle de confiance à 95%. .	78
5.1	Illustration en dimension 2 du gain G_{g1} (ici égal à 0.17 dB) pour une source uniforme. .	83

5.2	Illustration en dimension 2 du gain G_{g2} pour une source gaussienne sans mémoire ($L = 16$).	84
5.3	Illustration en dimension 2 du gain G_{g3} pour un processus autorégressif ($L = 16$).	85
5.4	Exemple de réseau régulier de points	87
5.5	Réseaux \mathbb{Z}^2 , D_2 et A_2 avec leurs régions de Voronoï	88
5.6	Exemples de régions de support pour le réseau A_2	90
5.7	Exemple d'une quantification de x par un réseau régulier de points (a) avec saturation et (b) sans saturation.	91
5.8	Différentes graduations et ordres de transmission du signal quantifié : (a) transmission graduelle des vecteurs du signal, (b) transmission graduelle des vecteurs du signal ainsi que de leurs amplitudes.	94
5.9	Quantification LVQ multi-débit	95
5.10	Illustration d'une extension de Voronoï d'ordre 1 du dictionnaire C	98
5.11	Exemple d'extension de Voronoï pour une constellation C du réseau A_2 . Le point noir indique le centroïde de la constellation.	99
5.12	Décomposition d'un vecteur y de A_2 par l'extension de Voronoï d'ordre 2.	99
5.13	Illustration d'une extension de Voronoï graduelle dans le réseau A_2	102
5.14	Extension de Voronoï graduelle pour une constellation C du réseau A_2 : illustration de la dérive du centroïde (point noir) du dictionnaire.	103
5.15	Décomposition d'un vecteur y de A_2 par l'extension de Voronoï graduelle d'ordre 2.104	
5.16	Codes de Voronoï dans A_2 répondant à la condition de l'équation 5.32 ($a = 0.6$).	105
5.17	Nouvelle extension de Voronoï graduelle pour une constellation C du réseau A_2 . Le point noir indique le centroïde de la constellation.	106
5.18	Décomposition d'un vecteur y de A_2 par la nouvelle extension de Voronoï graduelle d'ordre 2.	107
5.19	Décodage par raffinements successifs d'une extension de Voronoï graduelle dans le réseau A_2	110
5.20	Allocation des ressources selon le remplissage inverse des eaux.	112
5.21	Contrôle par un gain global du débit de sortie du codage LVQ multidébit.	113
5.22	Contrôle par troncature du débit de sortie du codage LVQ multi-débit	114
5.23	Exemple d'allocation par troncature avec détermination de l'ordre de transmission. 115	
5.24	Organisation du train binaire généré pour une transmission graduelle des amplitudes des vecteurs codés par une quantification par extension de Voronoï graduelle. 117	

5.25 Système de quantification LVQ multi-débit dans RE_8 avec décodage à débit variable.	118
5.26 Performances de l'extension de Voronoï graduelle pour le cas gaussien avec un dictionnaire de base nul.	121
5.27 Performances de l'extension de Voronoï graduelle pour le cas gaussien avec les dictionnaires de base de l'AMR-WB+.	122
5.28 Performances de différentes stratégies de quantifications à raffinements successifs de la cible TCX dans RE_8 comparées à une quantification à débit fixe.	124
6.1 Schéma de principe du codage hiérarchique.	128
6.2 Schéma bloc du codeur hiérarchique.	129
6.3 Schéma bloc du décodeur hiérarchique.	129
6.4 Illustration de la reconstruction parfaite à l'aide de deux fenêtres KBD au carré $h^2(n)$ successives avec $\alpha = 4$.	131
6.5 Illustration du processus de masquage du signal différence.	133
6.6 Illustration du blanchiment du signal de différence.	134
6.7 Densité de probabilité de la cible TCX normalisée avec et sans masquage (les valeurs à -3.5 et 3.5 représentent les probabilités cumulées pour les valeurs < -3.5 et > 3.5 respectivement).	136
6.8 Fréquence d'utilisation des dictionnaires de la LVQ.	136
6.9 Numéro moyen de dictionnaire utilisé par vecteur de dimension 8 selon les bandes de fréquence.	137
6.10 segSNR du codeur hiérarchique comparé à l'AMR-WB.	137
6.11 Résultats du test MUSHRA évaluant notre codeur hiérarchique.	139
6.12 Schéma bloc du codeur hiérarchique optimisé	140
6.13 Schéma bloc du décodeur hiérarchique optimisé	140
6.14 Fenêtrage adaptatif entre une résolution temporelle normale et une résolution huit fois plus grande.	142
6.15 Cas particulier du fenêtrage adaptatif où la résolution temporelle devient maximale. Les fenêtres en traits continus sont codées dans le domaine fréquentiel alors que la fenêtre en pointillé est codée dans le domaine temporel.	144
6.16 Illustration du repliement temporel lors d'une MDCT.	145
6.17 Illustration de la reconstruction parfaire à l'aide de la TDAC.	145
6.18 Repliement temporel dans le domaine temporel (TDAC temporelle) permettant la comptabilité avec la MDCT.	146

6.19	Commutation du domaine de codage avec des fenêtres de transition directement dans le domaine temporel.	147
6.20	Résultats du test MUSHRA évaluant les optimisations approtées au codeur hiérarchique.	149
6.21	Résultats globaux sur l'ensemble des signaux.	150
7.1	Décodage d'une trame à un débit de décodage donné par le codage hiérarchique avec extension de bande.	153
7.2	Schéma de principe du codage hiérarchique avec extension de bande.	154
7.3	Caractéristique des filtres QMF d'ordre 63.	155
7.4	Schéma bloc du codeur hiérarchique avec extension de bande.	156
7.5	Schéma bloc du décodeur hiérarchique avec extension de bande.	156
7.6	Codage paramétrique de l'excitation de la bande haute.	159
7.7	Calcul des gains de correction servant à la prédition des gains en énergie de l'excitation de la bande haute.	159
7.8	Codage de l'enveloppe spectrale par une analyse LP.	160
7.9	Illustration de la génération de la cible à quantifier.	162
7.10	Densité de probabilité des coefficients de cible de la MDCT après normalisation, et sa modélisation par plusieurs gaussiennes généralisées de différents γ (les valeurs à -3.5 et 3.5 représentent les probabilités cumulées pour les valeurs <-3.5 et >3.5 respectivement).	164
7.11	Fréquence d'utilisation des dictionnaires de la quantification.	165
7.12	Numéro du dictionnaire moyen utilisé par chaque vecteur de dimension 8.	166
7.13	Erreur de codage avant et après quantification de la cible.	166
7.14	Multiplexage fréquentiel entre le signal estimé et codé.	167
7.15	Performances WPESQ du codage hiérarchique pour différents types d'allocation.	170
7.16	Prédition linéaire fréquentielle pour mettre en forme temporellement l'erreur du codage par transformée.	171
7.17	Codage sans mise en forme temporelle.	172
7.18	Codage avec mise en forme temporelle.	172
7.19	Enveloppe temporelle issue du filtre de synthèse de la prédition fréquentielle.	172
7.20	Résultats du test Mushra avec intervalles de confiance à 95%.	176
A.1	Processus de Markov du premier ordre.	185

A.2	Distribution mesurée versus distribution issue du processus de Markov.	186
A.3	Iso-contours de l'entropie $H(Z)$ en fonction des espérances des plages de 1s et de 0s.	187

LISTE DES TABLEAUX

4.1	SegSNR moyen de la synthèse audio avant et après Posttraitement (dB).	60
4.2	Entropie du processus binaire du signal M pour différents signaux.	61
4.3	Entropie du signal M obtenu par un codage par plages.	62
4.4	Débits du codage par plages avec des codes de Huffman pour le signal M	63
4.5	Débits du codage RLC du masque M avec limitation du débit à 1 bit/échantillon.	63
4.6	Débits pour le codage par plages du masque remis en forme M'	66
4.7	Débits pour le codage par plages du masque remis en forme M''	68
4.8	Débits pour le codage multimodal du masque avec $F = 2$	73
4.9	Débits pour le codage multi-modal du masque avec $F = 4$	74
5.1	Performances des réseaux importants.	89
5.2	Spécifications de Q_0, Q_2, Q_3 et Q_4 dans RE_8	101
6.1	Débits moyens alloués aux différentes couches (kbit/s).	148
7.1	Allocation binaire (bits/trame) pour la quantification vectorielle des LSFs.	160
7.2	Performances du codage de l'enveloppe spectrale.	161
7.3	Allocation binaire pour le codage des gains en énergie de l'excitation de la bande haute.	161

CHAPITRE 1

Introduction

Le travail de cette thèse porte sur le codage audio hiérarchique à faibles débits. Nous nous intéressons à coder l'information contenue dans les signaux audio, comme la parole ou la musique, afin d'obtenir une représentation compacte de l'information. Plus précisément, on souhaite étudier les diverses façons de représenter le signal de manière hiérarchisée afin qu'il puisse être transmis sur des réseaux de communication à des débits oscillants entre 8 kbit/s et 32 kbit/s. Nos objectifs sont doubles. D'une part, on souhaite obtenir une bonne qualité de restitution pour de la parole même au débit le plus bas (8 kbit/s), et de pouvoir traiter convenablement des signaux audio plus généraux pour les débits supérieurs (jusqu'à 32 kbit/s). D'autre part, on impose que l'information codée sous forme hiérarchique soit transmise à l'aide d'un train binaire encastré. Un tel train binaire est composé de plusieurs sous-flux dont le premier, le sous-flux de base, est décodable indépendamment des autres et est associé à la qualité minimale du codage. Les sous-flux suivants, dits d'amélioration, raffinent graduellement la qualité de la restitution audio. Un train binaire encastré peut s'adapter facilement à la source à coder ainsi qu'aux conditions variables des liaisons de communication. Le travail de cette thèse a été motivé par le contexte actuel des télécommunications et de sa future évolution.

1.1 L'évolution des télécommunications

Les systèmes de communication ont évolué grandement ces dernières années. La téléphonie mobile est devenue omniprésente dans les communications modernes. Après la première génération analogique de téléphonie sans fil, c'est en 1991 avec l'apparition de la deuxième génération numérique (2G) que la téléphonie mobile s'est démocratisée. Le trafic, constitué essentiellement au début de communications parlées, s'est enrichi de contenus multimédias. Cette intégration de services a amené en 2002 à la définition d'une troisième génération (3G). Grâce à sa vitesse accrue de transmission de données, cette nouvelle génération de réseaux (UMTS et CDMA2000) ouvre la porte à des applications et à des services nouveaux. Elle permet en particulier de transférer dans des temps relativement courts des contenus multimédias tels que des images, du son ou bien de la

vidéo. La qualité des communications parlées s'en trouve aussi améliorée avec l'introduction du codage large bande (50-7000 Hz) de la parole étendant ainsi la bande téléphonique habituellement transmise (300-3400 Hz).

Par ailleurs, la démocratisation d'Internet haut débit grand public a donné à chacun accès à des contenus multimédias grâce à des technologies telles l'ADSL. Elle permet entre autres l'utilisation d'applications comme la voix sur IP (VoIP) pour des communications interactives et temps réel, réservées jusqu'alors à des systèmes dédiés.

Cette convergence des applications fait que le marché s'oriente vers une intégration des accès qu'ils soient radios ou filaires. La nouvelle génération de téléphonie (4G) sera vraisemblablement bien plus qu'une évolution de l'ancienne génération, mais une solution globale pour les communications radios. Certains parlent alors plutôt de *beyond 3G*. La tendance est donc à l'utilisation d'une plateforme de communication basée sur l'unique protocole IP (*all-IP*) permettant ainsi la coopération entre différentes technologies d'accès au réseau (GSM, UMTS, WiFi, Bluetooth, Ethernet...). Cette plateforme assurera alors aussi bien les services de téléphonie traditionnelle que le transport de données.

Fournir une communication téléphonique fiable sur des réseaux IP prévus initialement à la transmission de données n'est pas si simple à réaliser. De nombreux travaux actuels [1] se penchent sur l'amélioration de la qualité de service (*Quality of Service*, QoS) des réseaux IP pour assurer une communication temps réel interactive et fiable. La définition de protocoles de communication avec une QoS garantie est alors essentielle pour le bon fonctionnement de certaines applications.

En amont des liaisons de communication, le codage des données est aussi un élément essentiel au système de transmission des données. En effet, le codage permet de compresser les données transmises et peut ainsi assurer une ou plusieurs communications fiables sur le même lien. Le codage classique assure généralement une communication pour un débit constant garanti. Or, comme on vient de le voir, la nature des réseaux et des terminaux est de plus en plus hétérogène, ce qui conduit à des conditions de transport variables dans le temps et dans l'espace. Il est donc nécessaire de pouvoir adapter le codage d'une même source aux capacités fluctuantes des liens d'un réseau pour satisfaire ses différents destinataires.

De surcroît, avec l'émergence de l'intégration des services, de nouvelles applications transmettant des données audio numériques apparaissent aussi. Il s'agit entre autres d'audio/vidéo-conférence,

de lecture audio/vidéo en transit (*audio streaming*), de diffusion audio/vidéo ou bien de messagerie audio. Ces applications ont besoin, comme les applications téléphoniques classiques, de transmettre essentiellement de la parole, mais ont aussi la nécessité de transmettre du contenu sonore plus varié comme de la musique, du bruit ambiant ou une mixture parole-musique. Le codage devra donc aussi pouvoir s'adapter au contenu de la source.

1.2 Le contexte normatif en codage audio

Depuis la fin des années 60 et la numérisation des communications téléphoniques, la qualité de la parole transmise n'a presque pas évolué, et est restée équivalente à celle de la téléphonie analogique. Le signal parole transmis est principalement monaural et à bande étroite, c'est à dire coupé en dehors de la bande 300-3400 Hz. Les premiers codeurs basés sur la technologie de Modulation d'Impulsions et Codage (*Pulse-Code Modulation*, PCM), ont des débits élevés comme la norme G.711 de l'ITU-T (*International Telecommunication Union*) à 64 kbit/s [2]. Actuellement les codeurs bande étroite faisant état de l'art se basent sur la technologie CELP (*Code Excited Linear Prediction*) introduite dans les années 80 par Atal et Schroeder [3]. Ils ont des débits de l'ordre de 8 kbit/s. Dans cette famille de codeurs, on peut citer la recommandation AMR (*Adaptive Multi Rate*) [4] du 3GPP (*Generation Partnership Project*) pour la téléphonie mobile de 3^e génération ainsi que la norme G.729 [5] de l'ITU-T pour les applications de téléphonie sur IP.

À la fin des années 80 avec la normalisation par l'ITU-T du G.722 [6], la bande codée de la parole a été élargie à 7 kHz afin d'améliorer la qualité de la communication. Mais ce n'est seulement dix ans plus tard que des codeurs CELP large bande ont été normalisés. Dans cette catégorie, la recommandation du 3GPP pour la transmission de la parole large bande en téléphonie mobile est l'AMR-WB (*AMR-Wide Band*) [7], qui a aussi été adoptée pour des applications sur IP par l'ITU-T sous le nom de G.722.2. L'AMR-WB, comme l'AMR, dispose de plusieurs modes de codage générant des débits différents. Le choix du mode doit alors être contrôlé par les conditions du réseau (*network-controlled mode switching*). Le consortium 3GPP2 pour la 3^e génération de mobile sur les réseaux CDMA2000 a récemment normalisé le VMR-WB (*Variable Rate Multi-Mode WB*) [8]. Contrairement à l'AMR-WB, le débit de sortie du VMR-WB est aussi contrôlé par la source.

L'ITU-T se penche actuellement sur le codage à débit variable à travers la question Q9/16. Deux approches sont étudiées, l'une utilisant un contrôle par la source et l'autre le codage hiérarchique. Le codeur hiérarchique G.729EV est dans ce cadre en cours de normalisation [9]. L'approbation des termes de référence ainsi que la phase de concours et de sélection se sont déroulées pendant le travail de cette thèse. Le G.729EV est une extension hiérarchique du codeur bande étroite G.729 à 8 kbit/s. Pour un premier incrément de débit, il permet d'améliorer la qualité en bande étroite, et, à partir de 14 kbit/s, d'étendre la bande codée à 7 kHz. Ensuite, par incrément d'au plus 2 kbit/s, il améliore la synthèse large bande jusqu'à atteindre 32 kbit/s. À ce débit, il peut aussi bien traiter de la parole que de la musique. Les applications visées sont essentiellement la transmission haute qualité de la voix (voire de la musique) sur des réseaux par paquets (ATM, IP).

Historiquement, le codage de la parole et le codage générique audio ont été étudiés séparément et se basent ainsi sur des technologies différentes. Les codeurs de parole, fondés sur un modèle de production de la parole, bien qu'étant très performants pour cette famille spécifique de signaux, sont inefficaces pour des signaux plus généraux. Les codeurs audio génériques, appelés aussi par extension codeurs de musique, ont quant à eux des débits nominaux plus élevés. À faibles débits, leurs performances sont très limitées, surtout pour des signaux non stationnaires comme la parole. Ces deux paradigmes de codage ont ainsi leurs propres applications. Pour les applications de diffusion ou du stockage n'ayant pas de fortes contraintes sur le débit et le délai, les codeurs audio génériques tels MP3 [10] ou AAC [11] sont généralement utilisés.

Récemment, de nouveaux codeurs ayant des débits intermédiaires apparaissent permettant ainsi de pouvoir manipuler aussi bien de la parole que de la musique. En effet, le 3GPP vient d'adopter en 2004 deux codeurs audio, l'AMR-WB+ [12] et l'Enhanced aacPlus [13], pour les services multimédias. Les deux codeurs proviennent respectivement du monde du codage de la parole et du codage générique audio. Ils sont principalement destinés à des applications non interactives transmettant de la parole aussi bien que de la musique à des débits allant de 10 kbit/s à 48 kbit/s. Ils codent tous deux le signal sur une bande allant jusqu'à 20 kHz et permettent la reproduction d'une image stéréophonique à l'aide d'un codage paramétrique. Néanmoins, garantir une bonne qualité de restitution à faible débit quel que soit le signal audio à coder reste encore un problème ouvert qui n'admet pour le moment aucune solution véritablement générale. Cette problématique

connue sous le nom de codage audio universel à bas débit est de nos jours un des grands défis du codage audio.

1.3 Les avantages du codage audio hiérarchique

Pour s'adapter aux diverses conditions de transmission et pouvoir aussi s'adapter à la source, une solution consiste à disposer de plusieurs techniques de codage ayant entre autres des débits différents. Le codage le plus adapté à la source et répondant aux contraintes du réseau sera alors utilisé pour la transmission. Cette solution simple, mais très lourde et peu économique, est dans de nombreux cas inapplicable.

L'utilisation d'un codage hiérarchique est alors une solution beaucoup plus élégante. Les décodeurs hiérarchiques ont la particularité de permettre un décodage incrémental, c.-à-d. formé par ajouts successifs de descriptions afin d'améliorer la restitution du signal. Il s'agit d'algorithmes de compression audio capables de représenter l'information de façon hiérarchique par un train binaire composé de plusieurs sous-flux binaires. Le premier sous-flux appelé sous-flux de base est décodable indépendamment des autres et permet la restitution du signal à un niveau de qualité minimal. Les prises en compte successives par le décodeur des sous-flux suivants, appelés sous-flux d'amélioration, permettent d'obtenir des incrémentations de qualité de la restitution. Ces sous-flux, emboîtés dans un même train binaire (on parle alors de train binaire encastré) ou bien diffusés sur des canaux distincts, permettent une transmission puis un décodage à débit variable sans contrôle du codeur que ce soit par la source ou bien par le réseau. La Figure 1.1 compare l'approche conventionnelle avec l'approche hiérarchique.

En particulier, le codage hiérarchique est une solution très appropriée pour la transmission sur des réseaux hétérogènes. La sélection par les noeuds du réseau (ou par le récepteur) d'un sous-ensemble des sous-flux générés par le codage hiérarchique, permet d'ajuster le débit de transmission ou de décodage à la capacité du canal ou du récepteur respectivement. Il est tout aussi adapté pour des transmissions point à point que multipoint. La Figure 1.2 donne l'exemple d'un scénario de transmission multi-point d'un train binaire encastré. Le débit localement disponible au niveau des décodeurs varie d'un récepteur à l'autre.

Du fait des contraintes inhérentes à la hiérarchisation, le codage hiérarchique est encore un codage peu utilisé et déployé. Cependant, il fait l'objet d'un intérêt grandissant pour les raisons évoquées

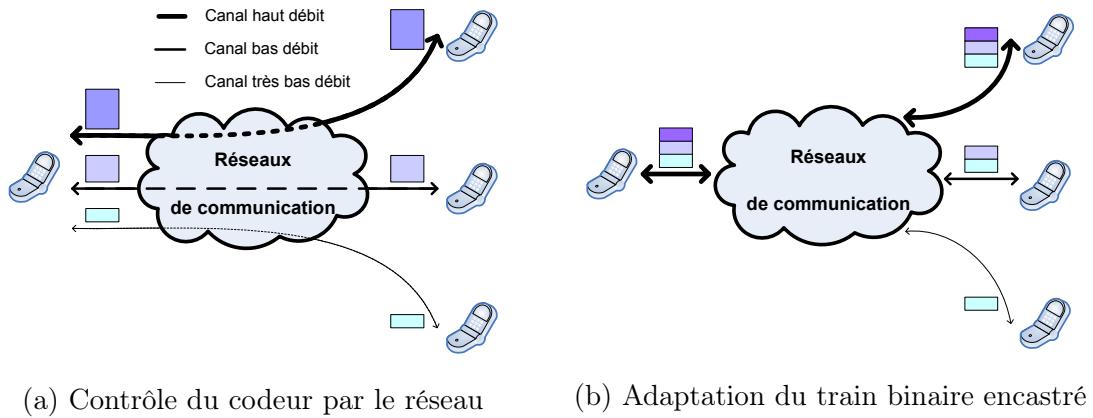


Figure 1.1 Comparaison de l'approche conventionnelle (a), avec l'approche hiérarchique (b) selon les conditions du réseau.

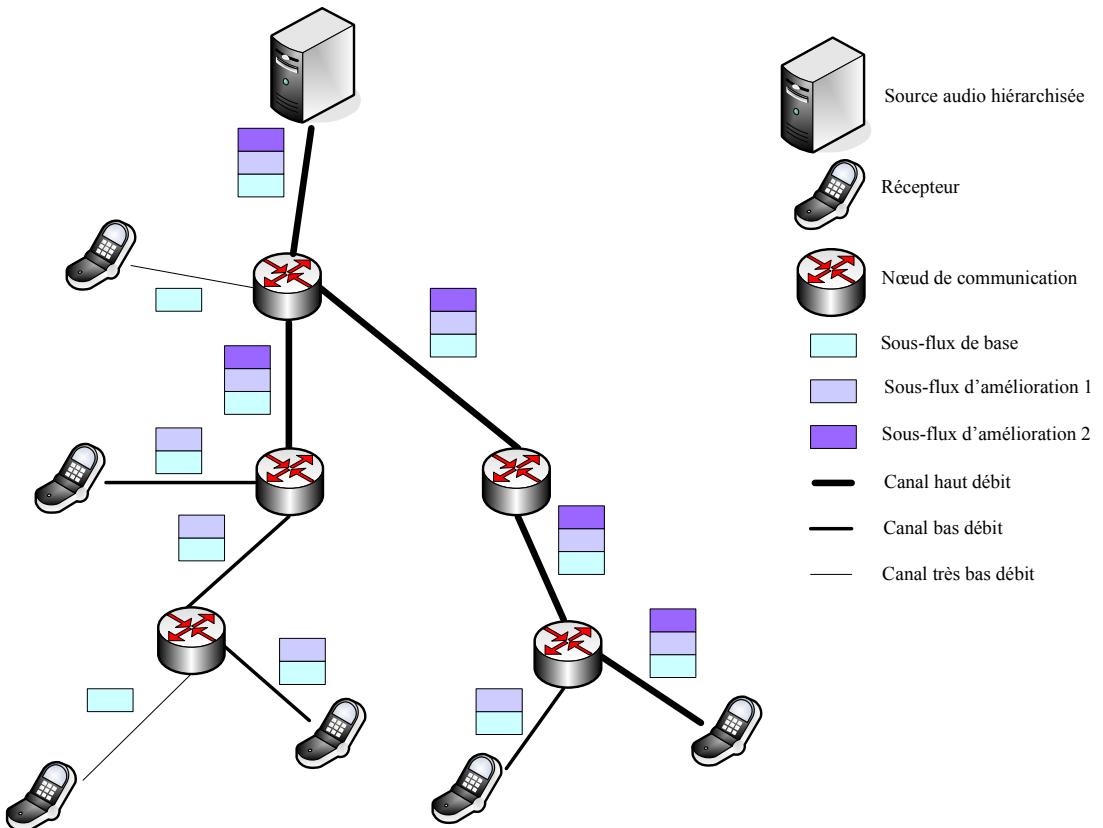


Figure 1.2 Illustration d'une transmission multipoint de données codées hiérarchiquement.

ci-dessus. De plus, il pourrait apporter des attributs au codage audio encore non couverts par les normes actuelles. Un codage hiérarchique pourrait par exemple couvrir par un unique codage une plage de fréquences permettant d'aller de la qualité téléphonique à la qualité haute-fidélité. Il est aussi envisageable de faire un codage hiérarchique multicanal permettant de passer d'une restitution monaurale à une ambiance stéréophonique, voire encore plus réaliste. De plus, le codage hiérarchique peut aussi s'adapter à la source. Sa structure imbriquée permet l'association de techniques de codage venant des deux paradigmes de codage, à savoir le codage de la parole et le codage audio générique. Il pourrait alors être une solution pour le codage audio universel à bas débit.

1.4 Positionnement du projet

Le cadre de notre travail peut se résumer par la Figure 1.3. Elle représente la vision à partir de laquelle on a entrepris nos travaux de recherche sur le codage hiérarchique. Tout d'abord, on part d'une couche de base générant le sous-flux de base du train binaire. Pour répondre au problème du codage universel, on choisit d'utiliser une technique propre au codage de la parole comme couche de base. On assure alors une bonne qualité de compression lors du traitement de la parole même pour le débit de décodage le plus faible. Les couches d'amélioration du SNR (*Signal to Noise Ratio*), auront pour objectif de réduire le bruit de codage introduit par le codeur parole. Pour réduire le budget de bits nécessaire, il faudra se concentrer essentiellement sur le bruit perceptible. Il s'agira entre autres d'améliorer les performances du codeur de base lors du traitement des signaux autres que la parole. Les couches d'amélioration de la largeur de bande permettront quant à elles d'étendre la bande originale codée. La largeur de bande de la synthèse est un critère perceptuel de qualité très important. Pour finir, les couches d'amélioration du nombre de canaux, permettront de passer d'un codage monaural à un codage multicanal. Cette dernière fonctionnalité n'a pas été étudiée lors de cette thèse. Elle pourra faire l'objet de travaux futurs. Il est à noter que le volume des couches de la Figure 1.3 n'est pas forcément fonction du débit qui leur est alloué.

Au cours de la thèse, on a étudié deux structures de codage hiérarchique. La première se base sur le codeur large bande AMR-WB à 12.65 kbit/s. L'objectif est de lui ajouter les fonctions des couches d'amélioration du SNR. On veut ainsi améliorer la qualité de la synthèse large bande essentiellement lorsque le signal d'entrée n'est pas de la parole. On vise pour ce codage des

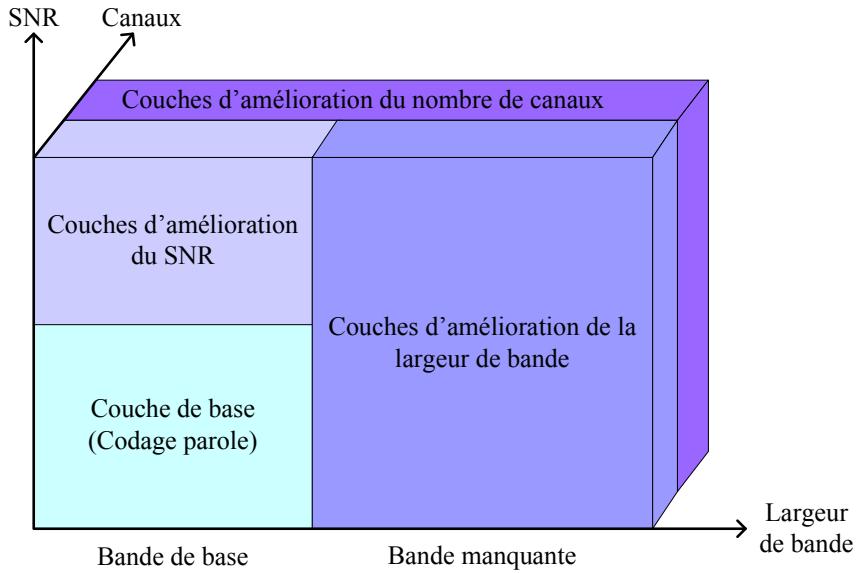


Figure 1.3 Configuration de notre codeur hiérarchique.

débits allant de 12.65 à 24 kbit/s. Dans cette gamme, on peut toujours parler de bas débits et comparer les performances à débit équivalent avec celles de codeurs large bande existants. La deuxième structure étudiée est calquée sur les exigences définies par l'ITU-T pour la future norme G.729EV. Il est alors question d'ajouter à la norme G.729 les fonctions des couches d'amélioration du SNR et de la largeur de bande. Les débits vont de 8 kbit/s à 32 kbit/s. Pour ces deux structures, on s'est aussi contraint à avoir des délais algorithmiques en dessous de 60 ms pour permettre une interactivité des communications. Bien sûr, les solutions ont été développées dans le souci d'obtenir une complexité raisonnable, pour que les solutions puissent être intégrées à des terminaux portatifs, comme les téléphones mobiles.

1.5 Plan de la thèse

Ce document contient 7 chapitres, dont 4 faisant l'objet de contributions.

Le deuxième chapitre s'attache à faire une présentation générale du codage audio. On s'intéresse en particulier au codage de la parole et au codage par transformée. On finit le chapitre en soulevant le problème du codage universel à bas débit. Le chapitre suivant passe quant à lui en revue les différentes techniques de codage hiérarchique à bas débits existantes dans la littérature. On étudie plus précisément les solutions à base d'un codeur de parole.

Le quatrième chapitre introduit un nouveau posttraitemet pour les codeurs de parole. Ce posttraitemet nécessite la transmission d'informations supplémentaires qui permettront au niveau du décodeur d'améliorer la qualité de la synthèse du codeur de parole surtout lorsqu'il s'agit d'un signal musical. Ce posttraitemet s'inscrit dans notre représentation du codage hiérarchique de la Figure 1.3 comme une couche d'amélioration du SNR. La solution a été testée dans le cas de l'AMR-WB et se montre d'une grande efficacité particulièrement pour les signaux pénalisés par la structure spécifique des codeurs de parole.

Au cinquième chapitre, on introduit une quantification algébrique à raffinements successifs. Après une brève introduction à la quantification vectorielle, on détaillle en particulier une quantification par contrainte intéressante en codage audio, la quantification par réseau régulier de points. Cette quantification algébrique a une très faible complexité et nécessite peu d'espace de stockage pour ses dictionnaires. De plus, elle peut s'adapter à différentes sources et à différents débits selon la définition de ses dictionnaires. En utilisant les propriétés remarquables des réseaux réguliers de points, on définit une nouvelle méthode pour définir les dictionnaires : l'extension de Voronoï graduelle. Les vecteurs codés par ces dictionnaires sont alors décodables graduellement. En plus de cette fonctionnalité importante pour le codage hiérarchique, la nouvelle quantification présente des performances intéressantes dans le cas d'une source gaussienne et pour les coefficients fréquentiels d'une source audio.

Le sixième chapitre présente une structure hiérarchique à 24 kbit/s à base du codeur de parole large bande AMR-WB à 12.65 kbit/s. Elle consiste en une superposition d'un codage par transformée au-dessus du codeur de parole. Le codage par transformée joue le rôle de la couche d'amélioration du SNR de la Figure 1.3. L'amélioration apportée est plus fine et précise que le posttraitemet du chapitre 4, mais aussi plus complexe et plus gourmande en bits. Elle utilise la quantification algébrique à raffinements successifs du chapitre précédent pour décrire les coefficients transformés. Des précautions particulières sont prises pour que la couche d'amélioration prenne en compte seulement les composantes audibles de l'erreur de codage de la couche de base. On obtient ainsi un codeur mixte parole/musique ayant des performances très stables indépendamment du signal d'entrée. Néanmoins, pour améliorer les performances pour certains types de signaux, on propose l'ajout d'optimisations perceptuelles. La première est l'insertion du posttraitemet fréquentiel du chapitre 4 afin d'améliorer la qualité pour les sons plutôt musicaux.

La seconde est un commutateur adaptatif du domaine de codage qui permet à la couche d'amélioration de passer d'un domaine fréquentiel à un domaine temporel lors de fortes transitoires.

Dans le septième et dernier chapitre, on présente un codage hiérarchique à 32 kbit/s à base du codeur de parole bande étroite G.729 à 8 kbit/s. Cette fois-ci, les couches d'améliorations permettent d'affiner la description en bande codée par le G.729 mais aussi de l'étendre pour obtenir une synthèse large bande. Les couches d'amélioration améliorent le SNR ainsi que la largeur de bande comme illustré à la Figure 1.3. Une décomposition en sous-bandes permet de dissocier la bande de base de la bande manquante. La bande de base est codée par le codeur de parole suivi d'un codeur par transformée comme au chapitre 6. Dans la bande manquante, un codage paramétrique est utilisé en parallèle à un codage par transformée. Un multiplexage fréquentiel est réalisé au décodeur en fonction du débit de décodage pour associer les coefficients estimés du codage paramétrique avec les coefficients codés. Une analyse du codeur hiérarchique permet de proposer des améliorations et de discuter de divers compromis selon l'application visée.

Le document se termine par une conclusion sur le travail présenté. Il revient sur les résultats importants, et sur les contributions majeures. Enfin, il donne les perspectives envisagées du projet ainsi que d'autres pistes de recherche intéressantes à suivre pour le codage hiérarchique.

CHAPITRE 2

Le Codage Audio

Le codage du signal est un traitement grandement utilisé dans les communications modernes. Ce chapitre a pour objectif d'introduire brièvement les techniques du codage audio pour positionner le projet de recherche de la thèse. Des descriptions complètes et exhaustives peuvent être trouvées dans la littérature [14, 15].

2.1 Codage de source

Les motivations du codage de source se trouvent dans la transmission de l'information d'un signal sur des médias de communication. Ces médias de communication ne sont en général pas parfaits et sont sujets à deux types de contraintes majeures : limitation de la bande passante et présence de perturbations aléatoires désignées par le terme générique de bruit. L'étude d'un tel système de communication a amené C.E. Shannon à introduire la théorie de l'information. La Figure 2.1 représente le système de communication désigné sous le nom *paradigme de Shannon*.

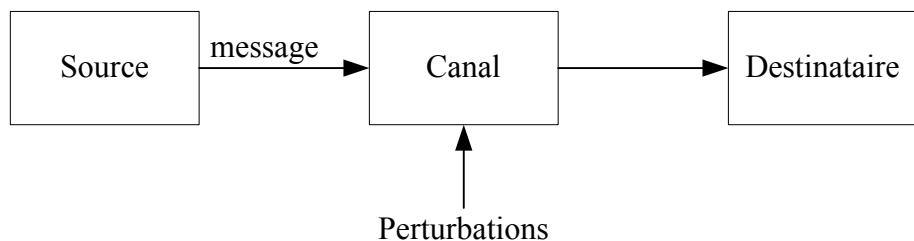


Figure 2.1 Schéma fondamental d'un système de communication.

Un système de transmission assure, par définition, la transmission d'une certaine information depuis la source jusqu'à un destinataire éventuellement distant. L'information transmise est aux yeux du destinataire, intrinsèquement aléatoire : si elle était parfaitement connue de celui-ci, la transmission deviendrait sans objet. Les perturbations de nature aléatoire ont pour effet de créer une différence entre le message émis et celui qui est reçu.

Le résultat fondamental de la théorie de l'information établie par C.E. Shannon [16], est qu'il est possible de réaliser une transmission d'information exempte d'erreur, malgré l'existence de bruit sur le canal. Cela suppose une représentation appropriée de l'information (on utilisera dans la suite le terme de codage) et impose des contraintes sur le débit de l'information transmise, qui dépendent des caractéristiques du canal. Un système de communication "idéal" peut être schématisé par la Figure 2.2. Ce schéma s'appuie sur le théorème de séparation énoncé par Shannon :

Théorème source-canal 2.1.1 *Si le débit minimum réalisable d'une source pour une distorsion minimale donnée, est au-dessous de la capacité du canal, alors la source peut être transmise convenablement au travers du canal, tout en s'approchant au plus près de la distorsion minimale imposée, si on lui associe un codage de canal ajoutant une redondance suffisamment grande. Au contraire, si le débit minimum de la source excède la capacité du canal, alors une transmission sûre ne peut pas être effectuée.*

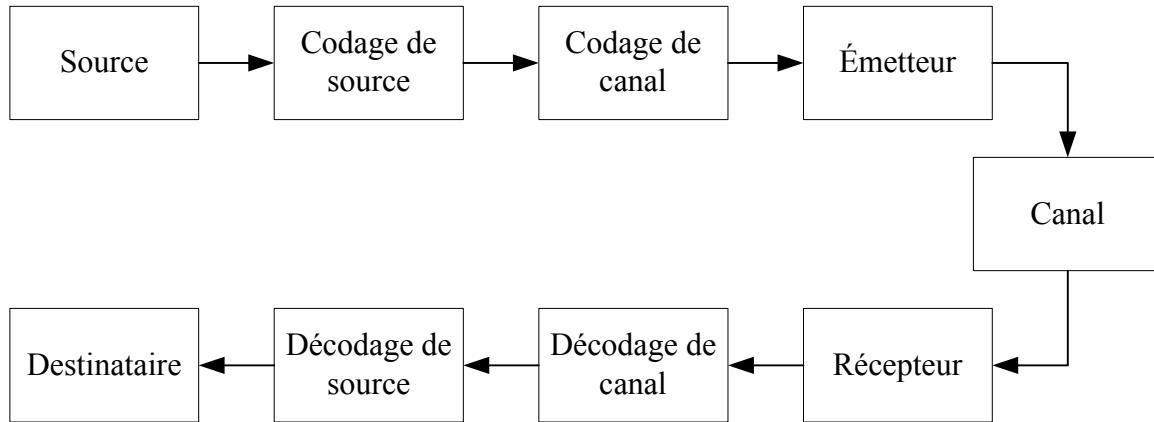


Figure 2.2 Séparation codage source-canal.

Ce théorème est la combinaison de deux principes élaborés par Shannon : le théorème du codage de canal et le théorème du codage de source. Shannon prétend qu'on peut tendre vers une limite optimale de débit avec un système de codage source et de codage canal séparés. Le théorème du codage de canal dit qu'une transmission est fiable dans un canal bruité si le débit binaire à la sortie du codage de canal R_C est inférieur à la capacité C du canal. La capacité est par définition la quantité d'information maximale qu'il peut transporter. Il existe alors un code permettant de transmettre l'information avec une probabilité d'erreur arbitrairement petite : c'est le codage

de canal. Par contre le théorème du codage de source affirme qu'il existe un débit binaire R_s vers lequel on peut tendre mais en deçà duquel on ne peut comprimer davantage une source d'information. Dans le cas d'un codage "sans perte", cette limite est, en théorie l'*entropie* de la source. Plus généralement, pour un codage "avec pertes", le débit est une fonction $R(D)$ d'une mesure de distorsion D .

Il découle de ce théorème que pour effectuer une bonne transmission il suffit de combiner indépendamment le codage de source et le codage de canal. De notre côté, on s'intéresse seulement au codage de source et plus spécifiquement au codage audio. Notre seul objectif est donc de comprimer l'information à son maximum pour une qualité de restitution donnée (ou l'inverse, avoir la meilleure qualité possible pour un débit donné). La qualité d'un son peut être une erreur calculée à l'aide d'une distance mathématique (distance euclidienne entre le signal d'origine et son approximation) ou tout simplement une qualité subjective (qualité qui dépendra de l'auditeur) qu'on a évaluée communément par des tests d'écoute.

Depuis trois décennies, beaucoup de travaux se sont penchés sur le codage de source d'un signal audio numérisé et ont abouti à des progrès considérables. Ceci a permis notamment de développer des applications telles que la téléphonie mobile, la transmission de la voix sur les réseaux par paquets ou encore le stockage et la diffusion de musique de haute qualité. Ces résultats ont été obtenus grâce entre autres aux organismes de normalisation internationale comme l'ITU-T (*International Telecommunication Union*), l'ETSI (*European Telecommunication standards Institute*) et l'ISO (*International Organization for Standardization*) [17]. Néanmoins, il reste encore des progrès à faire afin d'améliorer les performances des codeurs et de leur adjoindre de nouvelles fonctionnalités, comme la hiérarchisation de l'information.

Historiquement, le codage de la parole et le codage générique audio ont été étudiés séparément et se basent ainsi sur des technologies différentes. Nous allons donc dans un premier temps parler du codage de la parole, puis dans un second temps introduire le codage par transformée habituellement utilisé par les codeurs génériques. La dernière partie du chapitre est consacrée au codage universel à bas débit.

2.2 Codage de la parole

Pour obtenir de bonnes performances de compression d'un signal, il est nécessaire de bien connaître les caractéristiques de ce dernier et la façon dont il est perçu. Dans le cas de la parole, il s'agira de bien modéliser l'appareil vocal et de prendre en compte les caractéristiques de l'audition humaine.

2.2.1 Modélisation de la parole

Le signal de la parole n'est pas un signal déterministe. Il ne peut pas être décrit a priori de façon précise. Néanmoins, ses caractéristiques fortes permettent de bien le modéliser. Une bonne modélisation de la source permet alors une représentation fidèle et compacte de l'information qu'elle transporte.

L'appareil vocal humain est schématisé à la Figure 2.3. Dans un premier temps, l'air comprimé par les poumons (1 sur Figure 2.3) vient exciter les cordes vocales (2 sur Figure 2.3). On distingue généralement deux principales formes d'excitation. Dans le premier cas, les cordes vibrent à une fréquence fondamentale appelée *pitch*. Cela produit alors un son *voisé*. Généralement, la fréquence est relativement faible allant de 50 Hz pour les hommes à 600 Hz pour les voix les plus aiguës telle une voix d'enfant. Dans le second cas, les cordes vocales ne vibrent pas, et le son dit *non-voisé* peut alors s'apparenter à un bruit blanc. Le système exciteur est en réalité plus complexe comme c'est le cas pour la production des fricatives voisées où l'excitation vient en même temps de la vibration des cordes et d'un point de constriction. Après l'excitation du flot d'air, le signal est amplifié par le conduit vocal (3 sur Figure 2.3) qui est formé du pharynx, de la cavité buccale, de la cavité labiale et de la cavité nasale. Ce conduit vocal est un résonateur qui amplifie certaines caractéristiques spectrales. On obtient alors des formants qui sont un facteur fondamental dans la caractérisation du timbre.

La modélisation du signal parole est souvent donnée par le schéma de principe proposé à la Figure 2.4. Un filtre de synthèse est excité soit par un train d'impulsions périodiques à la fréquence fondamentale (son *voisé*), soit par un bruit aléatoire (son *non-voisé*) [18]. C'est le modèle appelé source-filtre. L'analogie avec l'organe de production de la parole est alors simple. L'énergie du flot d'air produit par les poumons (1) est modélisée par la multiplication de l'excitation par un gain. Cette excitation est soit un train d'impulsions périodiques soit un bruit blanc selon la

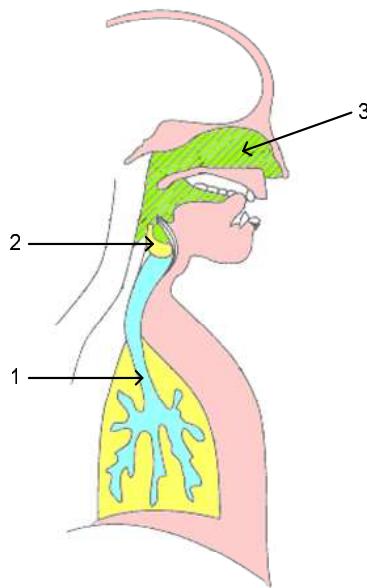


Figure 2.3 Mécanisme de production de la parole.

contribution ou la non-contribution des cordes vocales (2). Finalement, le filtre de synthèse joue le rôle du conduit vocal (3).

Cette modélisation est assez simple et aura des performances restreintes. En effet l'excitation est seulement à deux états, voisée ou non-voisée. Des modèles source-filtre plus évolués existent et permettent de meilleures performances. Nous allons voir par la suite les grandes catégories des codeurs de parole existants et les différentes méthodes de représentation associées. Dans un premier temps, nous allons présenter les caractéristiques fondamentales du signal parole pouvant être modélisées par un codage prédictif, codage étant à la base de nombreux codeurs de parole actuels.

2.2.2 Codage prédictif

La parole peut être considérée comme étant un signal pseudostationnaire, c.-à-d. stationnaire sur de courtes durées allant en général de 20 à 30 ms. Sur cette période il est possible de caractériser le spectre du signal par deux attributs :

- L'enveloppe spectrale.
- La structure fine du spectre.

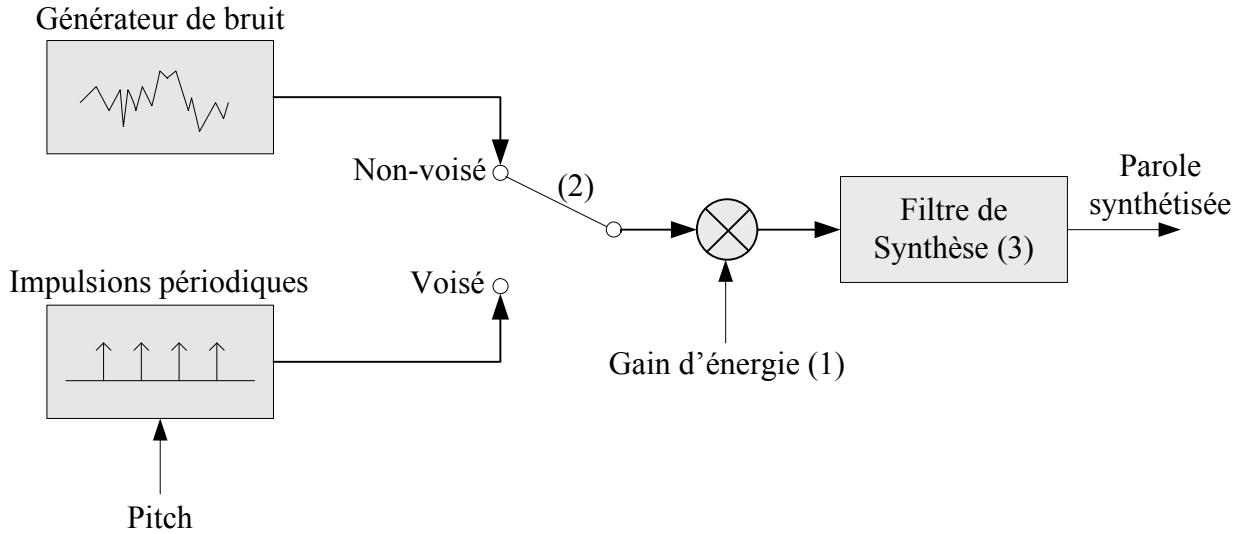


Figure 2.4 Modèle source-filtre de production de la parole.

L'enveloppe spectrale est reliée à la forme du conduit vocal alors que la structure fine du spectre correspond à la contribution de l'excitation. Pour les segments voisés de la parole, la structure fine du spectre est une structure harmonique avec une fréquence fondamentale correspondant à la fréquence du pitch. Cette harmonicité correspond simplement à la périodicité du signal sur une certaine durée d'analyse. Si par contre la structure fine du spectre n'est pas une structure harmonique, alors la parole n'est pas voisée. Le théorème d'autocorrélation (appelé théorème de Wiener-Khintchine) montre que la fonction d'autocorrélation d'un signal est la transformée de Fourier inverse de sa densité spectrale de puissance :

$$R_x(\tau) = \frac{1}{2\pi} \int_{-\pi}^{\pi} |F_x(\omega)|^2 e^{j\omega\tau} d\omega \quad (2.1)$$

De cette propriété, on en déduit que les variations lentes du spectre, c.-à-d. l'enveloppe spectrale, correspondent à la fonction d'autocorrélation pour un τ faible. Inversement, les variations rapides du spectre, c.-à-d. la structure fine, correspondent à la fonction d'autocorrélation pour un τ élevé. Le calcul de ces deux autocorrélations permet donc d'extraire deux composantes importantes du signal parole. C'est le rôle des deux prédictions à court et à long terme utilisées en compression de la parole.

2.2.3 Prédiction à court terme

La prédiction à court terme permet de modéliser l'enveloppe spectrale de la parole qui correspond à la réponse impulsionale du conduit vocal. Ce type de codage repose sur la constatation que les échantillons adjacents de la parole sont fortement corrélés. Dans l'hypothèse où les échantillons sont liés par une relation linéaire, l'échantillon présent peut être prédit par une combinaison linéaire de ses échantillons voisins. On parle alors de codage prédictif linéaire (*Linear Predictive Coding*, LPC). Habituellement cette combinaison se limite aux échantillons passés.

$$\tilde{s}(n) = \sum_{k=1}^p a_k s(n-k) \quad (2.2)$$

où p est l'ordre de prédiction, les a_k représentent les coefficients de la prédiction et $\tilde{s}(n)$ la prédiction de $s(n)$. L'ordre p de prédiction est relativement petit car la prédiction à court terme n'exploite seulement que la redondance des échantillons voisins. Pour un codage en bande étroite, c.-à-d. pour une fréquence d'échantillonnage de 8 kHz, on utilise généralement une prédiction d'ordre 10 comme c'est le cas pour la norme G.729 [5]. Pour un codage en bande élargie, c.-à-d. pour une fréquence d'échantillonnage de 8 kHz, on utilise plutôt un ordre de 16.

La prédiction 2.2 est une façon efficace de représenter la corrélation des coefficients. L'observation passée permet de réduire l'incertitude sur l'échantillon présent à coder, ce qui engendre une réduction du nombre de bits nécessaire pour transmettre l'information. Le résidu de la prédiction est obtenu en soustrayant à l'échantillon présent la prédiction $\tilde{s}(n)$. Il correspond alors à l'excitation du modèle source-filtre.

$$\begin{aligned} e(n) &= s(n) - \tilde{s}(n) \\ &= s(n) - \sum_{k=1}^p a_k s(n-k) \\ &= \sum_{k=0}^p a_k s(n-k) \quad \text{avec } a_0 = 1 \end{aligned} \quad (2.3)$$

Cette opération correspond à un filtre dit d'analyse, $A(z)$, formé des coefficients a_k issus de la prédiction. Le filtre inverse, dit de synthèse, est alors un filtre autorégressif (AR) :

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - \sum_{k=1}^p a_k z^{-k}} \quad (2.4)$$

Minimiser le signal $e(n)$ pour une durée où le signal $s(n)$ est considéré stationnaire permet de trouver la relation linéaire optimale qui lie les échantillons voisins, c.-à-d. trouver les coefficients a_k

du filtre de synthèse H . De cette relation se déduit alors le filtre de synthèse $H(z)$ qui représente au mieux l'enveloppe spectrale du segment. L'optimisation d'une telle équation peut se ramener à un système mettant en jeu soit des équations d'autocorrélation ou de covariance [19]. La résolution de tels systèmes se fait généralement par des méthodes itératives telles que l'algorithme récursif de Levinson-Durbin [20]. La recherche des coefficients du filtre de synthèse est appelée analyse LP (*Linear Predictive*). La Figure 2.5 donne un exemple d'un segment voisé de parole représenté dans le domaine temporel et fréquentiel. La modélisation de l'enveloppe spectrale par une analyse LP d'ordre 10 est comparée au spectre d'amplitude.

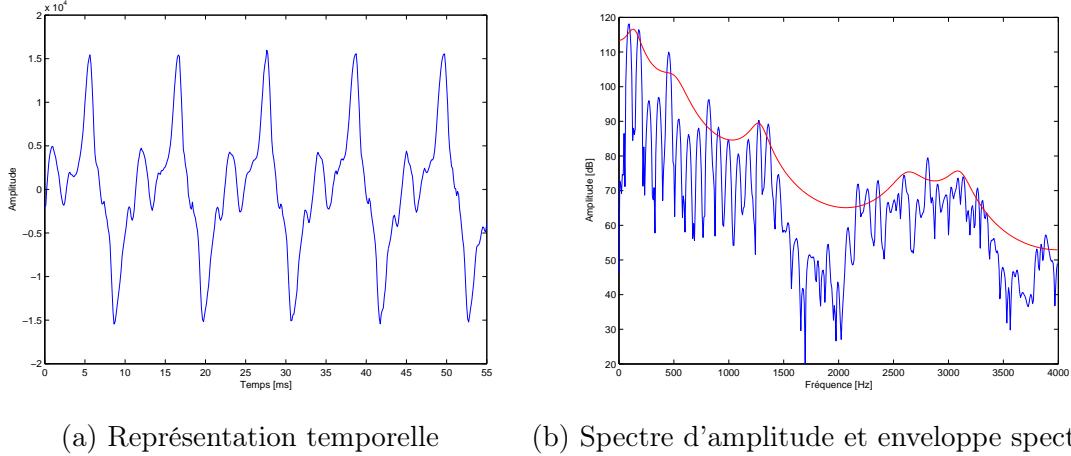


Figure 2.5 Représentations d'un segment voisé de parole.

L'estimation de l'enveloppe spectrale de la parole par un modèle AR est juste lorsque l'excitation est une simple impulsion ou un bruit blanc [21]. Elle devient plus grossière lorsque l'excitation est quasi périodique lors de segments voisés. Malgré cela elle est majoritairement utilisée en codage de la parole depuis son introduction en 1961 par Atal et Shroeder [22]. On peut citer tout de même d'autres techniques de modélisation de l'enveloppe spectrale. L'analyse cepstrale ajoute des zéros [23, 24] au modèle source-filtre afin qu'il devienne plus général. D'autre part, la résolution fréquentielle de l'analyse peut être modifiée pour mieux se rapprocher de celle de l'oreille humaine en utilisant des techniques comme l'analyse WLP (*Warped LP*) [25] ou bien l'analyse mel-cepstrale [26].

2.2.4 Prédition à long terme

Bien que la prédition à court terme supprime la corrélation entre les échantillons adjacents, il reste une corrélation à plus long terme se traduisant par des pics d'amplitudes quasi périodiques au niveau du résidu de la prédition à court terme. Ces pics sont d'autant plus marqués que le signal est voisé. Cette périodicité, correspondant au pitch du signal, peut être estimée par une prédition à long terme. La période du pitch comme nous l'avons vu précédemment va de 50 Hz à 600 Hz, ce qui correspond à une période de 2 à 20 ms, période bien au-delà de la durée de la prédition à court terme de l'ordre de la milliseconde.

Une forme générale du filtre de prédition à long terme est la suivante :

$$P(z) = 1 - \sum_{k=-m_1}^{m_2} b_k z^{(-T+k)} \quad (2.5)$$

T est la période fondamentale et les b_k les gains de prédition. Ce genre de filtre *multi-taps* est très rarement utilisé. On lui préfère une version simplifiée à l'extrême avec $m_1 = m_2 = 0$. On obtient alors un filtre *1-tap* :

$$P(z) = 1 - bz^{(-T)} \quad (2.6)$$

Le filtre inverse est stable si b est inférieur à 1. Si b est supérieur ou égal à 1 sur une courte période, alors l'instabilité n'a pas trop d'influence sur le signal parole reconstruit.

Généralement, les codeurs de parole tirent un gain de codage important de la prédition à long terme, du fait que la parole est un signal monoharmonique. Sa fréquence fondamentale est alors une caractéristique très importante et représentative du signal. La valeur du pitch est pour cette raison mise à jour fréquemment environ toutes les 5 ms. Dans le cas d'une simple analyse, la résolution du délai est contrainte à ne pas dépasser la période d'échantillonnage. Il se peut alors que la résolution ne soit pas suffisante surtout lorsque la fréquence d'échantillonnage est faible. Une analyse plus fine pourra utiliser une résolution fractionnaire du délai. De plus, une première estimation de la période fondamentale en boucle ouverte peut être raffinée par une deuxième analyse en boucle fermée. Une meilleure correspondance entre les échantillons présents et retardés sera ainsi faite ce qui engendra une diminution de l'erreur de prédition.

2.2.5 Les différentes approches en codage de la parole

Les méthodes de codage de la parole sont nombreuses et sont classées généralement en trois grandes catégories : les codeurs de forme d'onde, les codeurs paramétriques, appelés vocodeurs, et les codeurs hybrides. Le choix d'une méthode va dépendre surtout de l'application visée et des contraintes sur le débit. Les codeurs de forme d'onde sont surtout performants pour des débits élevés (au-delà des 16 kbit/s). Les vocodeurs quant à eux sont plutôt destinés aux très bas débits (au-dessous de 2.4 kbit/s) et aux bas débits (2.4 à 8 kbit/s). Les codeurs hybrides ont des débits nominaux intermédiaires (8 à 16 kbit/s), bien qu'ils puissent aussi être utilisés pour de bas débits. Une autre façon de distinguer les codeurs de parole est la largeur de bande codée. Historiquement, les codeurs de parole sont à bande étroite, c.-à-d. codent le signal sur la bande de 300 à 3400 Hz. Le signal d'entrée est alors échantillonné à 8 kHz. Ces dernières années, la tendance est d'augmenter la largeur de bande vers la bande élargie de 50 à 7000 Hz en utilisant une fréquence d'échantillonnage de 16 kHz. La qualité de la synthèse et de la communication s'en trouve nettement améliorée [27].

Le codage de forme d'onde

Les codeurs de forme d'onde s'attellent à représenter le plus exactement possible la forme de l'onde du signal sans nécessairement exploiter les propriétés de la parole et de l'audition. De ce fait, ils sont en général plus indépendants du signal d'entrée que les autres types de codeurs. En contrepartie, ils génèrent des débits plus élevés.

Le codage de forme d'onde la plus simple est sans conteste la modulation d'impulsion codée (*Pulse-Code Modulation*, PCM). Il s'agit d'une simple quantification scalaire uniforme ou non uniforme des amplitudes temporelles. À une fréquence d'échantillonnage de 8 kHz, le codage sur 8 bits de chaque échantillon par une quantification non uniforme permet de reproduire une synthèse indiscernable de l'original pour un débit de 64 kbit/s. On considère que c'est une technique d'échantillonnage sans compression. Par contre, la modulation d'impulsion codée différentielle adaptative (ADPCM *Adaptive Differential Pulse Code Modulation*) utilise un codage prédictif exploitant la corrélation interéchantillon à court terme. Elle atteint les mêmes performances du codage PCM avec de plus faibles débits. La Figure 2.6 est le schéma de principe d'un codeur ADPCM. La norme ITU-T G.721 [28] utilise un codage ADPCM à 32 kbit/s. Elle a été étendue pour les débits 16, 24 et 40 kbit/s avec les normes G.723 et G.726 [29]. Le codage ADPCM

offre une bonne qualité de codage pour des débits assez élevés mais ses performances chutent rapidement en dessous de 24 kbit/s. Un autre inconvénient majeur de ce type de codage est la faible robustesse face aux erreurs de canal.

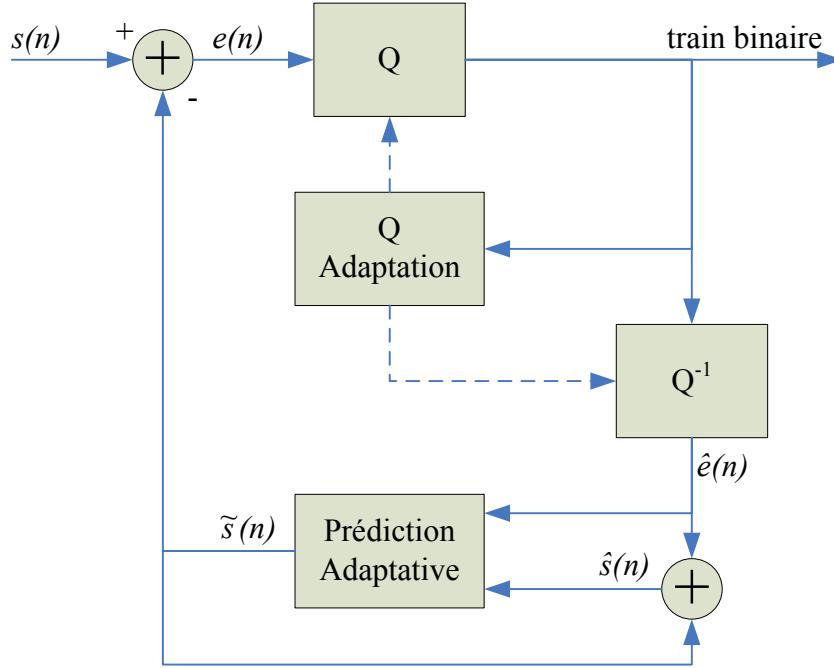


Figure 2.6 Système ADPCM.

Il existe aussi une version large bande de l'ADPCM avec la norme G.722 [6]. Elle combine une décomposition en deux sous-bandes de fréquence de largeurs égales (0-4000 Hz et 4000-8000 Hz) et une version modifiée du G.721 pour chaque sous-bande. Les débits de sortie sont soit 48, 56 ou 64 kbit/s.

Le codage de forme d'onde de la parole peut se faire aussi dans le domaine fréquentiel, bien que ce domaine se prête moins bien que le domaine temporel pour exploiter les caractéristiques remarquables d'un tel signal. Le codeur de parole large bande G.722.1 [30] utilise une transformation avant de quantifier les coefficients fréquentiels. Ce procédé, surtout utilisé et efficace pour le codage générique audio, sera abordé par la suite. Le G.722.1 travaille aux débits de 24 kbit/s et 32 kbit/s.

Les vocodeurs

Les vocodeurs posent un a priori important sur le signal d'entrée comme étant un signal issu de la parole humaine. Ils utilisent donc au maximum les caractéristiques de la parole vues précédemment. La figure 2.4 représente le modèle source-filtre généralement utilisé par les vocodeurs, où une série de paramètres est codée afin de caractériser suffisamment le signal d'entrée pour que la synthèse vocale soit intelligible sans essayer de restituer la forme d'onde exacte du signal. La qualité de reconstitution est bien sûr limitée par la justesse de la modélisation. Ainsi, sa qualité se trouve être relativement basse par rapport aux deux autres catégories de codeurs mais son avantage est que son débit l'est aussi. Comme exemple de vocodeur, on peut citer les codeurs homomorphiques qui utilisent la déconvolution homomorphique pour extraire les paramètres du cepstre associés au conduit vocal et au pitch [23, 24]. La famille des vocodeurs à prédiction linéaire quant à elle utilise un codage LPC pour modéliser le conduit vocal. C'est le modèle utilisé par la recommandation LPC-10 pour des communications fiables à 2.4 kbit/s [31]. L'excitation est généralement modélisée soit par une excitation à deux états (voisé ou non-voisé) soit par une mixture d'une composante harmonique et d'une composante stochastique.

Codage hybride

Le codage hybride est un compromis intéressant entre les deux catégories précédentes en termes de qualité et de débit. Il associe une modélisation poussée du signal de parole avec une optimisation par analyse par synthèse. L'analyse par synthèse consiste en une boucle fermée permettant d'optimiser le codage de l'excitation en minimisant une mesure de la différence entre le signal original et sa synthèse. La Figure 2.7 donne le schéma bloc d'un codage par analyse par synthèse.

Le codage CELP (*Code Excited Linear Prediction*) a été introduit par Atal et Schroeder [3] et fait parti des codeurs hybrides. Il est très efficace pour les débits intermédiaires de 5 kbit/s à 16 kbit/s, comme en témoignent les nombreuses normes qui l'utilisent [32, 33, 5, 7]. La Figure 2.8 représente le principe du codage CELP.

Le système consiste en un filtre de synthèse $1/A(z)$ à court terme représentant l'enveloppe spectrale du signal. Une prédiction linéaire à long terme représentée par le filtre $P(z)$ permet de modéliser la périodicité du signal, c.-à-d. le pitch. Pour diminuer la complexité du codeur, la contribution du pitch est souvent traduite par l'excitation passée et périodisée par la période fondamentale obtenue par la prédiction. On parle alors de dictionnaire adaptatif. Sa sortie est

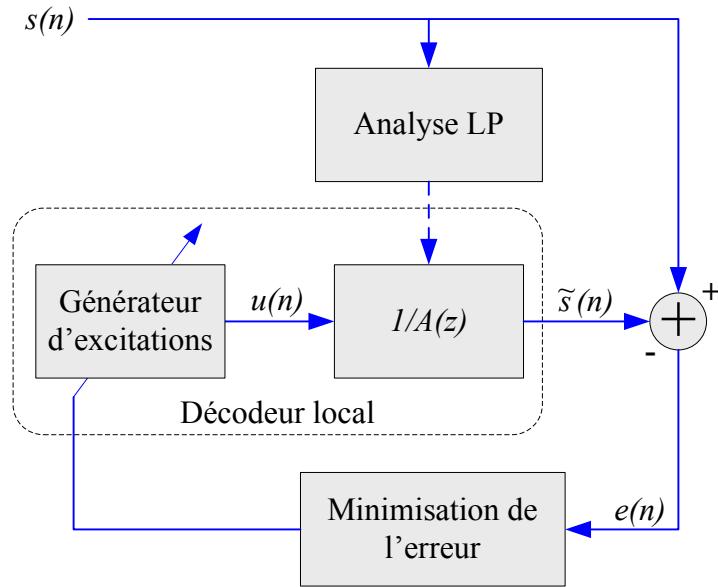


Figure 2.7 Principe de codage par analyse par synthèse.

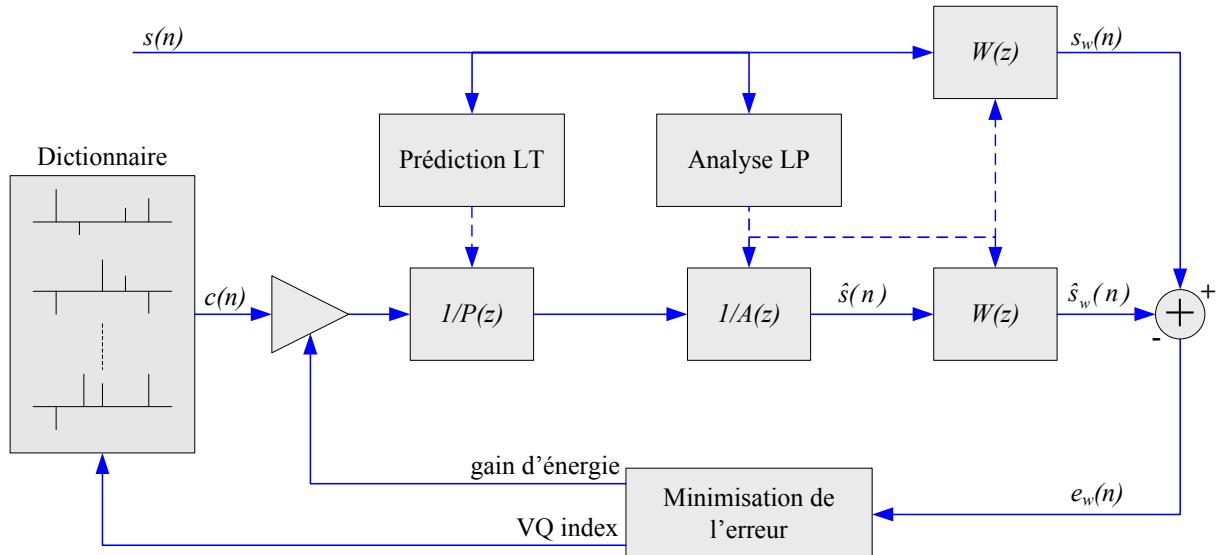


Figure 2.8 Principe du codage CELP.

ajoutée à un vecteur du dictionnaire innovateur choisi à l’issue de l’analyse par synthèse. La minimisation de l’erreur se fait selon un critère perceptuel, qui est obtenu en filtrant la synthèse par un filtre dit perceptuel ou pondéré $W(z)$. Ce filtre pondère l’erreur dans le domaine fréquentiel en tenant compte des caractéristiques de l’audition humaine. Il atténue les zones du spectre à forte amplitude (formants) et amplifie les zones de faible amplitude. Une forme classique du filtre est la suivante :

$$W(z) = \frac{A(z/\gamma_1)}{A(z/\gamma_2)} \quad \text{avec} \quad 0 < \gamma_2 < \gamma_1 \leq 1 \quad (2.7)$$

Le facteur γ_2 a pour effet de déplacer les pôles vers le centre du cercle par rapport aux pôles du filtre de synthèse $1/A(z)$. Le filtre est donc plus stable, il y a moins de résonance au niveau des formants. La figure 2.9 représente le spectre de $W(z)$ pour $\gamma_1 = 0.98$ et $\gamma_2 = 0.6$. On voit bien que $W(z)$ est formé d’antirésonances au niveau des formants du signal. Le filtre $W(z)$ favorise alors l’erreur de quantification à avoir une énergie plus importante au niveau des formants hauts en énergie.

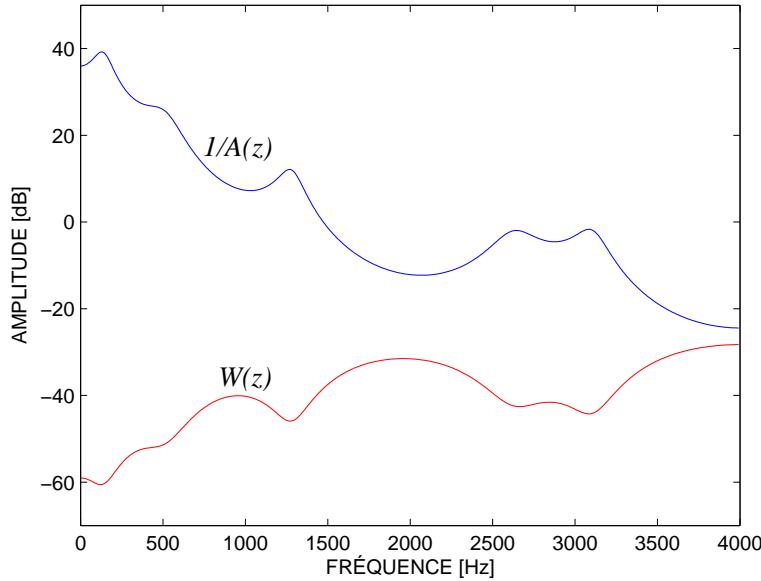


Figure 2.9 Spectres du filtre de synthèse et du filtre perceptuel.

Dans la version originale du CELP, le dictionnaire innovateur était un dictionnaire stochastique nécessitant d’une part de la mémoire de stockage, mais surtout un temps de calcul non négligeable

pour la recherche du meilleur représentant par le processus d'analyse par synthèse. Les efforts de recherche se sont alors concentrés sur ce problème. On peut citer par exemple la technologie *Vector-Sum Excited Linear Prediction* (VSELP) [33] utilisant des dictionnaires très structurés afin de réduire la complexité algorithmique. Elle a été adoptée en Amérique du Nord pour la téléphonie mobile de deuxième génération. Mais la technologie la plus répandue est la technologie ACELP (*Algebraic-CELP*) [34]. Elle utilise un dictionnaire algébrique pour modéliser l'innovation permettant ainsi de réduire en plus de la complexité algorithmique, les ressources de stockage. Elle est utilisée par de nombreux codeurs modernes de parole pour des communications sans fil et filaires, tels le G.729 [5] et l'AMR-WB [7].

2.3 Codage par transformée

On vient de voir dans la section précédente que pour un signal de parole les performances d'un codage de type CELP sont dues à la pertinence de la modélisation de l'appareil de production. Pour des signaux plus généraux, comme la musique, il n'existe pas de modèles de production satisfaisant, car les sources sont diversifiées. Par contre, il est possible d'utiliser un modèle d'audition commun. Pour avoir une compression efficace, le principe est de transmettre uniquement l'information perceptible. Le masquage auditif le plus important, et sûrement le plus facile à estimer, est le masquage simultané. Comme ce masquage se fait dans le domaine fréquentiel, il est alors pertinent de coder le signal audio dans ce même domaine. L'utilisation d'une transformation linéaire adaptée permet alors de passer du domaine temporel initial à un domaine fréquentiel. De plus, cet outil permet aussi d'exploiter les caractéristiques statistiques du signal, ce qui génère un gain de codage.

2.3.1 Modélisation de l'appareil auditif humain

La perception auditive humaine est un processus complexe. On sait du moins qu'une analyse temporelle et fréquentielle est réalisée au niveau de l'oreille. Lors d'une compression avec pertes, c.-à-d. lors de l'utilisation d'une quantification dans le processus de compression, un bruit va venir s'ajouter au signal original pour former le signal reconstruit. Dans le cas où la transmission est considérée parfaite, c'est l'unique source de bruit de la chaîne de transmission. Ce bruit détériore le signal, mais dans certains cas il peut être rendu inaudible à l'oreille humaine : c'est le phénomène de masquage. Le système auditif perçoit des stimuli isolés au-dessus d'un certain

niveau sonore dans le cas où sa fréquence se trouve dans l'intervalle des fréquences audibles. Le seuil de l'audition absolu pour chaque fréquence est bien connu. Par contre dans des conditions de masquage où un autre signal est présent dans la même zone fréquentielle, un stimulus peut ne pas être audible bien qu'il soit au-dessus du seuil d'audibilité. Pour qu'il le soit, son énergie doit dépasser un nouveau seuil appelé seuil de masquage. Le but du codage perceptuel est de définir le seuil de masquage associé au signal à coder et faire en sorte que le bruit injecté lors de la quantification soit au-dessous de ce seuil. En ajustant la quantification du signal judicieusement, il est possible de rendre inaudible l'effet du bruit injecté par la quantification si le débit disponible est suffisamment élevé. De ce fait, l'approximation faite par le codage sera transparente à l'oreille humaine tout en permettant une forte compression de l'information.

Pour définir le seuil de masquage on se base sur des critères psycho-acoustiques, qui prennent en compte le contenu spectral du signal ainsi que des considérations temporelles.

Masquage temporel

Lorsque deux sons ont lieu successivement, deux types de masquage temporel peuvent avoir lieu :

- Le masquage postérieur : le seuil d'audition est modifié par un son masquant qui le précède.
- Le masquage antérieur : le seuil d'audition est modifié par un son masquant qui le suit.

Le masquage postérieur est de loin le plus important des deux masquages. La durée du masquage postérieur est de l'ordre de 25 ms, alors que celui du masquage antérieur est de l'ordre de 5 ms. Ces phénomènes sont difficiles à modéliser donc rarement utilisés.

Masquage fréquentiel

Avant d'introduire le masquage fréquentiel, encore appelé masquage simultané, nous allons introduire un principe fondamental de la psycho-acoustique [35] : les filtres auditifs.

Le fonctionnement de l'oreille interne peut être modélisé par un banc de filtres à forts recouvrements dont la largeur de bande augmente avec les fréquences. La bande de fréquences audibles qui s'étend typiquement de 20 Hz à 20 kHz peut être ainsi divisée en à peu près 24 bandes dites critiques. Un exemple de banc de filtres non uniforme modélisant les bandes critiques auditives est donné à la Figure 2.10. Dans chacune de ces bandes, l'oreille assimile les fréquences et ne les différencie plus. Ces bandes correspondent d'ailleurs à la répartition des cellules ciliées dans l'oreille interne. Au dessous de 500 Hz, les bandes critiques sont de même largeur (environ 100 Hz). Au-delà de 500 Hz elles sont non uniformes. Une façon d'introduire et de mesurer la largeur

d'une bande critique est de considérer un bruit à bande étroite de fréquence centrale f_1 et de largeur de bande df . Ce bruit a une puissance égale à $P_1 = \sigma^2 df$ avec σ^2 constant. Introduisons maintenant une sinusoïde pure de même fréquence f_1 de puissance P_2 juste suffisante pour qu'elle soit audible en présence du bruit à bande étroite. Si l'on augmente df , la puissance P_2 croît d'abord de façon proportionnelle puis n'augmente plus lorsque df devient supérieure à la largeur de la bande critique de fréquence centrale f_1 . Il est possible de cette manière de définir la largeur des bandes critiques de l'oreille humaine [36].

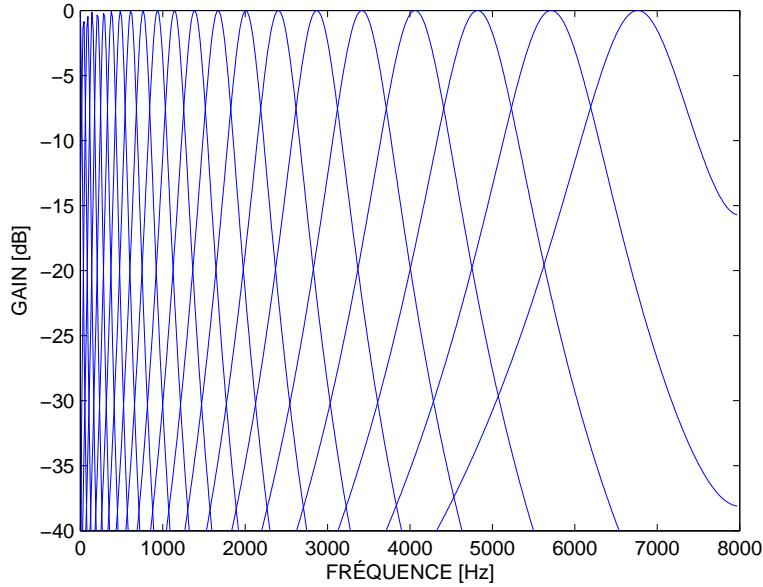


Figure 2.10 Exemple d'un banc de filtres auditif non uniforme.

Le masquage fréquentiel est une caractéristique importante de l'audition qui repose sur la décomposition en bandes critiques. Il dépend essentiellement de la fréquence f , de la puissance P et de la nature du signal masquant. Que le signal soit une sinusoïde ou un bruit à bande étroite, la courbe de masquage a approximativement la même allure en forme de triangle comme illustré à la Figure 2.11. Le maximum de cette courbe se trouve être en f , et l'amplitude en ce point est inférieure à P : cette différence entre la courbe de masquage et la puissance P s'appelle l'indice de masquage. Une propriété remarquable est la dissymétrie importante de la courbe de masquage qui a une pente plus faible vers les fréquences supérieures que vers les fréquences inférieures. Il est possible de modéliser cette courbe par des segments de droite si on utilise une échelle des fréquences adaptée correspondant au découpage en bandes critiques. Une relation entre le numéro

de la bande critique f_{Bark} (en Bark) et la fréquence f (en Hz) a été introduite par Zwicker et Terhardt [37] et est définie comme suit :

$$f_{Bark} = 13 \arctan[0.76 \frac{f}{1000}] + 3.5 \arctan[(\frac{f}{7500})^2] \quad (2.8)$$

Une approximation assez juste du masquage fréquentiel utilise alors deux segments de droite avec respectivement une pente de 30 dB/Bark pour les fréquences inférieures et une pente de -10 dB/Bark pour les fréquences supérieures comme l'illustre la Figure 2.11. On obtient alors une fonction d'étalement.

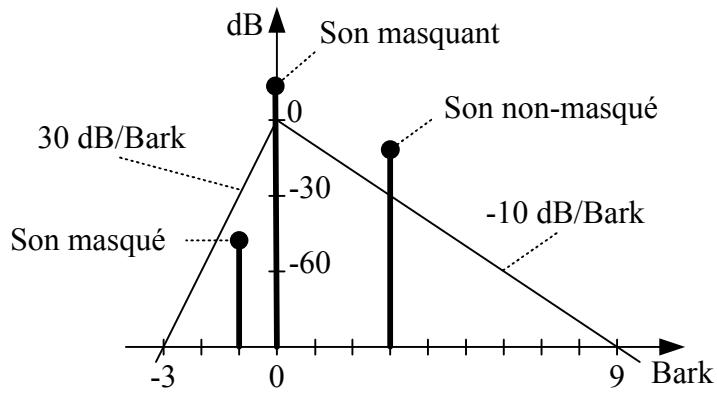


Figure 2.11 Modèle de masquage fréquentiel par une fonction d'étalement.

Plusieurs modèles existent tel le modèle de Johnston [38] ou bien le modèle PEAQ (*Perceptual Evaluation of Audio Quality*) [39]. Ils utilisent tous le principe d'étalement pour calculer un seuil de masquage pour un segment donné d'un signal. Néanmoins, ces modèles doivent généraliser le principe de masquage pour des sons beaucoup plus complexes. Les simplifications des interactions entre les différentes composantes de masquage font que le seuil calculé n'est qu'une approximation du seuil réel [40].

La courbe de masquage aussi juste qu'elle peut l'être, permet de distribuer les ressources disponibles, c'est à dire les bits, entre les différentes composantes du signal, c.-à-d. ses coefficients transformés. L'allocation binaire sera optimale au sens de la perception humaine uniquement lorsque le budget de bits est suffisant pour que le bruit de quantification puisse être en dessous du seuil de masquage pour toutes les fréquences du spectre. Lorsque le débit est faible, l'erreur de quantification sera dans la majorité des cas au-dessus de ce seuil. La mise en forme optimale du bruit perceptible ne découle alors plus uniquement du seuil de masquage. Les connaissances

psycho-acoustiques sont encore trop limitées pour connaître l'allocation optimale. Elle est donc souvent optimisée expérimentalement selon les applications visées. Le filtre de pondération $W(z)$ de l'équation 2.7 est un exemple de ce qui se fait souvent en codage de la parole. Ainsi l'utilisation d'une courbe de masquage est souvent réservée pour le codage générique pour des débits supérieurs à 40 kbit/s.

2.3.2 Transformation Temps-Fréquence

La transformation temps-fréquence est un vaste domaine qui, entre autres, permet l'analyse de signaux. Dans le cas du codage elle permet essentiellement de passer dans un domaine plus propice pour la compression du signal. La section suivante montre le gain atteignable par un tel procédé.

Gain de codage

La transformation d'un signal est un processus permettant de décorrélérer les coefficients et de concentrer l'énergie sur un nombre minimal de coefficients transformés. Ainsi les coefficients d'un signal sont rendus non homogènes, en donnant plus d'importance à certains qu'à d'autres. Le but est de pouvoir exploiter les redondances du signal pour obtenir une distorsion plus faible que celle obtenue en quantifiant directement le signal d'entrée $x(n)$. Le bon choix d'une transformation est alors essentiel pour obtenir un gain en performance. Les propriétés remarquables des transformations unitaires (respectivement orthogonales pour les transformations à coefficients réels) font d'elles un choix privilégié, et en particulier pour le théorème de Parseval (conservation de l'énergie). La Figure 2.12 montre le schéma de principe d'un codeur par transformée utilisant N quantifications scalaires. Si la transformation T est unitaire, on peut écrire :

$$\sigma_x^2 = \frac{1}{N} \sum_{k=0}^{N-1} \sigma_{X_k}^2 \quad (2.9)$$

De cette propriété on en déduit que la variance de l'erreur de reconstruction totale, σ_Q^2 , entre x et \hat{x} , est égale à la moyenne des puissances des erreurs de quantifications, $\sigma_{Q_k}^2$, dans les différentes sous-bandes :

$$\sigma_Q^2 = \frac{1}{N} \sum_{k=0}^{N-1} \sigma_{Q_k}^2 \quad (2.10)$$

D'après la relation fondamentale donnée dans [15], on connaît l'erreur quadratique d'une quantification scalaire Q_k de résolution b_k d'un signal de variance $\sigma_{X_k}^2$ dans l'hypothèse de haute

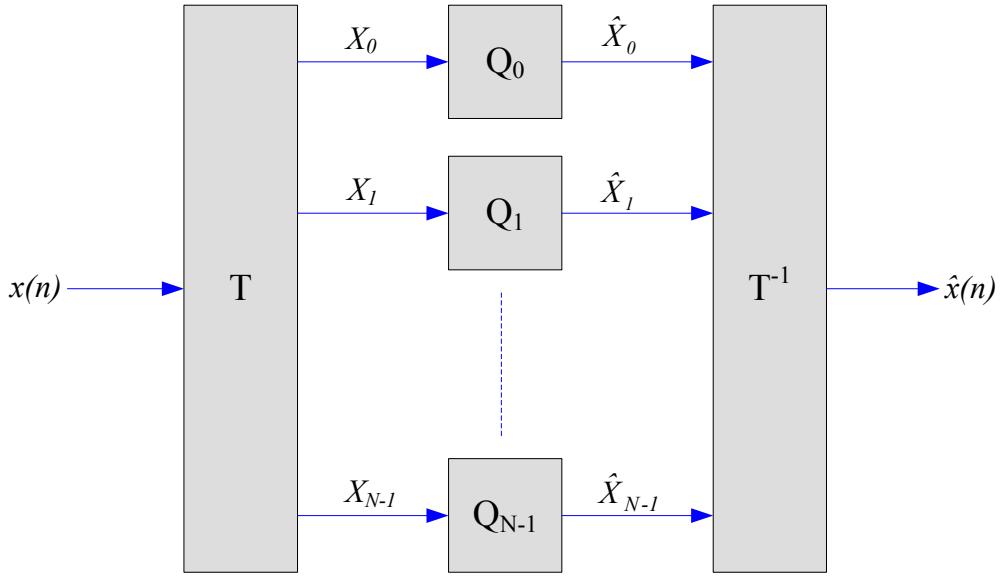


Figure 2.12 Schéma de principe d'un codeur par transformée.

résolution :

$$\sigma_{Q_k}^2 = c(1)\sigma_{X_k}^2 2^{-2b_k} \quad (2.11)$$

$c(1)$ est un facteur de correction qui est égal à 1 si la distribution du signal est uniforme ou bien égal à $\frac{\sqrt{3}}{2}\pi$ si c'est une distribution gaussienne. Dans le cas d'un codage par transformée on obtient l'expression suivante :

$$\sigma_Q^2 = \frac{c(1)}{N} \sum_{k=0}^{N-1} \sigma_{X_k}^2 2^{-2b_k} \quad (2.12)$$

La distorsion globale σ_Q^2 est alors minimisée lorsque pour tout k , l'expression suivante est vérifiée :

$$\sigma_{X_k}^2 2^{-2b_k} = \left(\prod_{k=0}^{N-1} \sigma_{X_k}^2 \right)^{1/N} 2^{-2 \sum_{k=0}^{N-1} b_k / N} \quad (2.13)$$

Si on pose b comme étant le nombre moyen de bits par composante, alors l'allocation optimale des bits au sens de l'erreur quadratique moyenne est donnée par l'équation suivante :

$$b_k = b + \frac{1}{2} \log_2 \frac{\sigma_{X_k}^2}{\left(\prod_{k=0}^{N-1} \sigma_{X_k}^2 \right)^{1/N}} \quad (2.14)$$

L'équation 2.13 signifie que la quantification scalaire optimale entraîne la même erreur quadratique moyenne pour chaque variable aléatoire X_k . C'est le principe du remplissage inverse des

eaux [41]. Bien sûr cette méthode optimale est difficile à mettre en oeuvre dans la pratique du fait principalement que le nombre de niveaux de reconstruction, voire le débit de chaque quantification, doit être un entier. Il existe des algorithmes utilisés dans la pratique comme l'algorithme glouton (*greedy algorithm*) [14], qui alloue 1 bit à la fois au quantificateur dont l'erreur quadratique engendrée est la plus grande. À chaque itération, l'erreur quadratique de la quantification associée à une variable, tout d'abord initiée à l'énergie de la composante non quantifiée, est divisée par 4 lorsqu'un bit lui est alloué. Dans ce cas-ci, le critère de distorsion est l'erreur quadratique (*Mean Square Error*, MSE). Pour les codeurs perceptuels, le critère dépend d'un modèle perceptuel. Dans le cas où un seuil de masquage est calculé, il est possible d'adapter l'allocation pour prendre en compte le rapport bruit à masque (*Noise-to-Mask*, NMR) ou bien le niveau de bruit au-dessous du seuil de masquage (*audible noise energy*).

Si on reste dans le cas de l'erreur quadratique, il est possible de montrer que pour une allocation optimale des bits b_k , on obtient une puissance d'erreur de quantification qui a l'expression suivante :

$$\sigma_Q^2 = c(1) \left(\prod_{k=0}^{N-1} \sigma_{X_k}^2 \right)^{1/N} 2^{-2b} \quad (2.15)$$

De cette expression, le gain du codage par transformée se déduit facilement :

$$G_T = \frac{\sigma_x^2}{\left(\prod_{k=0}^{N-1} \sigma_{X_k}^2 \right)^{1/N}} = \frac{\frac{1}{N} \sum_{k=0}^{N-1} \sigma_{X_k}^2}{\left(\prod_{k=0}^{N-1} \sigma_{X_k}^2 \right)^{1/N}} \quad (2.16)$$

La transformation de Karhunen-Loeve réalise une décomposition du signal $x(n)$ sur les vecteurs propres de la matrice de covariance, ce qui revient à une décorrélation totale du signal. C'est la transformation qui entraîne le gain de transformation le plus élevé. Par contre, cette transformation présente plusieurs inconvénients majeurs. Elle dépend du signal d'entrée, de sorte que pour chaque nouvelle trame les vecteurs et valeurs propres doivent être recalculés et transmis. De plus, elle n'a pas d'interprétation fréquentielle simple. En pratique des transformations de la famille de Fourier sont utilisées : transformée de Fourier discrète (DFT), transformée en cosinus discrète (DCT). En codage audio, les transformations les plus employées sont des transformations avec recouvrement, comme la transformée en cosinus discrète modifiée (MDCT) [42]. Cette classe de transformation issue des transformations modulées permet entre autres d'atténuer les effets de bloc.

Les Transformations avec recouvrement

Les transformations en bloc classiques, comme les transformées discrètes de Fourier (DFT) et en cosinus (DCT), ont le gros inconvénient de créer des effets de bloc, du fait de la transition à brut entre les fenêtres d'analyse rectangulaires. De plus, elles sont peu sélectives en fréquence, ce qui réduit le gain de codage et introduit à la synthèse un étalement spectral excessivement important des erreurs de quantification.

La transformation avec recouvrement permet alors de réduire ces désavantages. La transformation de Fourier à court terme avec un chevauchement des fenêtres d'analyse est un exemple classique et simple à réaliser. Malheureusement, elle produit un suréchantillonnage des coefficients transformés pour une reconstruction parfaite, ce qui va à l'encontre des exigences de la compression du signal.

Au contraire, la contrainte d'échantillonnage critique dans un banc de filtres impose que le nombre d'échantillons obtenus après décomposition d'un bloc du signal soit égal au facteur de décimation M . Ainsi la transformée de Fourier, pouvant être représentée par un banc de filtres, répond à cette contrainte lorsque les fenêtres d'analyse ne se chevauchent pas. D'une manière générale, les bancs de filtres à décimation maximale répondent à cette exigence. Par contre, la contrainte de reconstruction parfaite de la synthèse en absence d'erreur de quantification est beaucoup plus dure à satisfaire. Les filtres miroirs en quadrature (*Quadrature Mirror Filter*, QMF) [43] et les filtres conjugués en quadrature (*Conjugate Quadrature Filter*, CQF) [43] sont des décompositions du signal en deux sous-bandes ($M = 2$) et à échantillonnage critique qui permettent respectivement une quasi-reconstruction parfaite et une reconstruction parfaite du signal. La généralisation à une décomposition en $M > 2$ sous-bandes peut se faire par un banc de filtres pseudo-QMF [44], ou dans le cas de la reconstruction parfaite, par une transformation modulée avec recouvrement (*Modulated Lapped Transform*, MLT) [45]. La figure 2.13 représente un banc de filtres à échantillonnage critique à M sous-bandes. Une telle représentation est possible pour les MLT. Si $x(n) = \hat{x}(n)$ en absence de quantification des X_k , alors la transformation est à reconstruction parfaite.

Une transformation modulée est définie par un filtre prototype de taille N et par un nombre M de sous-bandes. Les M filtres d'analyse h_k sont alors obtenus par modulation d'un filtre prototype d'analyse h par des fonctions périodisées de période M issues d'une transformation unitaire T de taille M . Les filtres de synthèse g_k sont obtenus par une modulation d'un filtre prototype de

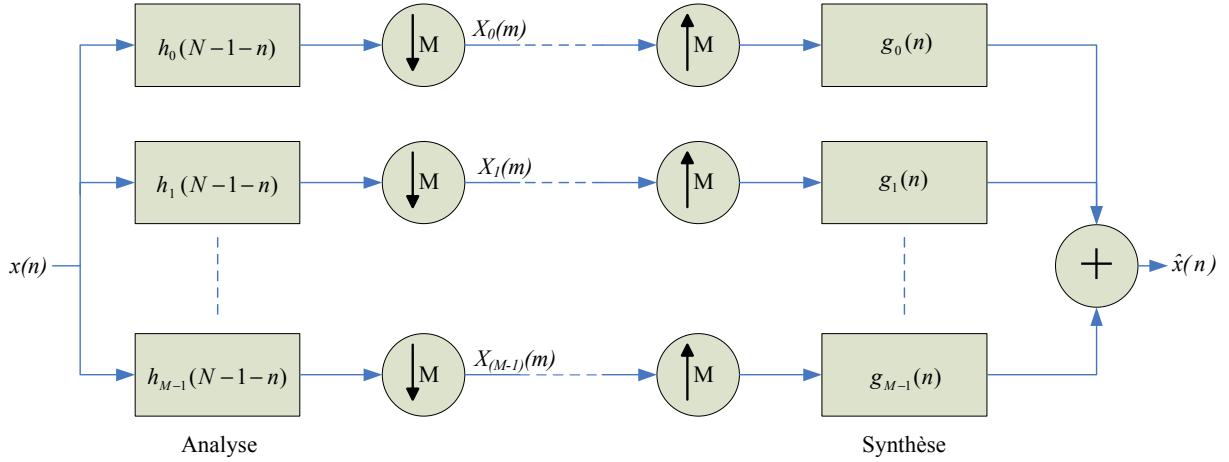


Figure 2.13 Représentation d'une transformation modulée par un banc de filtres.

synthèse g par les fonctions périodisées de la transformation inverse $T^{-1} = T^T$. On peut écrire sous forme matricielle la transformation. On pose \tilde{H} la matrice $M \times N$ formée des vecteurs lignes h_k , \tilde{T} la matrice $M \times N$ la matrice formée des fonctions périodisées de la transformation T , et H la matrice diagonale $N \times N$ ayant les composantes de h sur sa diagonale. On a alors :

$$X(m) = \tilde{H}x(m) = \tilde{T}Hx(m) \quad (2.17)$$

Si on prend l'exemple que la transformation T est une DFT, on peut alors expliciter les matrices comme suit :

$$\tilde{H} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{-j\frac{2\pi}{M}} & \dots & e^{-j\frac{2\pi}{M}(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{2\pi}{M}(M-1)} & \dots & e^{-j\frac{2\pi}{M}(M-1)(N-1)} \end{bmatrix} \cdot \begin{bmatrix} h(0) & 0 & \dots & 0 \\ 0 & h(1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & h(N-1) \end{bmatrix} \quad (2.18)$$

On voit bien que si $N = M$ alors on est dans le cas particulier d'une transformation DFT en bloc, et la reconstruction parfaite est assurée pour des filtres prototypes h et g équivalents à des fenêtres rectangulaires. La théorie des transformations modulées permet ainsi de faire un lien entre le codage en sous-bandes et le codage par transformée. Dans le cas où on utilise le même filtre prototype pour l'analyse et la synthèse ($g = h$) et que celui-ci est symétrique ($h(n) = h(L - n - 1)$), donc à phase linéaire, alors on obtient une classe de transformées orthogonales avec recouvrement (*Lapped Orthogonal Transform*, LOT) [45] appelées *Perfect Reconstruction*

Modulated Filter Banks (PRMF) si elles vérifient les conditions de reconstruction parfaite [46]. Dans le cas particulier où $N = 2M$ et que les filtres prototypes sont modulés par des cosinus, on obtient la *Time Domain Alias Cancellation* (TDAC), connue encore sous le nom de MDCT. Les conditions de la reconstruction parfaite s'écrivent alors :

$$\begin{cases} h^2(n) + h^2(n + M) = 1 \\ h(n) = h(N - 1 - n) \end{cases} \quad (2.19)$$

Les deux conditions sont appelées respectivement condition de Nyquist et condition de phase linéaire. Les fonctions de base d'une MDCT ont la forme suivante [42] :

$$h_k(n) = h(n) \sqrt{\frac{2}{M}} \cos \left(\left(\frac{2\pi}{N} \left(k + \frac{1}{2} \right) \right) (n + n_0) \right) \quad (2.20)$$

$$g_k(n) = g(n) \sqrt{\frac{2}{M}} \cos \left(\left(\frac{2\pi}{N} \left(k + \frac{1}{2} \right) \right) (n + n_0) \right) \quad (2.21)$$

où $k = 0, 1, \dots, M - 1$ et $n = 0, 1, \dots, 2M - 1$ et $n_0 = \frac{1}{2} + \frac{N}{4}$, avec $N = 2M$. La MDCT est alors définie comme suit :

$$X(k) = \sqrt{\frac{2}{M}} \sum_{n=0}^{N-1} h(n) x(n) \cos \left(\left(\frac{2\pi}{N} \left(k + \frac{1}{2} \right) \right) (n + n_0) \right) \quad (2.22)$$

La transformée inverse est donnée par :

$$y(n) = g(n) \sqrt{\frac{2}{M}} \sum_{n=0}^{M-1} X(k) \cos \left(\left(\frac{2\pi}{N} \left(k + \frac{1}{2} \right) \right) (n + n_0) \right) \quad (2.23)$$

Le gain de codage après une transformation MDCT est plus important qu'avec les transformations en bloc [45]. De nombreux travaux ont démontré cet avantage pour différentes fenêtres [45, 47]. Finalement, le principal inconvénient vient du délai engendré qui est une fois et demie plus important dans le cas de la MDCT que pour une transformation en bloc avec le même nombre de sous-bandes. Pour des codages exigeants en terme de délai, une transformée en bloc peut être alors remplacée avantageusement par une MDCT de plus petite taille ayant un gain de codage avoisinant. Les effets de bloc seront alors atténués. Elle est d'autant plus attrayante qu'il existe des algorithmes rapides de calcul de la MDCT qui permettent l'utilisation d'une transformation rapide de Fourier (*Fast Fourier Transform*, FFT) de $N/4$ points [48]. Tous ces aspects font que la MDCT est un choix privilégié de nos jours en codage audio.

Il est à noter que les MLT à reconstruction parfaite ne sont pas orthogonales sur une seule fenêtre d'analyse, comme c'est le cas pour les transformations en bloc. Mais c'est sur un ensemble de

fenêtres qu’elles le sont. Le processus d’addition est donc indissociable de la reconstruction parfaite. Cette propriété peut être contraignante dans certaines situations par rapport à l’utilisation d’une transformation en bloc [49].

Les bancs de filtres multirésolution, comme la transformée en ondelettes ou en paquets d’ondelettes [50, 51], sont attrayants du fait du relâchement de la contrainte sur l’uniformité de la largeur des sous-bandes. Il paraît alors évident qu’une division de l’axe des fréquences suivant les bandes critiques de l’audition humaine peut être avantageuse. Mais il semble que malgré de nombreux travaux sur l’approche multirésolution, la transformée en ondelettes soit moins adaptée à bas débits qu’un découpage fréquentiel uniforme [52, 45]. L’hypothèse implicite du codage multi-résolution que les composantes hautes fréquences ont une petite durée et qu’inversement les composantes basses fréquences une grande, n’est pas forcément applicable pour les signaux audio [53].

2.3.3 Schéma de codage par transformée

Il existe de nombreux codeurs utilisant le codage par transformée ou par sous-bandes en association avec des critères perceptuels. D’un côté, il y a des formats propriétaires (AC-3, ATRAC, MUSICAM et PAC par exemple) et de l’autre des standards (MPEG 1,2 et 4). La référence [54] donne une liste exhaustive des principaux codeurs existants. Le schéma de principe de ces différents codeurs se résume toujours au schéma générique de la figure 2.14. Le signal est décomposé dans un premier temps dans le domaine fréquentiel soit par une transformation linéaire unitaire ou par un banc de filtres. Puis une fonction de coût, calculée à partir des coefficients transformés ou en parallèle de la transformation, définit un critère d’importance relative des coefficients transformés pour la distribution des ressources. Cette fonction se base généralement sur des critères psycho-acoustiques. Vient ensuite l’allocation des bits qui distribue les ressources disponibles aux coefficients issus de la transformation suivant leur importance définie par la fonction de coût. La quantification permet alors d’échelonner les coefficients selon l’allocation faite précédemment. Enfin, le codage entropique exploite au travers de codes à longueurs variables les redondances résiduelles intra et inter-trame. L’utilisation d’une quantification vectorielle permet aussi d’exploiter les redondances intra-trame et peut ainsi se passer en partie du codage entropique.

Le codage audio perceptuel qui vient d’être introduit a de bonnes performances pour des débits élevés. Par contre, le codage est fortement limité à bas débit et en particulier pour la parole. Le

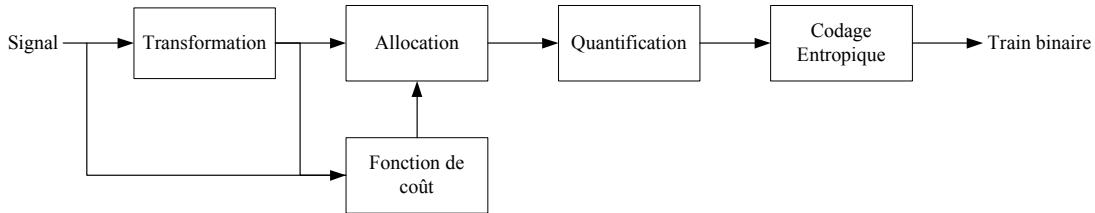


Figure 2.14 Chaîne de codage par transformée.

codage fréquentiel ne prend pas en compte explicitement les structures importantes de la parole (pitch, formants) et représente mal certains sons critiques comme les attaques et les plosives. Ceci est d'autant plus vrai que le codage par transformée utilise de longues trames pour maximiser le gain de codage, ce qui introduit des artefacts tels que du préécho survenant principalement lors d'attaques. Ce défaut correspond à un étalement temporel de l'erreur de codage sur toute la longueur de la trame.

2.4 Codage universel

De nos jours de nouvelles applications émergent avec les dernières générations de téléphones mobiles et la démocratisation de la voix sur IP. Des débits plus élevés sont disponibles pour les communications de la voix, mais avec une demande de plus en plus exigeante sur la qualité de la restitution et sur la polyvalence à coder autre chose que de la parole, comme de la musique ou l'ambiance sonore d'une conversation. Or, par le passé comme les applications visées se différenciaient grandement, deux paradigmes se sont développés pour le codage audio : le codage prédictif pour la parole et le codage par transformée ou en sous-bandes pour des signaux audio de toute nature et plus particulièrement pour la musique. Les codeurs parole travaillent avec des faibles débits mais sont quasi exclusivement destinés au signal parole. Au contraire, les codeurs audio génériques traitent indifféremment tout signal audio avec une qualité quasi constante pour des débits élevés, mais s'adaptent difficilement à de faibles débits. La Figure 2.15 résume les caractéristiques principales des différentes catégories de codage audio.

Un des problèmes les plus étudiés en codage audio de nos jours est alors la recherche d'un format universel de codage à bas débit, c.-à-d. indépendant de la nature du signal d'entrée. Plusieurs approches ont été proposées à cet effet. En partant du codage fréquentiel, le principal problème est de pouvoir traiter les signaux non-stationnaires et transitoires avec de faibles débits. La dé-

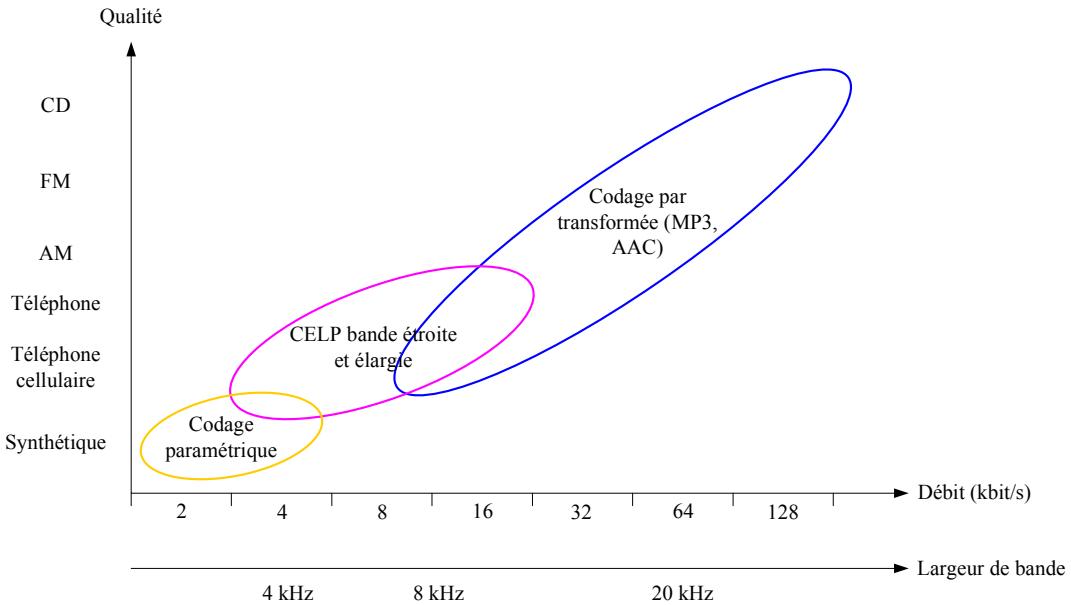


Figure 2.15 Chaîne de codage par transformée.

tection de tels signaux peut permettre au codeur de changer sa résolution temporelle à l'aide de commutation de fenêtre d'analyse (*window switching*) [55] ou bien de banc de filtres [56]. Il est aussi possible d'utiliser un postfiltrage adaptatif [57] de la synthèse ou bien de faire une mise en forme temporelle du bruit (*Temporal Noise Shaping, TNS*) [58]. L'allocation optimale à faible débit crée aussi des problèmes dans le cas de la quantification scalaire où on ne peut attribuer qu'un nombre entier de bits par coefficient. Une répartition fractionnaire des bits est alors possible avec l'utilisation d'une quantification vectorielle [59]. Pour atteindre des débits plus faibles, l'utilisation d'extensions artificielles ou paramétriques de la bande codée (*BandWidth Extension, BWE*) permet par réplication spectrale d'économiser des ressources en estimant une partie du spectre de la synthèse. L'association de la technologie SBR (*Spectral Band Replication*) [60] avec le codeur MPEG AAC (Advanced Audio Coding) [11] fait par le codeur Enhanced-aacPlus [61] permet d'atteindre des débits de 16 à 48 kbit/s pour des signaux stéréo échantillonnés à 48 kHz. Une revue des techniques BWE sera faite dans le chapitre suivant.

En partant des codeurs de parole, le problème est de pouvoir s'affranchir du modèle spécifique de production de la parole ainsi que d'augmenter le débit afin de les rendre plus universels. Les techniques de type CELP deviennent très vite complexes et gourmandes en temps de calcul lorsque le débit augmente, et ce, principalement lors de la détermination de l'excitation innovatrice par

la boucle d'analyse par synthèse. Le codage prédictif par transformée permet de s'affranchir de ce problème en codant directement l'excitation dans le domaine fréquentiel tout en introduisant une technique propre aux codeurs par transformée. Plusieurs codeurs utilisent cette approche, comme les codeurs TCX [62] et les codeurs TPC (*Transform Predictive Coding*) [63]. La technologie TCX initialement introduite pour le codage de la parole en bande étroite à 8 kbit/s, a été étendue pour le traitement de la parole large bande à 16 kbit/s ainsi que pour de la musique à 24 kbit/s [64]. La figure 2.16 montre le principe du TCX avec une prédiction de pitch.

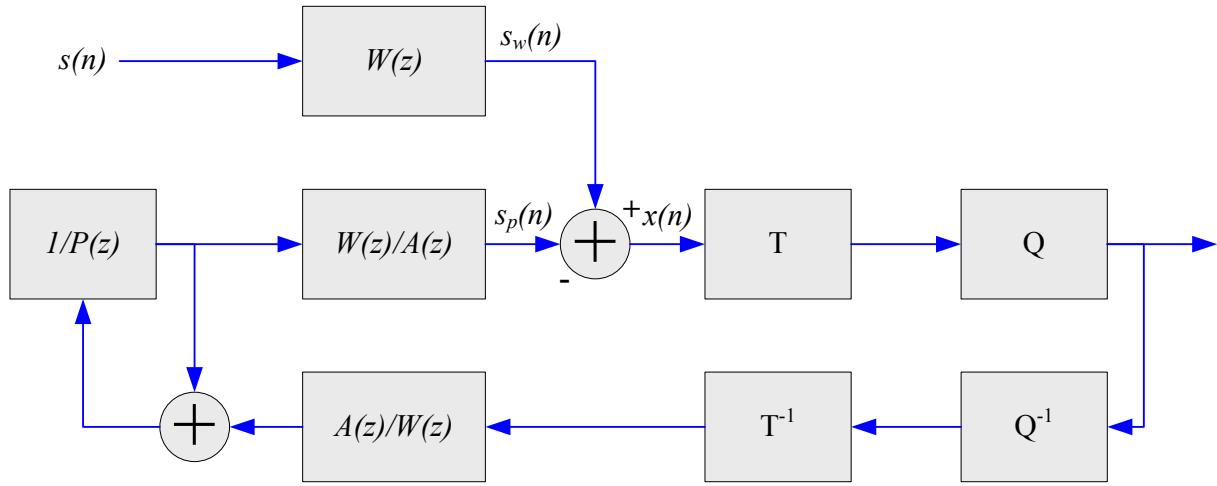


Figure 2.16 Principe du codeur TCX.

L'approche du codage prédictif par transformée est tout de même moins efficace que le CELP à débit comparable pour coder de la parole. Toute approche fréquentielle nécessite une description minimum des phases pour obtenir une bonne cohérence temporelle avec le signal original. Or, ce critère est essentiel en parole. L'utilisation d'un codage multimodal est alors une solution efficace. Le choix du mode de codage peut se faire soit par discrimination du signal d'entrée [65, 66] ou par concurrences des modes en boucle fermée [67]. Une telle solution est utilisée par l'AMR-WB+ [12] qui est recommandé par le 3GPP pour les systèmes de communications mobiles de 3^e génération. Dans ce cas, le choix du codage se fait entre la technologie ACELP et plusieurs modes TCX.

En ce qui nous concerne, l'approche hiérarchique permet aussi de répondre au problème du codage universel. L'imbrication d'un codeur par transformée à la suite d'un codeur parole par exemple permet d'utiliser des méthodes de codage venant de deux paradigmes dans le but qu'elles puissent se compléter mutuellement [68, 69, 70]. De plus, une telle approche permet d'une part un décodage à débit variable et d'autre part de garder une grande interopérabilité avec tous

systèmes utilisant déjà le même codeur de parole. Le chapitre suivant est dédié à ce genre de codage.

2.5 Conclusion

Dans ce chapitre, nous avons dans un premier temps fait un exposé succinct des techniques de codage particulières au signal parole à bas débits. Dans un second temps, nous avons introduit le codage fréquentiel par transformée et par sous-bandes. Le domaine fréquentiel est généralement utilisé par les codeurs audio génériques utilisés à des débits élevés.

Ce tour d'horizon a permis de soulever une problématique importante en codage audio qu'est le codage universel à bas débit. Les solutions proposées sont diversifiées et peuvent venir d'une extension soit d'un codeur de parole ou d'un codeur audio générique. Le codage hiérarchique pouvant combiner des techniques venant des deux paradigmes est aussi une solution potentielle. C'est une des facettes qui font de ce codage une technique attrayante. Tout au long de la thèse, nous allons essayer de proposer des techniques de codage hiérarchique répondant au mieux au critère d'universalité.

CHAPITRE 3

Codage Audio Hiérarchique

La hiérarchisation de l'information lors du codage est de plus en plus demandée par les applications de communication. Le codage hiérarchique peut se ramener la plupart du temps à une mise en cascade de deux codeurs ou plus. Chaque codeur représente une couche du codage hiérarchique. La première couche, appelée couche de base, code le signal original alors que les suivantes, les couches d'amélioration, codent l'erreur de codage à la sortie de la couche précédente. La qualité globale de la synthèse est ainsi augmentée pour un surplus de débit à transmettre. Le train binaire composé des sous-flux des différentes couches est alors qualifié d'encastré. Le schéma d'un système simple à deux couches est donné à la Figure 3.1. Deux qualités de décodage peuvent être obtenues \hat{x}_1 et \hat{x}_2 pour deux débits différents. La première qualité de synthèse est obtenue grâce seulement au sous-flux 1 provenant du codeur 1. Pour un débit plus élevé lors de la prise en compte du sous-flux 2, la qualité de la restitution sonore augmente à l'aide de l'amélioration apportée par le second codage. Ce principe est connu sous le nom de codage à raffinements successifs. Viennent à cela s'ajouter les avantages intrinsèques du codage hiérarchique abordés lors de l'introduction.

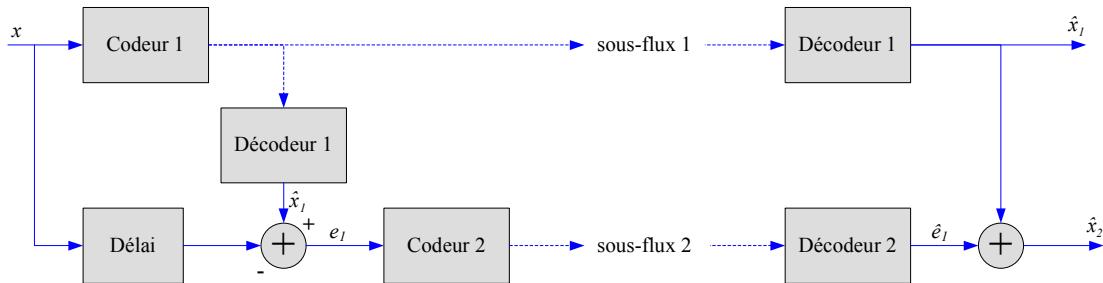


Figure 3.1 Codeur/Décodeur (Codec) à deux couches.

Toutefois, la hiérarchisation de l'information est une contrainte très forte sur le codage. Le codage hiérarchique a pour cette raison des performances généralement en retrait par rapport au codage classique. Cette baisse de performance vient de la coopération restreinte entre les couches du codage hiérarchique. Les données provenant d'une couche doivent être décodées indépendamment de celles des couches supérieures. Une optimisation globale n'est donc pas possible. L'objectif est

alors de se rapprocher des performances des codeurs classiques à débit fixe pour tous les débits intermédiaires du codage hiérarchique. La Figure 3.2 illustre les compromis pouvant être faits. Soit on privilégie les performances à bas débit comme dans le cas du codeur A, soit au contraire on privilégie les débits élevés comme pour le codeur B.

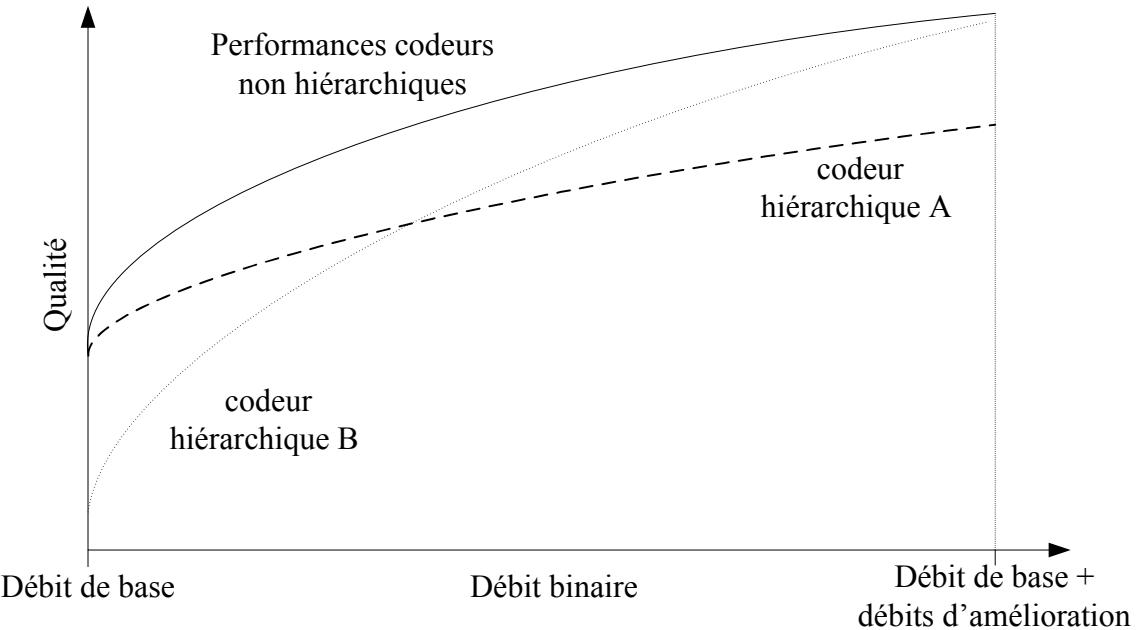


Figure 3.2 Compromis entre la qualité du codeur de base et celle des couches d'amélioration.

Il existe de nombreuses stratégies de hiérarchisation, donc de nombreuses solutions possibles. On s'intéresse principalement aux codeurs pouvant coder convenablement de la parole à bas débit, c'est à dire pouvant prétendre à être universel. De tels codeurs sont essentiellement à base d'un codeur dédié parole, c'est pourquoi une majorité des codeurs présentés ici utilisent une technique CELP pour le codage de base.

3.1 Amélioration graduelle de la description

3.1.1 CELP imbriqué

Les différents codeurs à prédiction linéaire se distinguent principalement par le codage de l'excitation (multi-impulsions, CELP, RELP...). Dymarski et al. [71] ont proposé une méthode générale permettant d'utiliser tous types de modélisation de l'excitation au sein d'un même codeur. Le principe est d'utiliser plusieurs dictionnaires afin de calculer une excitation optimale qui sera

finalement une combinaison linéaire de différents code-vecteurs. Plusieurs travaux [71, 72, 73] proposent que l'ensemble des code-vecteurs soient orthogonaux entre eux pour que la modélisation soit optimale. Ils utilisent pour ce faire des variantes de l'algorithme de Gram-Schmidt afin d'orthogonaliser à chaque étape les code-vecteurs où le signal sera projeté. Ceci revient à calculer une base orthogonale récursivement. Ainsi le prochain code-vecteur codé sera orthogonal aux autres code-vecteurs déjà utilisés pour modéliser l'excitation. On obtient ainsi une recherche multiétages de l'excitation du signal qui, due à la récursivité, s'adapte particulièrement bien à une structure de codage hiérarchique.

On retrouve ainsi plusieurs codeurs CELP imbriqués dans la littérature. Dans [74] trois dictionnaires innovateurs sont associés de façon séquentielle. La Figure 3.3 schématisé le principe pour le cas générale où k dictionnaires innovateurs sont utilisés. Dans [75] l'utilisation d'un codage MP-CELP (*Multi-Pulse-based CELP*) permet de rendre facilement les différentes excitations orthogonales. Ce schéma de codage a été retenu pour un des codeurs CELP du standard MPEG-4 [76]. Il est possible aussi d'utiliser une décomposition de l'excitation en plusieurs composantes orthogonales avant de coder chacune d'entre elles de façon indépendante. Les travaux de [77] utilisent une décomposition pyramidale. De ce point de vue, le codage prédictif par transformée (TCX, TPC) sont des codeurs CELP imbriqués dont la décomposition de l'excitation n'est ni adaptable ni optimisée pour une transmission graduelle de la description.

3.1.2 CELP associé au codage par transformée

Le codage CELP est surtout dédié au codage de la parole. Pour coder des signaux plus généraux, même en codant le signal en large bande, les résultats sont pour la plupart du temps décevants. Le codage par transformée est quant à lui plus adapté à ce type de signaux et peut être ainsi combiné à un codage CELP [68, 69, 70] pour coder aussi bien de la parole à faible débit que de la musique pour des débits plus élevés. De plus, les coefficients transformés du codage par transformée peuvent être eux-mêmes envoyés graduellement [70], ce qui permet d'obtenir une granularité plus fine du raffinement. La Figure 3.4 schématisé le principe de l'association des deux techniques de codage.

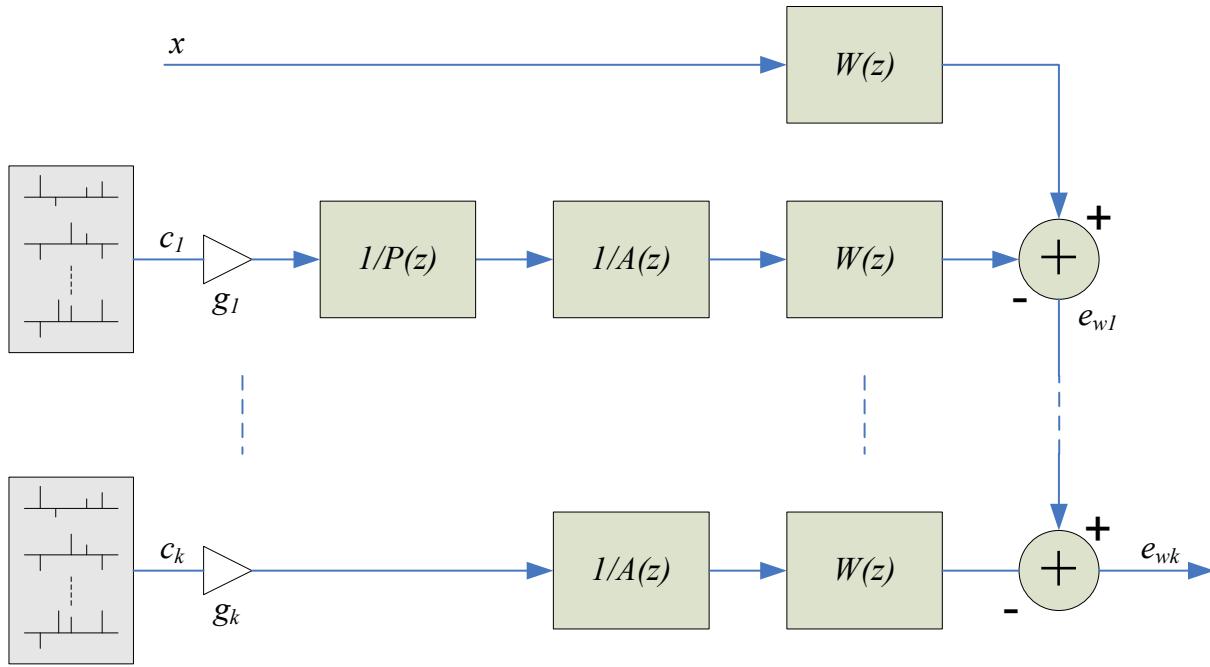


Figure 3.3 Codage CELP imbriqué avec k dictionnaires innovateurs.

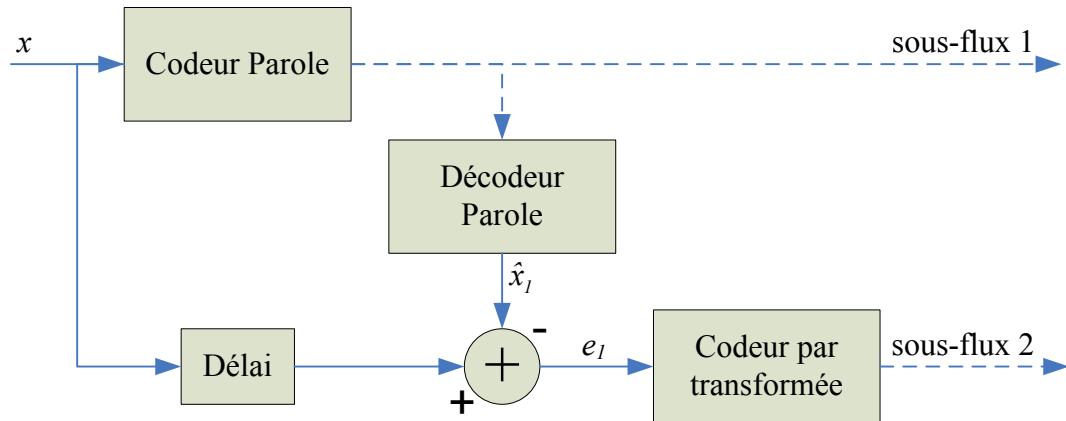


Figure 3.4 Association d'un codage parole avec un codeur par transformée.

3.1.3 Codage prédictif par transformée

Comme on l'a vu précédemment, le codage prédictif par transformée peut être vu comme un cas particulier du codage CELP imbriqué. Si les coefficients transformés sont décrits et envoyés graduellement, alors on obtient un codage hiérarchique. Dans [78], un codage prédictif par transformée utilise plusieurs modes selon le signal d'entrée afin de faire varier certains paramètres de codage et d'adapter l'ordre de transmission des sous-flux du train binaire à la source.

3.2 Extension graduelle de la bande

Bien que l'amélioration de la modélisation de l'excitation augmente la qualité de la synthèse, celle-ci peut être limitée par la largeur de bande codée. Les codeurs de parole traitent en général le signal sur la bande téléphonique de 300 à 3400 Hz. Bien que cette bande soit suffisante pour la bonne compréhension du message, l'élargissement de la bande codée de 50 à 7000 Hz permet de rendre la parole reconstituée plus naturelle avec une qualité se rapprochant d'une communication face à face [79]. De plus, le codage de la bande élargie est indispensable pour prétendre transmettre de la musique. Pour s'approcher d'une qualité haute-fidélité pour des signaux musicaux, la largeur de bande codée peut s'étendre jusqu'à 20 kHz.

Beaucoup de travaux se sont penchés sur la question d'étendre un codeur de parole bande étroite en un codeur large bande. Cette extension graduelle de la bande codée est utilisée généralement pour pouvoir accroître les capacités des standards conventionnels déjà bien implantés [6, 80, 69, 81, 82, 83, 84, 85, 86, 70, 87]. Il existe aussi des travaux proposant leur propre schéma de codage hiérarchique intégrant d'origine une extension graduelle de la bande codée [88, 75, 89].

Deux types d'approches sont possibles : une extension par un codage en sous-bandes ou bien une extension par un codage en pleine bande. Une extension par codage en sous-bandes permet de décomposer le signal original en deux ou plusieurs sous-bandes par un banc de filtres. La bande de base, souvent regroupant les basses fréquences, est traitée par un codeur à bande limitée fournissant une première qualité de restitution. La description des autres fréquences, regroupées en une ou plusieurs bandes manquantes, permet l'extension de la largeur de bande de la synthèse. Par exemple, l'utilisation d'un banc de filtres à deux canaux de type QMF permet une décomposition simple en deux sous-bandes très adaptée pour l'extension en bande élargie

d'un codeur bande étroite [6, 80, 90, 87]. La Figure 3.5 donne le schéma de principe d'une telle extension.

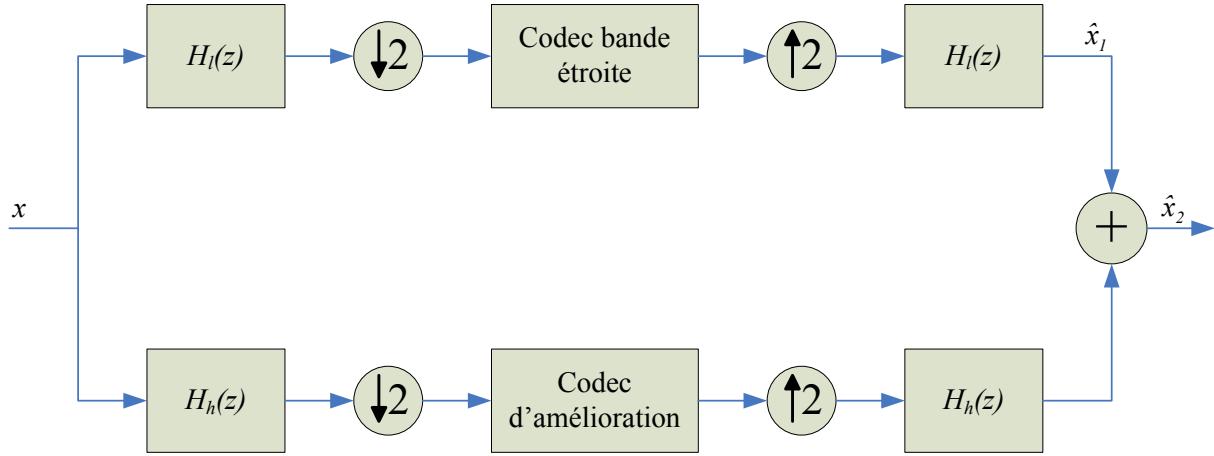


Figure 3.5 Extension d'un codeur bande étroite en bande élargie par une décomposition en deux sous-bandes.

Dans le cas de l'extension en pleine bande, la synthèse à bande limitée est suréchantillonnée et enrichie afin d'éteindre sa largeur de bande [75, 69, 82, 83, 70]. La Figure 3.6 montre un exemple d'extension de bande où le codeur d'amélioration prend en compte la différence entre le signal original et la synthèse sur-échantillonnée d'un codeur bande étroite. Le signal différence à décrire est alors composé de deux informations distinctes. La première partie de l'information, correspondant aux fréquences déjà traitées par le codeur bande étroite, regroupe une information permettant un raffinement de la qualité en bande étroite. La seconde partie correspondant aux fréquences manquantes du spectre, regroupe les composantes originales du signal d'entrée, qui par leur description permettront réellement l'extension de la bande codée.

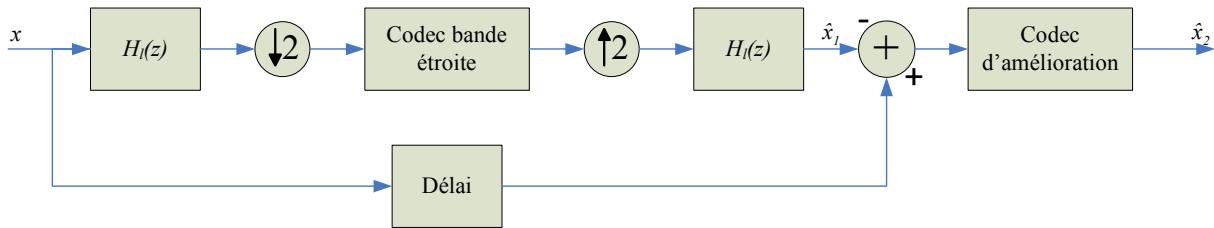


Figure 3.6 Extension en pleine bande d'un codeur bande étroite.

Comme dans le cas de l'amélioration graduelle de la description de la bande déjà codée, l'extension de la bande peut se faire aussi bien dans le domaine temporel que dans le domaine fréquentiel. De plus que ce soit dans l'un ou l'autre des deux domaines, ces dernières années ont vu l'émergence d'extensions de bande codée hautement paramétriques voire complètement artificielles. On nomme ces dernières techniques simplement *BandWith Extensions* (BWE).

3.2.1 Extension par codage prédictif

L'utilisation d'un même type de codage pour la bande de base et la ou les bandes d'amélioration est une méthode évidente, mais surtout attrayante pour sa simplicité et la possible réutilisation des modules de codage. Néanmoins, la bande de base nécessite souvent une description plus riche que les bandes d'amélioration, ce qui impose une certaine adaptation du codage. L'association d'un QMF avec un codage ADPCM dans chacune des deux sous-bandes est à la base de la norme G.722 [6]. La généralisation d'un tel codage pour plus de deux sous-bandes permet une extension graduelle de la bande avec une plus fine granularité [88]. Le codage CELP peut être aussi employé par une extension de bande que ce soit dans une structure en sous-bandes [80], ou bien dans une structure pleine bande [75, 82].

3.2.2 Extension par codage par transformée

Le codage par transformée peut, comme on l'a vu précédemment, se faire dans le domaine de l'excitation après une prédiction ou bien directement dans le domaine du signal. Dans tous les cas, il peut facilement s'intégrer dans une extension en sous-bandes [87], ou bien en pleine bande [69, 70]. Le codage par transformée est alors bien adapté pour sélectionner et ordonner les composantes fréquentielles lors de la transmission afin d'obtenir une extension finement graduelle et très flexible [70].

3.2.3 Extension artificielle ou hautement paramétrique

L'extension de bande artificielle ou hautement paramétrique consiste à enrichir le spectre de la synthèse en élargissant la bande synthétisée en ne transmettant pas ou peu d'information supplémentaire. Contrairement aux techniques classiques de codage d'onde de forme ou hybrides, on ne décrit pas explicitement le contenu des hautes fréquences. A contrario, en utilisant les propriétés psycho-acoustiques des signaux audio ainsi qu'en exploitant la forte corrélation entre

leurs différentes bandes, les fréquences manquantes sont estimées à partir de celles déjà transmises. Cela nécessite tout de même qu’au moins une partie du spectre soit transmise de façon classique. L’estimation de la bande manquante se fait au niveau du décodeur ce qui a pour effet d’alléger le codage mais d’alourdir le décodage. Comme pour le codage prédictif, le problème est généralement découpé en deux par la modélisation de deux composantes décorrélées : l’enveloppe spectrale et l’excitation. Le principe de l’extension peut être alors schématisé par la Figure 3.7.

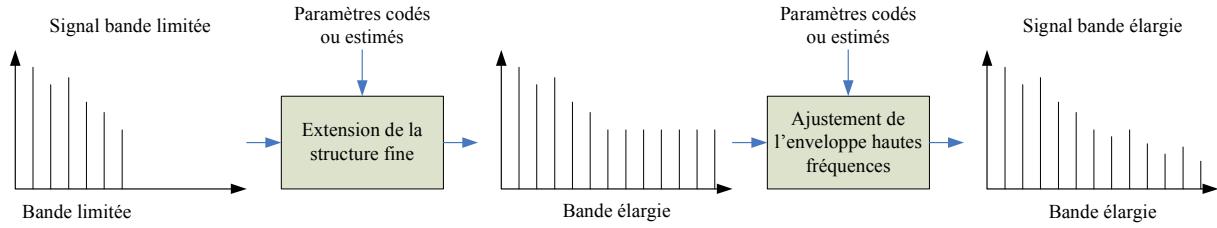


Figure 3.7 Principe de l’extension de bande artificielle ou paramétrique.

Modélisation de l’enveloppe spectrale

L’enveloppe spectrale est sûrement la composante spectrale la plus importante perceptuellement, surtout dans les hautes fréquences. Son estimation est un sujet très étudié ces dernières années [91, 92, 93, 94, 95]. Les méthodes d’estimation sont basées sur les dépendances statistiques mutuelles entre l’enveloppe de la bande de base et celle de la bande manquante. Ces méthodes sont généralement basées sur des mises en correspondance de vecteurs issus de dictionnaires associatifs (*codebook mapping*) [91, 92] ou bien des méthodes de recouvrance statistique (*statistic recovery*) [93, 94, 95]. Elles nécessitent alors un processus d’apprentissage des modèles. D’un point de vue théorique, les deux bandes partagent une information mutuelle au niveau de l’enveloppe [96, 97]. Cependant, cette information mutuelle est assez faible [98]. Ainsi, ces méthodes d’estimation sans transmission d’informations sont limitées en qualité.

L’envoi d’information supplémentaire, si les conditions le permettent, offre de meilleurs résultats. Il est possible alors d’envoyer quelques bits afin de raffiner l’estimation [83, 98]. Mais la solution la plus répandue consiste à envoyer directement sous forme paramétrique l’enveloppe spectrale de la bande manquante. L’utilisation d’un filtre de synthèse tout-pôle est la solution la plus commune dans le domaine temporel [81, 12]. Il est aussi possible de quantifier directement l’énergie par sous-bande dans le domaine fréquentiel [60].

Extension de la structure fine

Comme pour l'enveloppe spectrale, il existe des méthodes utilisant uniquement une simple estimation au décodage, et d'autres qui envoient de l'information supplémentaire. Partant de la représentation simpliste de la production de la parole, il est possible de modéliser l'excitation par un peigne d'harmonicité, par du bruit ou bien par une mixture des deux. Cette paramétrisation des signaux de parole a donné naissance à plusieurs techniques d'extension de la structure fine [92, 99, 94]. Les paramètres de la bande manquante peuvent être transmis ou bien simplement estimés à partir de la bande de base. Par exemple, l'utilisation de techniques de translation de l'excitation permet de faire hériter à la bande manquante les caractéristiques de la bande de base. La méthode la plus utilisée est la technique de repliement spectral [100] simple à mettre en œuvre que ce soit dans le domaine temporel ou fréquentiel. L'extension de bande par distorsion non linéaire [101] permet aussi de générer de manière simple des harmoniques ou du bruit dans la bande manquante. Dans tous les cas et lorsque c'est possible, l'envoi d'information par l'intermédiaire de paramètres de contrôle (une mesure de tonalité, un plancher de bruit, de sinusoïdes isolées ou bien de gains) permet de réajuster l'estimation [60].

L'extension paramétrique de la bande est de plus en plus présente dans les codeurs modernes. Elle donne de bons résultats pour l'extension d'un signal bande étroite en bande élargie, mais se justifie encore plus pour l'extension d'une synthèse bande élargie en bande FM (jusqu'à 16 kHz) voire CD (jusqu'à 20 kHz). Néanmoins, comme c'est une technique faisant beaucoup de suppositions, et au vu de la grande diversité des signaux audio, elle n'est pas aussi robuste qu'un codage classique de la bande manquante. Les artefacts introduits sont plus ou moins marqués selon l'adéquation de l'extension avec la nature du signal.

3.3 Conclusion

On a vu que dans le contexte actuel se dégage la nécessité de développer un codage universel et hiérarchique. Plusieurs modèles ont été présentés dans ce chapitre. Les codeurs CELP imbriqués, bien qu'intéressants pour certaines applications, sont trop spécifiques à la parole, et ne permettent pas un traitement efficace des autres signaux audio. Les techniques associant le codage de parole avec le codage par transformée sont très performantes à condition d'avoir un débit supplémentaire assez important. Pour l'extension de bande il est possible d'économiser du débit en utilisant une extension hautement paramétrique et d'allouer une partie des bits sauvés dans l'amélioration de la bande de base. C'est l'approche utilisée pour les codeurs mixtes parole-musique non hiérarchique

de dernière génération, comme l'Enhanced aacPlus [13] et l'AMR-WB+ [12]. Cette approche n'est toutefois pas transposable pour toute extension de bande. La bande de fréquence jusqu'à 7 kHz, nécessite tout de même une bonne description pour obtenir une bonne qualité de restitution.

Pour notre travail de thèse, on a retenu la technique d'association d'un codage CELP avec un codage par transformée. L'objectif est d'augmenter la qualité de la parole pour les débits supérieurs à celui du codeur de parole, mais surtout d'améliorer significativement la qualité de la musique synthétisée. Pour la structure à base du codeur bande étroite G.729 où une extension large bande est prévue, l'utilisation d'une technique d'extension paramétrique de la bande semble être intéressante pour y parvenir. Il faudra quand même s'assurer qu'elle soit suffisante pour décrire fidèlement la bande de 3400 à 7000 Hz, qui reste perceptuellement très importante surtout pour la musique.

CHAPITRE 4

Posttraitement Fréquentiel pour Codeurs de Parole

4.1 Introduction

Bien que le modèle source-filtre utilisé par les codeurs de parole leur permette de maintenir une bonne qualité pour le codage de la parole à bas débit, leur structure spécifique pénalise leur performance lors du traitement de signaux autres que de la parole. On a vu dans le chapitre 2 que majoritairement les codeurs de parole se basent sur un modèle de production du son propre à la parole qui leur permet d'atteindre des débits relativement faibles entre 2 et 20 kbit/s. Ce modèle est trop restrictif pour que les codeurs de parole puissent bien traiter des sons d'autre nature.

Nous proposons alors de traiter la synthèse issue d'un codeur de parole par un posttraitement pour en faire un codeur mixte de parole et de musique. Notre posttraitement nécessite la transmission d'informations supplémentaires ce qui engendre un léger surplus de débit. Il fait donc office de couche d'amélioration formant avec le codeur de parole, un codeur hiérarchique à train binaire encastré.

Nous nous proposons dans un premier temps de caractériser les principales défaillances d'un codeur parole faisant état de l'art, spécifiquement l'AMR-WB, lors du traitement de signaux musicaux. Puis, une fois les détériorations caractérisées, nous exposons le posttraitement d'amélioration, solution issue des observations.

4.2 Caractéristiques du signal de synthèse de l'AMR-WB

L'AMR-WB est un standard de codage de parole large bande de l'organisation ETSI/3GPP, qui a été aussi adopté par l'ITU-T sous la recommandation G.722.2 [7]. L'AMR-WB est basé sur la technologie ACELP et appartient donc à la classe des codeurs à prédictions et à analyse par synthèse. Il comprend 9 modes, chacun étant optimisé pour opérer à un débit différent allant de 6.6 à 23.85 kbit/s. Il code le signal en large bande en divisant la bande de 50 à 7000 Hz en

une bande basse de 50 à 64000 Hz et en une bande haute de 6400 à 7000 Hz. La bande basse fait l'objet du codage ACELP en manipulant le signal échantillonné à 12.8 kHz. À l'opposé, la bande haute de la synthèse est simplement une modélisation hautement paramétrique. Dans ce qui suit, nous nous intéressons uniquement à améliorer le codage dans la bande de 50 à 6400 Hz. Ainsi, le posttraitement proposé s'effectue à la suite du codage ACELP sur la synthèse échantillonnée à 12.8 kHz. L'extension de la bande à 7000 Hz se fait pour le moment de la même façon que l'AMR-WB. L'utilisation d'une extension plus sophistiquée comme celles entrevues à la section 3.2 pourra s'affranchir des 7 kHz et aller au-delà.

L'AMR-WB, comme la majorité des codeurs de parole à bas débit, éprouve énormément de difficultés à coder des signaux autres que la parole, ou du moins qui ne répondent pas au modèle source-filtre de production de la parole. Ces signaux ne sont alors ni du bruit, ni monoharmonique, ni une mixture des deux. La synthèse de tels signaux par l'AMR-WB comporte alors plusieurs artefacts plus ou moins gênants à l'oreille.

Pour illustrer les difficultés rencontrées par l'AMR-WB, nous allons étudier le signal synthétisé en utilisant le mode à 12.65 kbit/s lorsque le signal d'entrée est de l'orgue. Le mode à 12.65 kbit/s est privilégié dans cette étude, car il est le mode le plus populaire et sûrement le plus performant selon le rapport qualité/débit. Le signal d'orgue est quant à lui connu pour être difficile à coder avec des codeurs de parole. Ses caractéristiques multiharmoniques sont assez éloignées de celles de la parole. La Figure 4.1 compare le spectre d'amplitude du signal original avec celui sa synthèse, tous les deux moyennés sur une période de quelques secondes. La comparaison révèle que l'AMR-WB échoue d'une part à modéliser certains pics spectraux en haute fréquence. D'autre part, l'AMR-WB injecte entre les pics, c'est-à-dire dans les vallées du spectre, un bruit beaucoup trop élevé. C'est ce dernier défaut qui est le plus prononcé et qui a le plus de répercussions sur la qualité globale de la synthèse.

Pour analyser la source du problème, nous avons étudié l'erreur de codage après la seule contribution du dictionnaire adaptatif, et après l'ajout de celle du dictionnaire innovateur. Comme le montre la Figure 4.2, le dictionnaire innovateur diminue l'erreur surtout dans les basses fréquences, et dans une moindre mesure au niveau des pics spectraux des hautes fréquences. Par contre, il augmente l'erreur de codage au niveau des vallées à partir de la fréquence aussi basse que 2000 Hz. Cela montre que le dictionnaire innovateur n'est pas adapté ou suffisant pour coder ce type de signal. Cette inadaptation vient du dictionnaire lui-même, mais aussi des étages

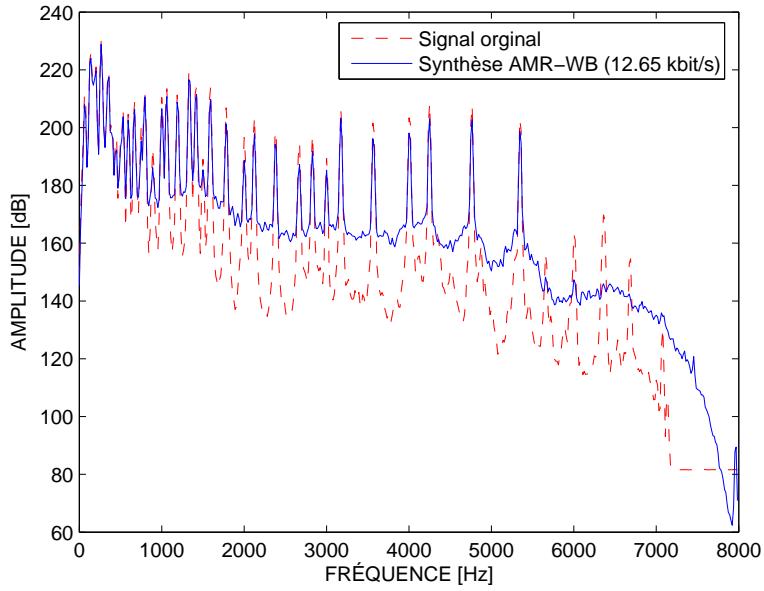


Figure 4.1 Comparaison du spectre d'amplitude d'un signal d'orgue original avec celui de sa synthèse par l'AMR-WB.

précédents. En effet, le dictionnaire innovateur met plus d'emphase sur les basses fréquences, alors que des sons comme l'orgue ont aussi de l'information importante et riche dans les hautes fréquences. D'autre part, la prédiction à long terme est adaptée spécifiquement pour les signaux monoharmoniques, ce qui n'est pas représentatif de la majorité des signaux musicaux. De plus, ces mêmes sons sont souvent plus stationnaires que la parole. Ils ont intérêt à être analysés sur une fenêtre plus grande et avec des ordres de prédiction à court terme plus élevés. Le résidu de la prédiction à court terme du codeur de parole montre alors une forte énergie, et la corrélation résiduelle intersymbole est encore élevée. Pour finir, le modèle tout-pôle du filtre de synthèse de la plupart des codeurs de parole peut être aussi limitatif. Un modèle plus sophistiqué, par exemple un filtre de synthèse à phase minimum obtenu par une analyse cepstrale, pourrait être plus adéquat pour des sons complexes.

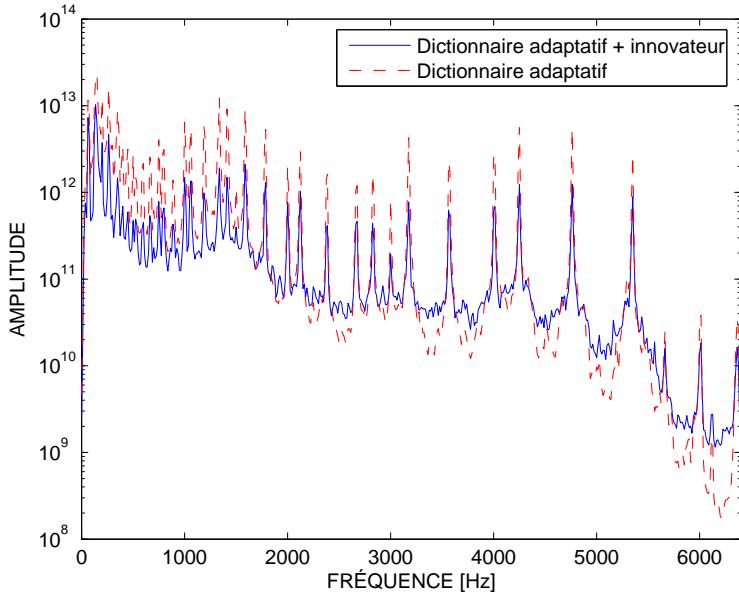


Figure 4.2 Erreurs de codage avec une excitation uniquement adaptative et une excitation adaptative et innovatrice.

4.3 Posttraitements fréquentiel

4.3.1 Principe

Considérant les observations précédentes, on propose un posttraitements de la synthèse issue de l'AMR-WB afin d'en améliorer sa qualité de restitution tout au moins lors du traitement des signaux autres que de la parole. Le schéma de principe de la Figure 4.3 donne une vision globale du rôle du posttraitements. Ce dernier forme avec le codeur de parole une structure hiérarchique. Le codeur de parole, dans notre cas l'AMR-WB, produit une première synthèse \hat{x}_{base} en transmettant un sous-flux de base. Le posttraitements est alors appliqué au décodeur à la synthèse de base en utilisant l'information supplémentaire provenant du sous-flux d'amélioration, calculé et transmis par le codeur. Le signal traité \hat{x}_{enh} est alors une version améliorée de la synthèse \hat{x}_{base} .

L'objectif du posttraitements au décodage est alors de baisser l'énergie du bruit introduit par le codeur de parole dans certaines zones du spectre et plus spécifiquement dans les vallées. La façon la plus directe pour y arriver est de transformer le signal dans un domaine spectral et de multiplier les coefficients spectraux de la synthèse \hat{x}_{base} de trop forte amplitude par un

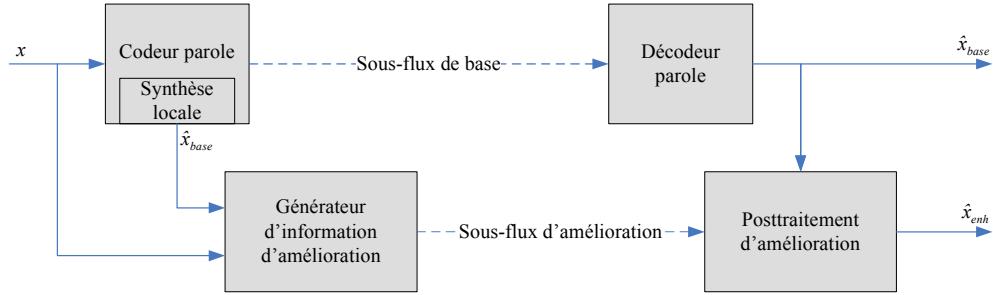
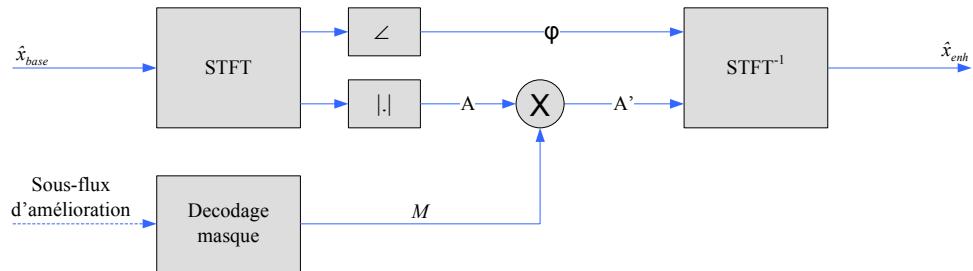


Figure 4.3 Diagramme de haut niveau du système proposé.

coefficient multiplicateur inférieur à l'unité. Les coefficients multiplicateurs sont regroupés sous forme d'un signal M , appelé masque, comme illustré à la Figure 4.4. La décomposition se fait grâce à une transformée de Fourier fenêtrée ($STFT$), permettant un compromis entre la résolution fréquentielle et temporelle. La fenêtre est le demi-cycle positif d'un sinus avec un recouvrement de 50% avec la fenêtre précédente. Un tel fenêtrage assure à l'analyse une bonne sélectivité en fréquence, et permet de circonscrire dans le domaine fréquentiel les répercussions dues à la modification d'un coefficient de la transformée par le posttraitements. Le posttraitements est alors précis et efficace. En outre, le recouvrement réduit les effets de bloc d'une trame à l'autre après le posttraitements fréquentiel. L'opération inverse $STFT^{-1}$ comprend l'opération d'addition dans le domaine temporel des coefficients fenêtrés de deux trames successives.

Figure 4.4 Le post-traitements est appliqué à la sortie \hat{x}_{base} du décodeur parole.

Le masque M est généré au codeur comme le montre la Figure 4.5. À la suite d'une comparaison dans le domaine fréquentiel entre l'erreur de codage e_{base} et le signal original x , le masque M est construit. Chaque composante du signal M correspond alors à une fréquence discrète de la transformée, et sa valeur dépend du résultat de la comparaison entre les amplitudes des coefficients correspondants des deux signaux. Le signal est ensuite codé sous forme d'un train binaire appelé

alors sous-flux d'amélioration. L'objectif est de transmettre le masque M au plus faible débit possible. La section 4.4 portera sur l'étude de ce codage.

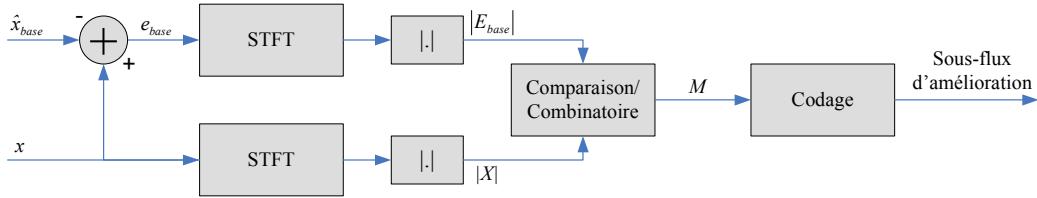


Figure 4.5 Génération du masque M et du sous-flux d'amélioration au codeur.

Comme tout système à raffinements successifs, l'objectif est de réduire l'amplitude de l'erreur de codage $|e_{base}|$. Le codage de e_{base} permet d'y arriver, mais il est souvent trop coûteux en terme de débit, et trop complexe pour être intégré simplement et rapidement à un système de communication existant. De plus, le codage explicite et direct nécessite un minimum de bits pour être efficace. Notre solution se veut au contraire simple et peu onéreuse, tout en ayant une amélioration franche lors du traitement de signaux musicaux. C'est pour ces raisons qu'on la voit comme un posttraitement. Le posttraitement proposé ici consiste alors à mettre à zéro la valeur de M si l'amplitude de l'erreur $|E_{base}|$ est supérieure à celle de $|X|$, et de la mettre à 1 dans le cas contraire.

$$M(k) = \begin{cases} 0 & \text{si } |E_{base}(k)| > |X(k)| \\ 1 & \text{sinon} \end{cases} \quad (4.1)$$

L'entièvre opération du posttraitement peut être alors résumée par une mise à zéro forcée de certaines composantes spectrales dans le cas où l'erreur de codage est supérieure au signal original.

$$\hat{X}_{enh}(k) = \begin{cases} 0 & \text{si } |E_{base}(k)| > |X(k)| \\ \hat{X}_{base}(k) & \text{sinon} \end{cases} \quad (4.2)$$

4.3.2 Caractéristiques du posttraitement

Le posttraitement a deux conséquences sur la synthèse audio. La première est de baisser l'amplitude de la synthèse dans certaines parties du spectre comme le montre la Figure 4.6 avec le spectre moyen du signal d'orgue posttraité. On note que le posttraitement dans le cas de l'AMR-WB ne s'applique que pour les premiers 6400 Hz tel que spécifié précédemment. Le spectre posttraité est beaucoup plus creusé au niveau des vallées après la mise à zéro forcée de certaines de ses

composantes. La deuxième conséquence du posttraitement est de diminuer l'erreur de codage comme illustré par la représentation vectorielle des composantes spectrales de la Figure 4.7. Le posttraitement ne fait pas seulement rapprocher le vecteur \hat{X}_{enh} de l'original X par rapport à \hat{X}_{base} , mais diminue aussi l'amplitude de l'erreur $|E_{enh}|$ par rapport à $|E_{base}|$. Le posttraitement code ainsi au travers du masque M une partie de l'information de l'erreur e_{base} . La philosophie du posttraitement peut ainsi se résumer par, plutôt ne rien transmettre que transmettre une information trop erronée.

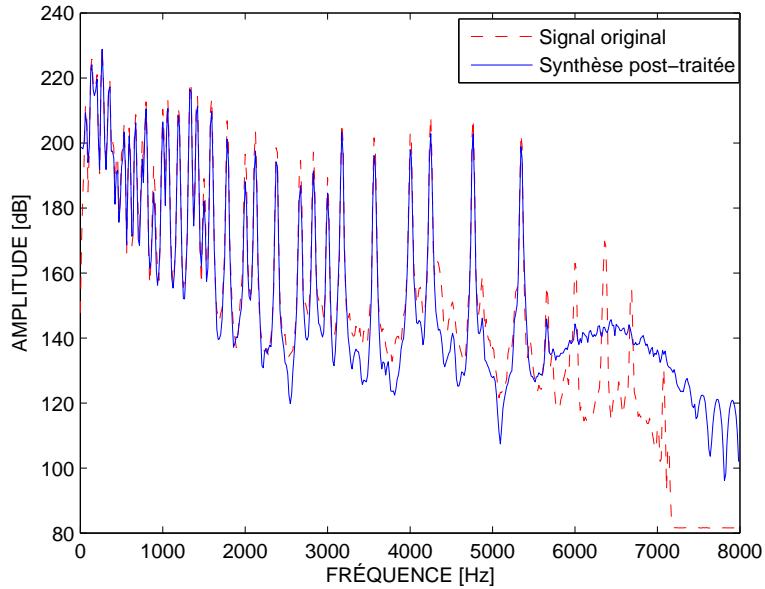


Figure 4.6 Signal original d'orgue et sa synthèse \hat{x}_{enh} après le post-traitement fréquentiel.

La performance du posttraitement dépend grandement de la taille de la fenêtre d'analyse lors de la décomposition fréquentielle. La taille de la transformée de Fourier est en effet un compromis entre la résolution temporelle et fréquentielle. Une bonne résolution fréquentielle permet un posttraitement fréquentiel précis et efficace. De longues fenêtres d'analyse, 40 ms (STFT à 512 points) voire 80 ms (STFT à 1024 points), sont alors très performantes pour des signaux de type musical, fortement stationnaires. En règle générale, le posttraitement apporte à ces signaux un gain en qualité très significatif.

Par contre pour des signaux à fortes transitoires, comme des castagnettes, ou des signaux peu stationnaires, comme de la parole, la précision temporelle est aussi essentielle. De longues fenêtres d'analyse produisent alors des artefacts de type préécho pouvant détériorer la qualité globale.

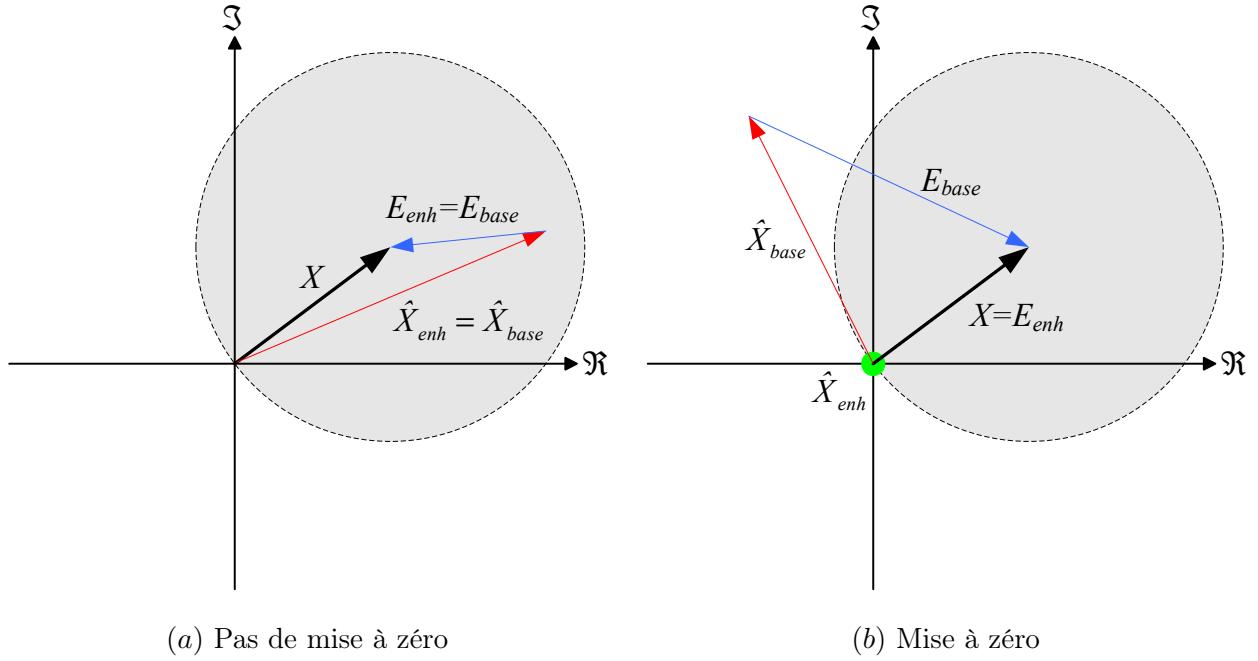


Figure 4.7 Illustration du traitement pour une composante fréquentielle : (a) pas de mise à zéro, (b) mise à zéro.

L'utilisation de fenêtres plus courtes de 20 ms (STFT à 256 points) à 40 ms (STFT à 512 points), permet de résoudre une partie du problème, mais rend le posttraitement moins efficace dans les parties plus stationnaires du signal. En effet, une résolution fréquentielle très faible rend la mise à zéro fréquentielle trop grossière et la synthèse trop synthétique. Pour ce genre de signaux, l'avantage du posttraitement est plus mitigé. De plus, le codeur de parole fournit déjà une très bonne qualité de restitution pour de la parole. Finalement, le meilleur compromis trouvé quelle que soit la nature du signal d'entrée est l'utilisation de trames de 40 ms, ce qui sera utilisé pour le reste du chapitre.

La Figure 4.8 permet de dresser certaines caractéristiques du posttraitement. Le graphique représente la densité des trames d'un signal mélangeant des séquences parlées et séquences musicales dans le plan du rapport d'énergies après et avant posttraitement en fonction du nombre de composantes inchangées. La première constatation est que le nombre de mises à zéros est important. Même dans le cas de la parole, presque 50% des coefficients sont mis à zéro par le posttraitement. Pour des signaux plus musicaux, comme l'orgue par exemple, le taux est plus important et approche les 65%. La deuxième constatation est que, malgré la forte mise à zéro, l'énergie du signal posttraité est peu diminuée. Ceci confirme que les mises à zéros ont lieu principalement

sur les coefficients du spectre à faible amplitude, c'est-à-dire dans les vallées. En moyenne le posttraitemet conserve environ 95% de l'énergie de la trame.

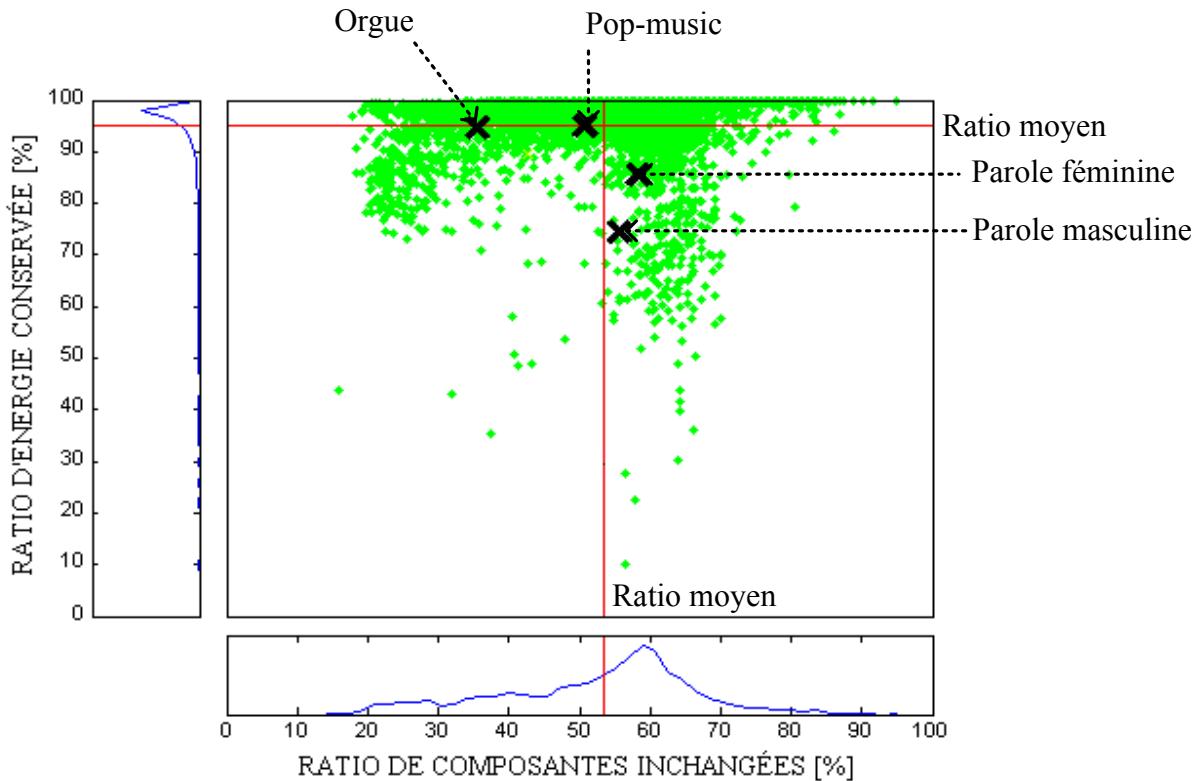


Figure 4.8 Répercussions du post-traitement sur l'énergie de la synthèse en fonction du nombre de composantes mise à zéro pour chacune des trames d'un signal audio constitué de paroles et de musiques.

Nous verrons dans la section 4.5 les résultats d'une évaluation subjective. Cependant, le rapport signal à bruit segmenté (segSNR) permet de donner un aperçu des améliorations apportées par le posttraitemet, comme le montre le Tableau 4.1. Les mesures ont été faites sur plusieurs type de sons : de la parole féminine en français, de la parole masculine en allemand, de l'orgue, de la musique pop et enfin une longue séquence mélangeant divers genres musicaux ainsi que de la parole. Il est à noter que le segSNR ne peut qu'augmenter du fait que le posttraitemet limite l'énergie de l'erreur de codage.

Signal	AMR-WB 12.65 kbits	AMR-WB + posttraitement
Parole féminine Fr.	9.25	9.80
Parole masculine All.	8.98	9.38
Orgue	8.81	10.05
Pop-music	7.26	7.9
Paroles + Musiques	8.40	9.25

TABLEAU 4.1 SegSNR moyen de la synthèse audio avant et après Posttraitement (dB).

4.4 Codage du masque M

Comme nous le verrons à la section 4.5, le gain en qualité obtenu après le posttraitement fréquentiel est significatif surtout pour de la musique très stationnaire. Mais ce gain en qualité nécessite un débit non négligeable, même si le traitement paraît simple. Le signal masque M est un signal binaire composé de 1s et de 0s. La valeur 1 permet de laisser inchangée la composante fréquentielle correspondante, alors que la valeur 0 la force à zéro. L'information peut être donc codée par un bit par composante spectrale. Utilisant une transformée avec un recouvrement de 50%, le débit nécessaire correspond alors à 1 bit/échantillon pour des signaux réels, ce qui nous donne pour une fréquence d'échantillonnage de 12.8 kHz un débit supplémentaire de 12.8 kbit/s dédié au posttraitement. Afin de réduire les ressources nécessaires pour le posttraitement, nous allons dans la suite de ce chapitre présenter plusieurs techniques de codage du masque M .

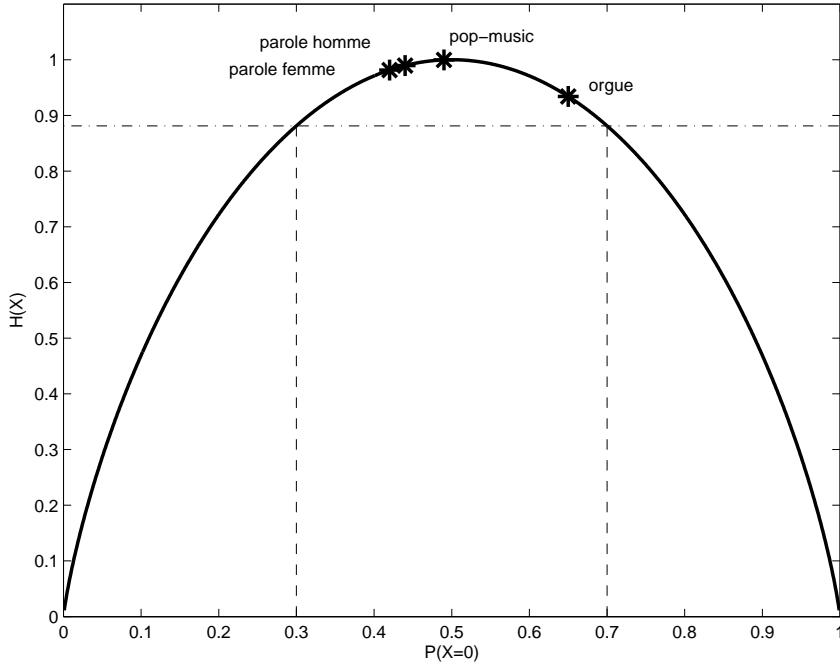
4.4.1 Codage sans perte

Le codage sans perte utilise les propriétés statistiques du signal pour supprimer les redondances. À partir du masque M , on extrait une variable aléatoire X d'alphabet binaire $B = \{0, 1\}$. Le débit minimal nécessaire pour transmettre le signal M en codant les symboles issus de B est donné alors par l'entropie du processus X :

$$H(M) = H(X) = P(X = 0).i(X = 0) + P(X = 1).i(X = 1) \quad (4.3)$$

$$= -P(X = 0).\log_2(P(X = 0)) - (1 - P(X = 0)).\log_2(1 - P(X = 0)) \quad (4.4)$$

$P(X = 0)$ et $P(X = 1)$ désignent la probabilité d'avoir respectivement 1 et 0, et $i(.)$ l'auto-information (*self-information*). Le graphique de la Figure 4.9, nous donne l'entropie du processus en fonction de la probabilité $P(X = 0)$, ce qui permet de connaître le débit minimal atteignable.

Figure 4.9 Entropie du signal masque M pour l'alphabet $B = \{0, 1\}$.

On peut voir que pour des taux de mise à zéro entre 30% et 70%, le débit ne peut être que très rarement inférieur à 0.9 bit/symbole, i.e. 0.9 bit/échantillon. Le Tableau 4.2 reporte l'entropie $H(X)$ calculée pour différents types de sons. On s'aperçoit que le débit minimal atteignable est assez élevé quelle que soit la nature du signal.

Signal	$P(X = 0)$	$P(X = 1)$	$H(X)$ (bit/éch.)
Parole féminine Fr.	0.42	0.58	0.98
Parole masculine All.	0.44	0.56	0.9907
Orgue	0.65	0.35	0.93
Pop-music	0.49	0.51	0.9998
Paroles + Musiques	0.47	0.53	0.9966

TABLEAU 4.2 Entropie du processus binaire du signal M pour différents signaux.

Nous avons vu que la défaillance du codeur parole intervient essentiellement dans des zones spécifiques du spectre, c'est-à-dire les vallées. Cette particularité peut laisser supposer que des valeurs identiques du masque M (série de 1s ou de 0s) sont concentrées dans des zones spectrales spécifiques. Il est alors évident au vu des constatations précédentes qu'un codage par plages

permettrait de réduire le débit nécessaire. Le codage par plages (*Run Length Coding*, RLC) exploite la redondance traduite par de longues séquences d'un même symbole. Le codage par plages transforme alors le processus X d'alphabet B de deux valeurs, 1 et 0, en un processus Y d'alphabet R de plusieurs valeurs correspondantes aux longueurs des plages des 1s et des 0s. Dans notre cas, il y a uniquement deux sortes de plage : une plage de 1s et une plage de 0s. Ainsi, une plage d'une valeur est obligatoirement suivie par une plage de l'autre valeur. La transmission explicite des valeurs n'est donc pas nécessaire, car l'alternance est implicite. Seulement la longueur des plages est à transmettre. On a alors une nouvelle expression de l'entropie du signal M en fonction du nouveau processus Y :

$$H(M) = H(Y) = P(X = 0).H(M|X = 0) + P(X = 1).H(M|X = 1) \quad (4.5)$$

$$= P(X = 0). \frac{H(Y|X = 0)}{E(Y|X = 0)} + P(X = 1). \frac{H(Y|X = 1)}{E(Y|X = 1)} \quad (4.6)$$

$E(Y|X = 1)$ et $E(Y|X = 0)$ représente l'espérance mathématique des longueurs des plages de 1s et de 0s respectivement. Les débits atteignables exposés au Tableau 4.3 sont alors légèrement inférieurs à ceux obtenus précédemment avec l'alphabet binaire (Tableau 4.2).

Signal	$P(X = 0).H(M X = 0)$	$P(X = 1).H(M X = 1)$	$H(Y)$ (bit/éch.)
Parole féminine Fr.	0.35	0.48	0.83
Parole masculine All.	0.24	0.4	0.64
Orgue	0.48	0.34	0.82
Pop-music	0.47	0.47	0.94
Paroles + Musiques	0.43	0.47	0.90

TABLEAU 4.3 Entropie du signal M obtenu par un codage par plages.

Nous avons utilisé un tel codage par plages pour coder le masque M . Pour coder les plages, nous utilisons un codage entropique de Huffman avec deux tables différentes de mots de code : l'une pour les plages de 1s et l'autre pour les plages de 0s. Les mots de code sont adaptatifs et sont générés à partir de tables d'occurrences mises à jour à chaque trame. Avec un tel codage, on obtient des performances proches de l'entropie comme le montre le Tableau 4.4.

On remarque que le débit moyen s'approche de l'entropie. Néanmoins, la variance des débits d'une trame à l'autre est importante, ce qui provoque des pics du débit de sortie. Il est possible de limiter ces débits extrêmes en limitant le débit maximal à 1 bit/échantillon en n'utilisant le

Signal	$H(M)$ (bit/éch.)	Codage Huffman			(kbit/s)
		débit min. (bit/éch.)	débit max. (bit/éch.)	débit moyen (bit/éch.)	
Parole féminine Fr.	0.82	0.15	1.75	0.86	11.00
Parole masculine All.	0.64	0.05	1.17	0.68	8.70
Orgue	0.82	0.19	2.1	0.84	10.75
Pop-music	0.94	0.68	2.43	0.98	12.54
Paroles + Musiques	0.90	0.43	1.58	0.93	12.03

TABLEAU 4.4 Débits du codage par plages avec des codes de Huffman pour le signal M .

codage d'Huffman que lorsque le débit généré est en dessous de 1 bit/échantillon. Si le débit est au-delà on envoie directement les valeurs binaires de M sans codage préalable. Cela amène à faire un codage multimodal du masque M . On obtient les nouveaux débits du Tableau 4.5. Malgré la limitation, les débits moyens restent les mêmes, ce qui prouve que les dépassements de débit au-delà de 1 bit/échantillon sont plutôt rares. Le codage sans perte du masque M montre donc des limites assez sévères. Ces limites peuvent être dépassées par l'utilisation d'un codage avec pertes.

Signal	$H(M)$ (bit/éch.)	Codage Huffman			(kbit/s)
		débit min. (bit/éch.)	débit max. (bit/éch.)	débit moyen (bit/éch.)	
Parole féminine Fr.	0.82	0.15	1.00	0.86	11.00
Parole masculine All.	0.64	0.05	1.00	0.68	8.70
Orgue	0.82	0.19	1.00	0.84	10.75
Pop-music	0.94	0.68	1.00	0.98	12.54
Paroles + Musiques	0.90	0.43	1.00	0.93	12.03

TABLEAU 4.5 Débits du codage RLC du masque M avec limitation du débit à 1 bit/échantillon.

4.4.2 Codage avec pertes

On a vu précédemment que les statistiques des plages de 1s et de 0s ne permettaient pas un gain en compression suffisant. Plusieurs techniques modifiant le masque M sont maintenant présentées. Dans un premier temps, on essaye de modéliser le processus du codage par plages, pour connaître quelles caractéristiques du signal M doivent être améliorées.

Modélisation par un processus de Markov

Il est connu qu'un signal binaire codé par plages peut être, dans certains cas, modélisé par un modèle de Markov de premier ordre [102]. Ce modèle simpliste permet alors de comprendre les statistiques du processus Y correspondant aux longueurs des plages de 1s et de 0s. L'annexe A démontre que le processus de Markov du premier ordre est particulièrement approprié pour modéliser le masque M . La Figure 4.10 représente différentes statistiques du masque M pour différents signaux dans le plan des isocontours de l'entropie du modèle de Markov. Il est alors évident que pour augmenter le taux de compression, c'est-à-dire diminuer l'entropie du processus Y , on doit augmenter l'espérance des longueurs des plages de 1s ou de 0s.

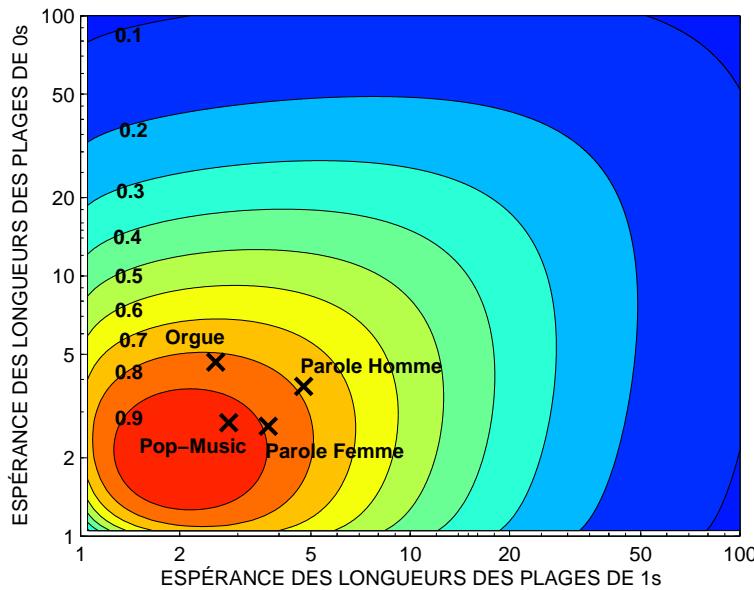


Figure 4.10 Projection des statistiques du masque M de différentes sources dans le plan des équi-contours de l'entropie du modèle de Markov.

Remise en forme du masque

L'objectif de la remise en forme est de rendre le masque M plus adapté à un codage des longueurs de plages tout minimisant l'impact sur la qualité globale du posttraitement. Partant du constat que l'espérance des longueurs de plages puisse être augmentée en évitant l'isolement d'une valeur binaire ou d'un groupe de valeurs au milieu d'une plage de valeur opposée, on se propose d'inverser certaines valeurs du masque M afin de générer de plus longues plages. Bien sûr, la modification

du masque ne doit pas avoir d'impact significatif sur les performances du posttraitement. Pour cela on définit une condition de candidature C pour minimiser l'impact de la remise en forme du masque :

$$C(k) = \begin{cases} 1 & \text{si } \frac{|X(k)|}{g_1} < |E_{base}(k)| < g_0 |X(k)| \\ 0 & \text{sinon} \end{cases} \quad (4.7)$$

où $g_0, g_1 \geq 1$ sont deux constantes fixées par l'utilisateur représentant une tolérance. $X(k)$ et $E_{base}(k)$ sont les composantes spectrales du signal original et de l'erreur du codage de parole respectivement. Si $C(k) = 1$, alors la composante associée du masque $M(k)$ est autorisée à être inversée. Plus g_0 (resp. g_1) est grand, plus les composantes du masque de valeur 0 (resp. 1) sont exposées à être candidates pour être inversées.

Une autre condition doit être définie (condition d'isolement) pour permettre d'assurer que la modification du masque améliore les statistiques du signal. Une première condition d'isolement est définie. Elle permet de détecter toute valeur isolée au milieu d'une plage de valeur différente.

$$I(k) = \begin{cases} 1 & \text{si } \forall i = \{k-p, \dots, k-1, k+1, \dots, k+p\} \quad M(k) \neq M(i) \\ 0 & \text{sinon} \end{cases} \quad (4.8)$$

Finalement, les composantes du masque M doivent vérifier les deux conditions précédentes pour être inversées :

$$I' = C \wedge I \quad (4.9)$$

On obtient alors un nouveau masque M' :

$$M' = (\neg I' \wedge M) \vee (I' \wedge \neg M) = I' \oplus M \quad (4.10)$$

En utilisant les paramètres $p = 1$ et $g_0 = g_1 = \infty$, on arrive à éliminer complètement les valeurs isolées, comme le montre l'exemple de la Figure 4.11. Le Tableau 4.6 donne les nouveaux débits obtenus en utilisant toujours une limitation du débit à 1 bit/échantillon. En moyenne, on obtient un débit de 0.5 bit/échantillon, pour une perte de qualité négligeable dans la grande majorité des cas. Le gain est assez impressionnant et encourageant pour essayer d'améliorer l'algorithme de remise en forme. On peut observer à la Figure 4.12 l'amélioration dans le plan des isocontours de l'entropie du modèle de Markov.

Signal	$H(M')$ (bit/éch.)	Codage Huffman			(kbit/s)
		débit min. (bit/éch.)	débit max. (bit/éch.)	débit moyen (bit/éch.)	
Parole féminine Fr.	0.49	0.12	0.73	0.51	6.52
Parole masculine All.	0.35	0.05	0.76	0.38	4.86
Orgue	0.52	0.19	0.88	0.53	6.78
Pop-music	0.59	0.37	1.00	0.61	7.80
Paroles + Musiques	0.56	0.16	1.00	0.57	7.30

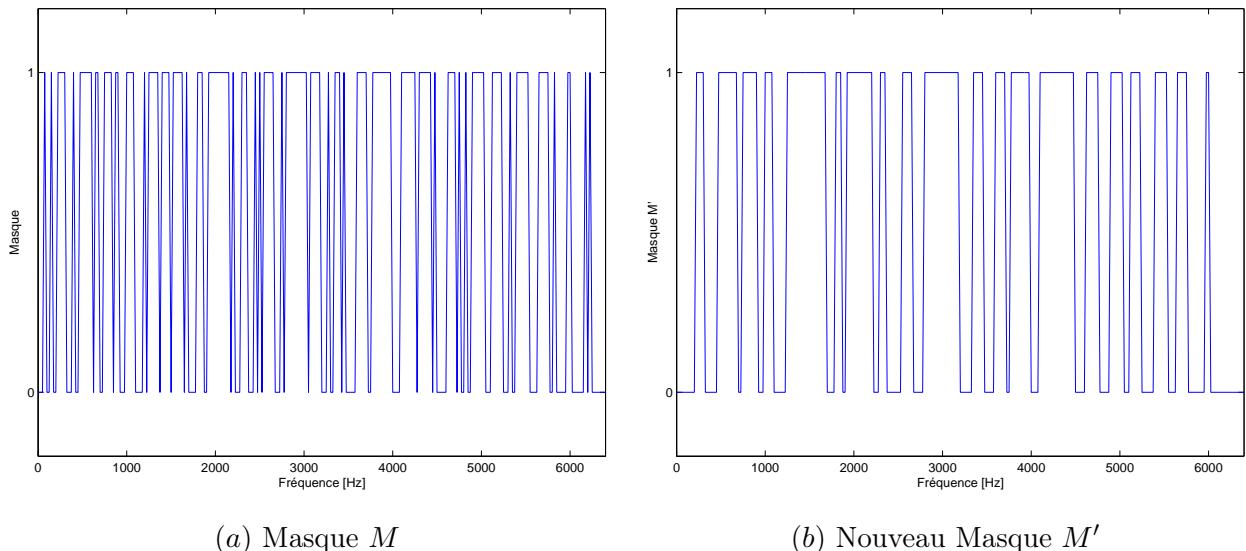
TABLEAU 4.6 Débits pour le codage par plages du masque remis en forme M' .

Figure 4.11 Remise en forme du masque par élimination des valeurs isolées.

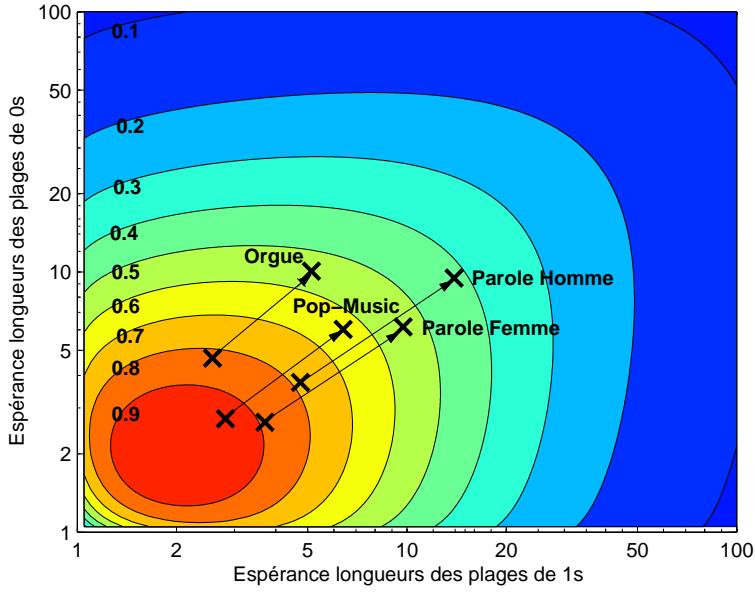


Figure 4.12 Améliorations des statistiques de M vers M' pour différentes sources dans le plan des isocontours de l'entropie du modèle de Markov.

La remise en forme correspondant aux équations 4.7 à 4.10 permet seulement de traiter des cas isolés, c'est-à-dire d'éviter qu'une composante du masque M soit noyée parmi des composantes de valeur opposée. La répercussion d'un tel procédé sur les statistiques du masque M est donc limitée. On introduit alors une remise en forme plus complexe. Les composantes du masque qui satisfont la condition de candidature sont traitées cette fois-ci par plages. Une plage de candidats NC prendra alors la même valeur suivant l'état de son voisinage. En effet, toute plage de candidats est entourée de chaque côté par deux coefficients non candidats. On procède alors selon l'algorithme 1 donné à la page suivante.

Le procédé génère un nouveau masque M'' . Pour cette méthode les constantes g_0 et g_1 jouent un rôle beaucoup plus important que lors de la première remise en forme. En effet, elles vont dicter la longueur des plages de candidats. L'expérience a montré que le posttraitement a tendance à mettre un peu trop de composantes à zéro, ce qui peut diminuer sensiblement la qualité de la synthèse de la parole. Pour cette raison, nous fixons les gains g_0 et g_1 de façon à avoir une stratégie plutôt conservatrice, c'est à dire privilégier davantage une modification des 0s vers des 1s que des 1s vers des 0s. Toujours par expérimentations les constantes g_1 et g_2 ont été fixées à 1.3 et 5 respectivement. Le Tableau 4.7 montre le gain remarquable en compression obtenu avec

Algorithme 1 Algorithme de remise en forme du masque par plages

Chercher une plage de candidats NC .

if NC est comprise entre deux non-candidats de même valeur **then**

Tous les candidats de la plage NC prennent la valeur des deux non-candidats voisins.

else

Calcul de la somme des erreurs relatives au signal original sur la longueur de la plage NC :

$$W = \sum_{k \in NC} |X(k)| - |E_{base}(k)|$$

if $W \geq 0$ **then**

Tous les candidats de la plage NC prennent la valeur 1.

else

Tous les candidats de la plage NC prennent la valeur 0.

end if

end if

le masque M'' . La Figure 4.13 illustre bien les longues plages générées par la remise en forme par plages. Cela se traduit à la Figure 4.14 par une amélioration des statistiques lors du passage du masque M au masque M'' . Bien sûr, ce gain en compression est dû à une forte distorsion du masque original M , bien qu'étant contrainte en fonction de l'impact sur la qualité de la synthèse (cf. équation 4.7). Cette distorsion, comme on le verra dans la section 4.5, n'est pas préjudiciable à la qualité de restitution de la parole. Au contraire, la stratégie conservatrice augmente même la qualité globale. Par contre pour de la musique, les performances sont en léger retrait par rapport au codage sans perte.

Signal	$H(M'')$ (bit/éch.)	Codage Huffman			
		débit min. (bit/éch.)	débit max. (bit/éch.)	débit moyen (bit/éch.)	(kbit/s)
Parole féminine Fr.	0.18	0.06	0.6	0.2	2.56
Parole masculine All.	0.13	0.04	0.61	0.15	1.92
Orgue	0.47	0.09	0.68	0.49	6.27
Pop-music	0.35	0.09	0.71	0.36	4.61
Paroles + Musiques	0.30	0.06	0.77	0.31	3.97

TABLEAU 4.7 Débits pour le codage par plages du masque remis en forme M'' .

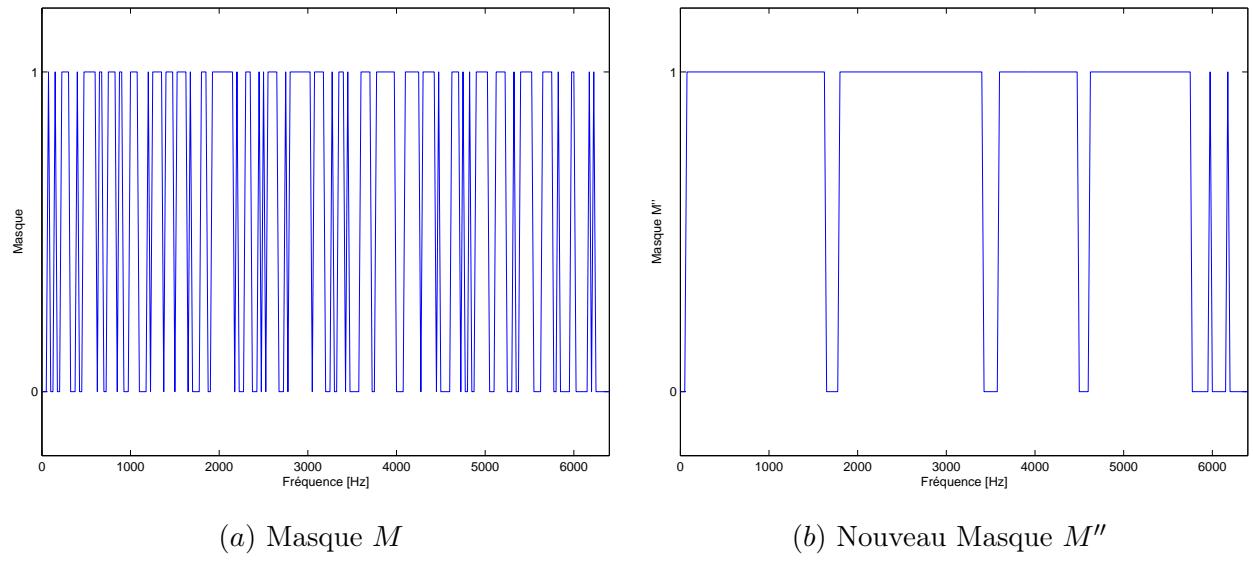
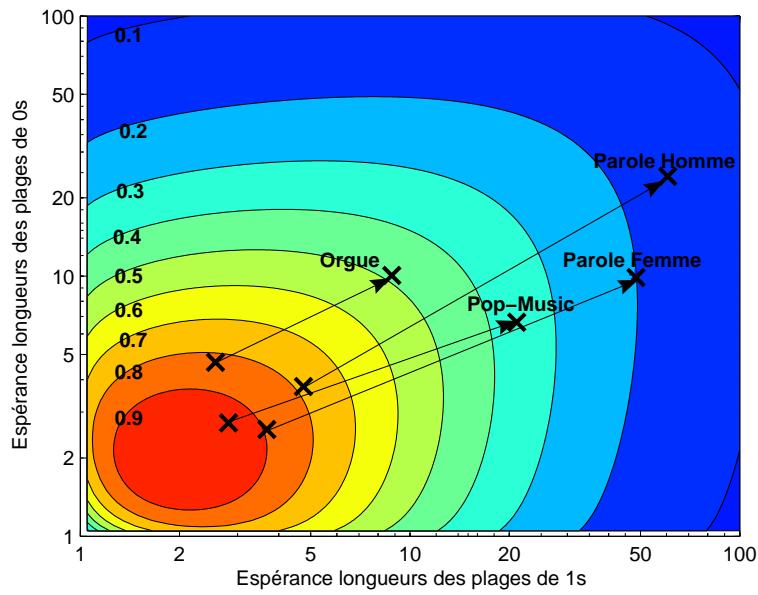


Figure 4.13 Remise en forme par plages du masque.

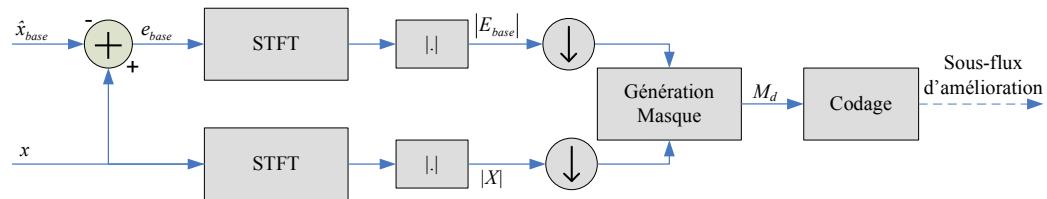
Figure 4.14 Améliorations des statistiques de M vers M'' pour différentes sources dans le plan des isocontours de l'entropie du modèle de Markov.

4.4.3 Compromis débit/délai/qualité

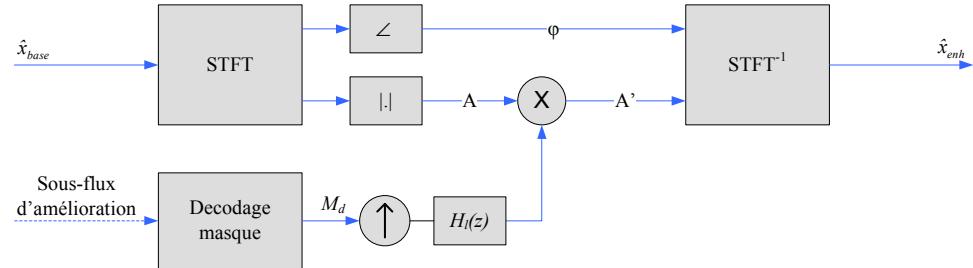
Dans cette section, on discute de différents compromis qui peuvent être faits que ce soit au niveau du délai ou du débit. Comme on l'a vu précédemment, une bonne précision fréquentielle est essentielle pour extraire la quintessence du traitement. Or l'augmentation de la taille de la transformée de Fourier va de pair avec l'augmentation du délai nécessaire. De plus, une fenêtre d'analyse trop grande diminue la résolution temporelle du traitement. On a donc un premier compromis entre la résolution fréquentielle et temporelle qui va de pair avec le compromis entre la résolution fréquentielle et le délai. Ce compromis se traduit par le choix de la longueur de la fenêtre d'analyse. Nous avons choisi une fenêtre d'analyse de 40 ms après plusieurs expérimentations avec plusieurs signaux audio en se basant sur la qualité perceptuelle de la synthèse post-traitée.

Il est possible de s'affranchir de ce compromis en augmentant le débit. En effet pour avoir une meilleure résolution fréquentielle sans pour autant diminuer la résolution temporelle, il est possible d'utiliser une technique tel le remplissage de zéros (*zero padding*). Cette technique consiste à appliquer une DFT de taille supérieure à la fenêtre d'analyse. Les échantillons temporels manquants sont alors mis à zéro. Le débit binaire du signal M augmente ainsi du même facteur que celui du *zero padding* appliqué. Le compromis est alors entre le débit et la résolution fréquentielle. Si on veut obtenir une meilleure résolution temporelle sans rogner la résolution fréquentielle, on peut utiliser un recouvrement supérieur à 50%. Le compromis est cette fois-ci entre le débit et la résolution temporelle.

Inversement, des applications peuvent être plus exigeantes sur le débit. Une façon radicale de diminuer le débit engendré consiste alors à sous-échantillonner le spectre en regroupant deux ou plusieurs composantes. On diminue alors la résolution fréquentielle tout en n'augmentant pas la résolution temporelle. Le débit binaire du masque M diminue ainsi du même facteur que celui de la décimation. La Figure 4.15 donne le schéma de principe de l'opération au codage et au décodage. Au décodeur, le masque décimé M_d est interpolé. L'interpolation est la même que le sur-échantillonnage d'un signal audio. Elle est réalisée par un filtrage passe-bas $H_l(z)$ de fréquence de coupure $\pi/2$. Les valeurs du masque prennent alors des valeurs réelles entre 0 et 1 ce qui pondère le posttraitement entre deux valeurs différentes du masque décimé M_d . Un exemple de masque interpolé d'un facteur 4 est donné à la Figure 4.16.



(a) Décimation du spectre d'amplitude et génération du masque décimé



(b) Interpolation et traitement au décodage

Figure 4.15 Décimation du masque pour une réduction du débit.

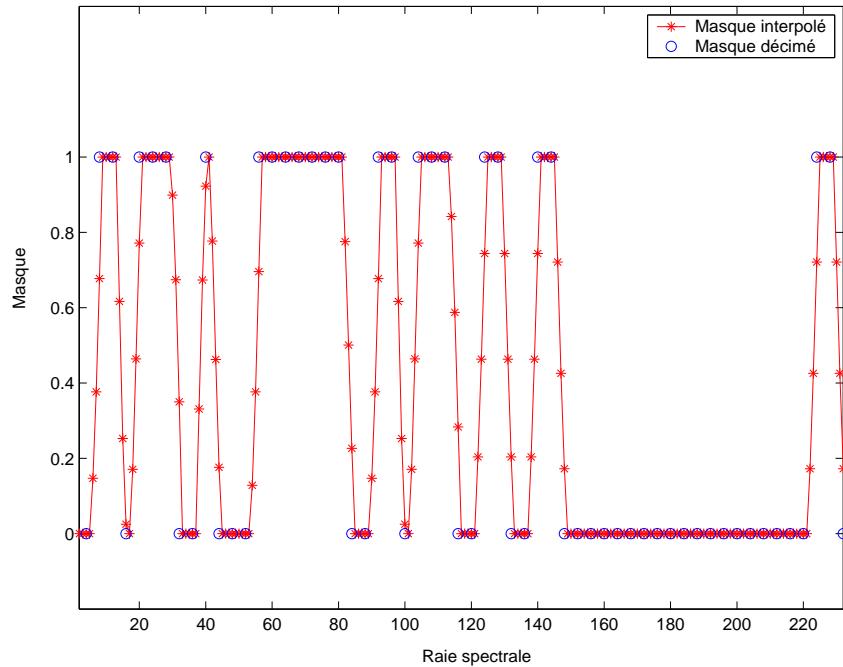


Figure 4.16 Interpolation de facteur 4 du masque.

4.4.4 Codage multimodal

On a déjà aperçu précédemment que le codage entropique a un inconvénient majeur qui est la fluctuation de son débit de sortie. Cette variabilité peut être gênante pour certaines applications du fait de l'impossibilité de prédire le débit de sortie exact du train binaire. L'ennui majeur est le manque d'une borne supérieure pour pouvoir au moins allouer les ressources nécessaires pour établir la transmission. On a vu qu'il était possible de limiter le débit du posttraitement à 1 bit/échantillon en décidant de ne pas appliquer de codage sur le masque lorsque le taux de compression du codage est inférieur à 1. Ce même principe de codage multimodal peut être utilisé avec la décimation du masque pour obtenir une limite supérieure du débit moins élevée. La Figure 4.17 donne le diagramme du codage multimodal du masque pour une borne supérieure du débit de $1/F$ bit/échantillon, où F est le facteur de décimation. La variable $Flag_d$ représente le mode sélectionné au codeur, et doit être transmis sur 1 bit. Si $F = 2$ alors le débit maximal est de 0.5 bit/échantillon, i.e. 6.4 kbit/s à 12.8 kHz d'échantillonnage. Pour $F = 4$ il est de 0.25 bit/échantillon (3.2 kbit/s).

Les Tableaux 4.8 et 4.9 donnent les débits obtenus pour le codage multimodal pour deux différents facteurs F donc deux limites supérieures de débit différentes. Il est intéressant de noter que pour la parole, la décimation du masque est beaucoup moins fréquente que pour la musique. Pour ces signaux et pour un facteur $F = 2$, le ratio de décimation est négligeable. On peut ainsi considérer que l'impact de la décimation sur les performances du posttraitement l'est aussi. Pour $F = 4$, le ratio de décimation est par contre autour de 25%. L'impact est plus important, et quelques petits artefacts viennent contrebalancer le gain en qualité global apporté par le posttraitement. Généralement, les artefacts apparaissant après le codage multimodal remplacent des artefacts déjà présents dans la synthèse du codeur de parole. Par contre, la nature des défauts n'est plus la même. L'apport du posttraitement est donc plus mitigé, et dépendra du signal parole d'entrée et de l'appréciation de l'auditeur. Dans le cas de la musique, la décimation est au contraire quasi-systématique, surtout pour $F = 4$. Il est alors évident que l'efficacité du posttraitement diminue. Ce compromis sur la qualité du posttraitement par rapport au débit, est légitime en considérant que l'apport du posttraitement pour ces types de signaux est tellement significatif que brider sa performance au profit d'une baisse significative du débit le rend quant même attrayant.

Comme tout codage multimodal, on a un critère de discrimination, qui est dans notre cas le débit généré par le posttraitement. Comme ce débit est issu d'un processus proche de celui du

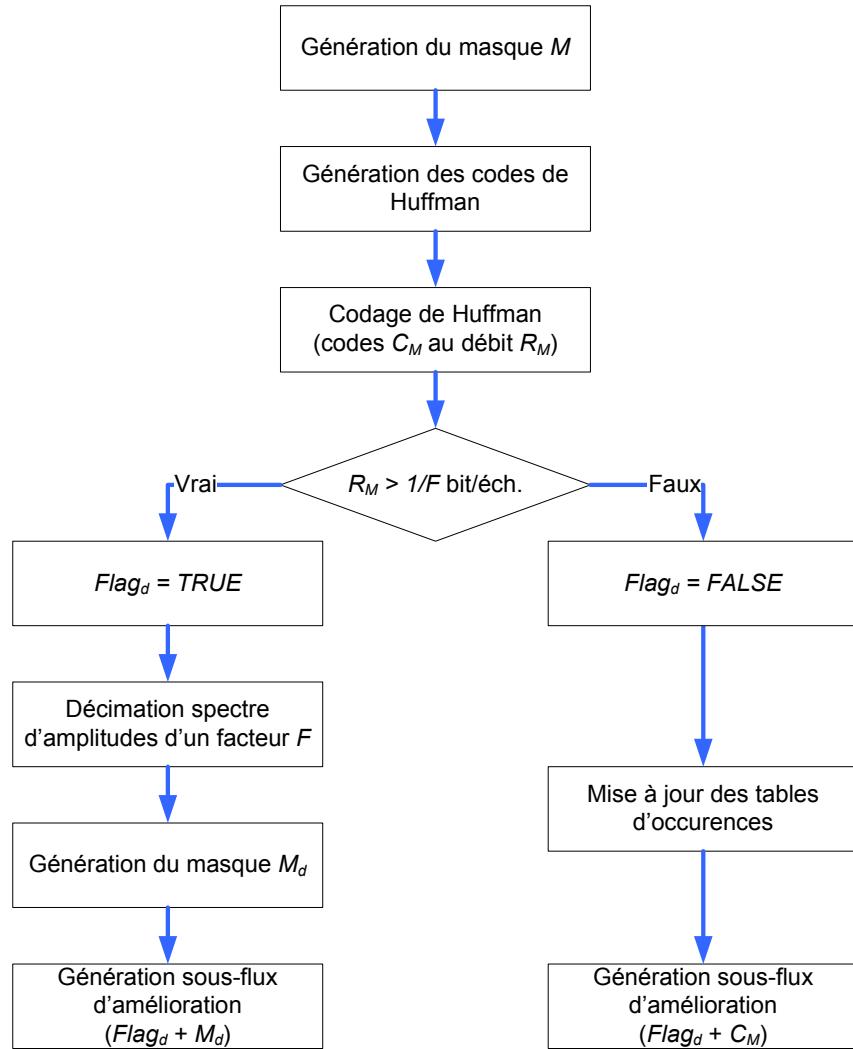


Figure 4.17 Codage multi-modal du masque.

Signal	Codage Huffman				Ratio $Flag_d$ (%)
	débit min. (bit/éch.)	débit max. (bit/éch.)	débit moyen (bit/éch.)	(kbit/s)	
Parole féminine Fr.	0.05	0.5	0.19	2.43	0.8
Parole masculine All.	0.04	0.5	0.15	1.92	0.6
Orgue	0.09	0.5	0.46	5.88	52.2
Pop-music	0.074	0.5	0.35	4.48	16.2
Paroles + Musiques	0.06	0.5	0.30	3.84	21.19

TABLEAU 4.8 Débits pour le codage multimodal du masque avec $F = 2$.

Signal	Codage Huffman			Ratio $Flag_d$ (%)
	débit min. (bit/éch.)	débit max. (bit/éch.)	débit moyen (bit/éch.) (kbit/s)	
Parole féminine Fr.	0.04	0.25	0.17	26.2
Parole masculine All.	0.04	0.25	0.14	11.6
Orgue	0.09	0.25	0.25	3.2
Pop-music	0.06	0.25	0.23	78.4
Paroles + Musiques	0.04	0.25	0.21	54.12

TABLEAU 4.9 Débits pour le codage multi-modal du masque avec $F = 4$.

modèle de Markov de premier ordre, on peut visualiser le critère de discrimination dans le plan des isocontours de l'entropie du modèle de Markov. La Figure 4.18 confirme que la parole se situe en moyenne dans la zone sous les 0.25 bit/échantillon, alors que les signaux musicaux se trouvent dans la zone entre 0.5 et 0.25 bit/échantillon.

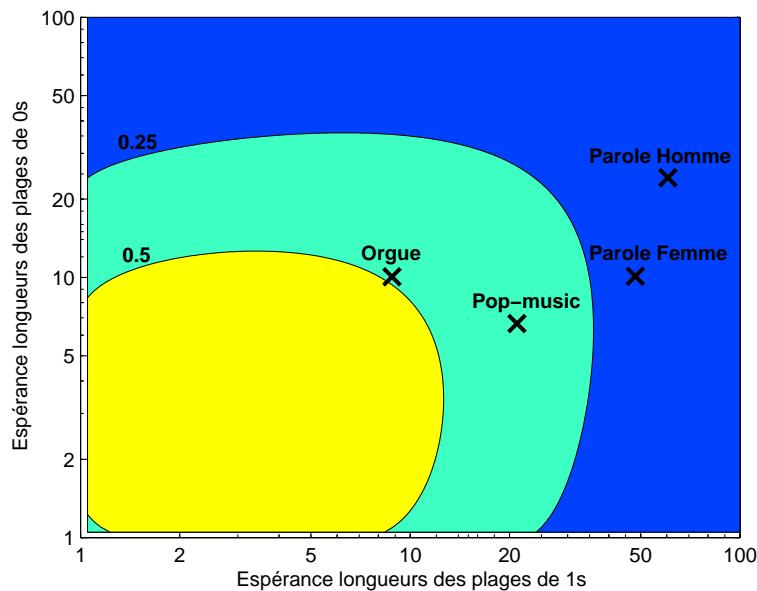


Figure 4.18 Représentation de la discrimination des signaux lors du codage multi-modal par des zones du plan des iso-contours de l'entropie du modèle de Markov.

4.5 Performances et tests

Afin d'évaluer les performances du posttraitement, nous avons conduit un test subjectif en utilisant la recommandation MUSHRA de l'ITU-R [103]. Huit auditeurs experts ont évalué sur une échelle de 0 à 100 quatre séquences audio : deux séquences de parole féminine et masculine, et deux séquences musicales. Chacune des séquences est échantillonnée à 16kHz et est traitée par 5 codecs différents : l'AMR-WB à 12.65 kbit/s, le G.722.1 à 24 kbit/s, et l'AMR-WB associé à notre posttraitement dans trois versions différentes. Les trois versions diffèrent par leur fenêtre d'analyse de 40 ou de 80 ms et par la description du masque, transmise, soit sans codage particulier ou soit avec le codage avec pertes tel que décrit par l'algorithme 1. Le débit associé à l'AMR-WB et au posttraitement sans codage du masque est de 25.45 kbit/s ($12.65 + 12.8$) alors que pour le codage avec pertes du masque le débit est fonction de la séquence. Il oscille entre 14.57 et 18.92 kbit/s. La référence cachée (*hidd. ref.*) est rajoutée aux 5 autres synthèses pour juger de la pertinence des votes des auditeurs. Le G.722.1 est un codeur large bande par transformée qui à l'inverse de l'AMR-WB est particulièrement efficace pour des signaux musicaux. Les résultats du test sont reportés à la Figure 4.19 avec l'intervalle de confiance à 95% associé.

La première constatation est que l'AMR-WB à 12.65 kbit/s devance le G.722.1 à 24 kbit/s pour la parole. Pour la musique les rôles sont inversés. C'est une mise en évidence des caractéristiques des deux paradigmes existant en codage audio, le codage de la parole et le codage audio générique. Le posttraitement affecte peu la qualité de l'AMR-WB dans le cas de la parole. Pour un fenêtrage de 80 ms, la qualité a tendance à baisser alors qu'elle se maintient pour 40 ms voire augmente dans le cas du codage avec pertes. Cette augmentation est due au caractère conservateur de la remise en forme du masque.

Dans le cas de la musique, le posttraitement augmente clairement la qualité de la synthèse de l'AMR-WB. Cela se traduit pour l'orgue par un gain de 30 à 46 points MUSHRA. Comme la musique est de nature plus stationnaire que la parole, c'est cette fois-ci les trames de 80 ms qui sont les plus adaptées. La version sans codage à 25.45 kbit/s est plus performante que celle du codage avec pertes autour de 18 kbit/s. Malgré cela, toutes les versions du posttraitement réduisent significativement l'écart entre les performances de l'AMR-WB à 12.65 kbit/s et du G.722.1 à 24 kbit/s pour les signaux musicaux. Pour la version avec codage avec pertes, il reste encore aux alentours de 6 kbit/s pour rattraper à débit équivalent la qualité du G.722.1.

4.6 Conclusions

Dans le présent chapitre, nous avons introduit une nouvelle technique de posttraitement fréquentiel de la synthèse d'un codeur parole. Le posttraitement permet d'améliorer la qualité de la restitution sonore, surtout lors du traitement de signaux autres que la parole. L'avantage est de pouvoir améliorer significativement la qualité pour des signaux musicaux avec un procédé simple et un débit pouvant être limité en faisant un certain compromis sur l'efficacité du posttraitement. De plus, le posttraitement ne détériore pas la qualité de la parole.

Le fait de modifier le spectre d'amplitude de la synthèse sonore n'est pas une approche unique. Il est courant d'utiliser une technique de postfiltrage à la sortie du décodeur parole pour améliorer la synthèse audio. Le postfiltrage s'appuie sur des filtres adaptatifs qui pondèrent le spectre d'amplitude de la synthèse. Les raisons peuvent se trouver dans la théorie de Wiener [104]. Le filtrage optimal pour minimiser un bruit dans un signal est d'appliquer un filtre de fonction de transfert $H(\omega) = S(\omega)/[S(\omega) + N(\omega)]$, où $S(\omega)$ et $N(\omega)$ sont respectivement la densité spectrale du signal et du bruit. Cela signifie que dans les parties du spectre où le SNR est grand le gain du filtre sera de l'ordre de l'unité, alors que dans les parties où le SNR est petit le gain sera très faible. Si on considère que le bruit est presque uniforme sur tout le spectre, le SNR est alors plus important au niveau des pics du spectre que dans les vallées. Même si le bruit n'est pas uniforme, la majorité des codeurs parole privilégie la description des pics spectraux aux vallées. Il est donc nécessaire de creuser les vallées du spectre de la synthèse. Le postfiltrage permet une telle optimisation à l'aide d'un filtre à court terme amplifiant les caractéristiques formatiques du signal et d'un filtre à long terme accentuant les harmoniques. Le postfiltrage est, contrairement à notre posttraitement, uniquement adapté pour de la parole.

D'un autre côté, notre posttraitement, bien qu'étant efficace, est une procédure simple et donc limitée. Il ne peut prétendre à atteindre une qualité transparente. Cette qualité ne sera atteignable que par une description explicite des distorsions venant du codeur de parole. Comme on l'a vu, le posttraitement permet surtout de réduire les distorsions hautement perceptibles. Dans le même temps, le posttraitement garantit aussi la diminution de l'erreur quadratique de la synthèse. Il n'est donc pas contradictoire d'utiliser le posttraitement en association avec un codage explicite des distorsions. Au contraire, le posttraitement sert alors de décision critique pour les couches supérieures du codage : soit on améliore la qualité du codeur de parole en codant les composantes

du signal différence, soit on ignore complètement l'apport du codeur parole en codant directement le signal original. Cette dernière décision est parfois nécessaire pour empêcher un gaspillage des ressources à défaire les défauts du codeur de parole au lieu de décrire directement les composantes originales. Cette philosophie se rapproche de la philosophie initiatrice du posttraitement qui est de ne rien transmettre au lieu d'hériter d'un défaut handicapant, ce qui se traduit encore plus simplement par plutôt ne rien faire que de faire mal. Il est à noter que ce même principe a été utilisé par un codeur hiérarchique combinant l'AMR-WB et le codeur AAC [105]. Ce travail a été effectué en parallèle du travail présenté ici. Contrairement à notre solution, il prend une décision au niveau d'un regroupement en sous-bandes suivant les bandes critiques de l'audition. Pour cette raison, la décision ne peut pas suffire à elle-même pour améliorer la qualité de la synthèse du codeur parole comme c'est le cas pour notre posttraitement. En effet, elle doit être suivie d'un codage explicite des distorsions ou des composantes originales selon la décision.

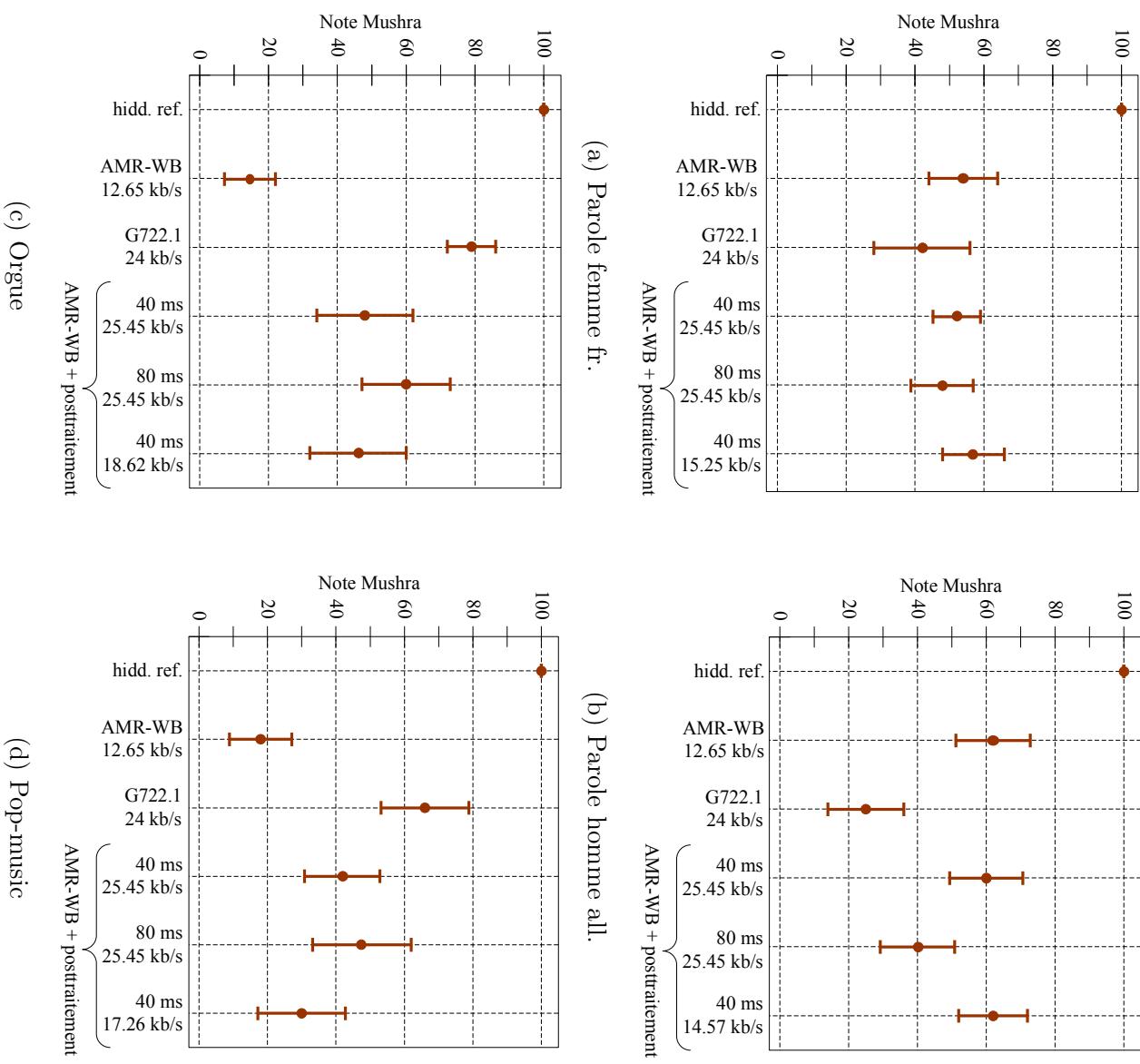


Figure 4.19 Résultats du test MUSHRA : notes moyennes avec l'intervalle de confiance à 95%.

CHAPITRE 5

Quantification Algébrique à Raffinements Successifs

5.1 Introduction

Dans le chapitre précédent, nous avons présenté un traitement pouvant dans certains cas améliorer significativement la qualité de la restitution sonore à la suite d'un codage par un codeur de parole. Nous souhaitons maintenant pouvoir affiner progressivement la description de cette restitution afin d'arriver à une qualité quasi transparente entre l'original et la synthèse audio. Il est donc nécessaire de quantifier l'erreur de codage. On impose en plus que le décodage de l'erreur puisse se faire de façon incrémentale avec une granularité très fine.

Le raffinement successif d'un signal se fait généralement à l'aide de quantificateurs contraints. La contrainte permet entre autres de hiérarchiser les informations à quantifier. Elle peut simplement consister à mettre en cascade une série de quantificateurs afin d'obtenir une quantification multiétages (*multistage VQ*) [14], où l'erreur de l'approximation de l'étage présent est quantifiée à l'étage suivant. Une autre façon est de contraindre le dictionnaire à avoir une structure arborescente (*tree-structured VQ*) [14]. À chaque niveau de l'arbre, le vecteur à quantifier est classé dans une classe qui sera précisée au niveau suivant.

La quantification algébrique par réseau régulier de points (*Lattice Vector Quantization*, LVQ) est aussi une quantification par contrainte. La contrainte porte alors sur le dictionnaire de quantification, dans le sens où celui-ci est défini comme un sous-ensemble d'un réseau régulier de points. Ce type de contrainte est généralement utilisé pour sa très faible complexité de recherche du plus proche voisin et pour l'espace négligeable de mémoire nécessaire au stockage du dictionnaire. En effet, la recherche du représentant ainsi que la définition du dictionnaire sont entièrement algébriques. Nous allons montrer dans ce chapitre que les propriétés remarquables des réseaux réguliers de points peuvent être aussi exploitées afin d'obtenir un raffinement graduel de la source. En effet, une indexation appropriée des points du réseau par une série de codes de Voronoï permet d'obtenir une quantification algébrique à raffinements successifs avec une granularité aussi fine que 1 bit/dimension.

Dans ce chapitre, nous introduisons dans un premier temps la quantification vectorielle et les avantages qui en découlent par rapport à la quantification scalaire. Nous nous penchons ensuite sur la quantification par contrainte avant d'introduire plus spécifiquement, dans la troisième section, la LVQ. Dans la quatrième section du chapitre, nous introduisons une nouvelle quantification algébrique à raffinements successifs. On présente tout d'abord l'extension de Voronoï, adaptation réussite de la LVQ en codage audio, avant de proposer une version graduelle permettant un raffinement graduel de la source avec une très fine granularité. Nous concluons le chapitre par des mesures de performance de la quantification algébrique à raffinements successifs proposée.

5.2 Quantification vectorielle

5.2.1 Définitions

La quantification vectorielle est un outil puissant en compression du signal. Elle peut être vue comme une généralisation de la quantification scalaire pour un ensemble de scalaires formant un vecteur. Mais elle est bien plus que cela, car le saut d'une dimension à plusieurs permet l'introduction de nouveaux concepts, idées et réalisations.

Un quantificateur vectoriel Q de dimension N et de taille L est défini comme une fonction qui associe un vecteur x d'entrée de dimension N dans l'espace Euclidien \mathbb{R}^N , à un vecteur de sortie y choisi parmi un ensemble fini $\mathcal{C} = \{y_1, y_2, \dots, y_L\}$ contenant L vecteurs de dimension N dans \mathbb{R}^N appelés code-vecteurs :

$$Q : \mathbb{R}^N \longrightarrow \mathcal{C} \quad (5.1)$$

L'ensemble \mathcal{C} est appelé dictionnaire. La résolution R , ou encore débit, du quantificateur est définie comme le nombre de bits par échantillon utilisé pour représenter le vecteur d'entrée.

$$R = \frac{\log_2 L}{N} \quad (5.2)$$

Contrairement à la quantification scalaire, il n'est pas nécessaire que R soit un entier, il suffit seulement que NR le soit. Cela permet de définir des résolutions fractionnaires.

Pour que la distorsion engendrée par la quantification soit minimale, ce qui correspond à un codage optimal en compression du signal, il faut que l'espace \mathbb{R}^N soit divisé en L régions V_i chacune associée à un code-vecteur y_i du dictionnaire, et répondant au critère du plus proche

voisin. La partition de l'espace est alors complètement définie par le dictionnaire et une mesure de distorsion. Généralement la distorsion euclidienne est utilisée :

$$d(x, y_i) = (\|x - y_i\|)^2 = \sum_{j=1}^N (x(j) - y_i(j))^2 \quad (5.3)$$

Les régions répondant au critère du plus proche voisin, encore appelées cellules ou régions de Voronoï, sont définies de la façon suivante :

$$V_i = \{x | d(x, y_i) \leq d(x, y_j) \quad \forall j \in 1, 2, \dots, L\} \quad (5.4)$$

Pour que la quantification vectorielle soit performante, elle doit être adaptée à la source. La génération du dictionnaire se fait dans la majorité des cas à l'aide d'une base d'apprentissage. Elle est composée d'un grand nombre de vecteurs représentatifs de la probabilité de distribution de la source. L'algorithme généralisé de Lloyd [14], connu aussi sous le nom de k -moyennes, est une méthode classique pour en extraire un dictionnaire. On parle alors de quantification stochastique. Elle nécessite le stockage en mémoire de son dictionnaire et le parcours exhaustif de l'ensemble de ses code-vecteurs afin de trouver le meilleur représentant du vecteur à coder.

5.2.2 Les avantages de la quantification vectorielle

Il est possible de définir la performance d'une quantification de façon analytique si on se situe dans l'hypothèse de haute résolution. La théorie du codage haute résolution ou asymptotique émise par Bennett [106] considère que l'espace vectoriel est tellement morcelé en un grand nombre de cellules que la distribution intracellulaire est uniforme. Cette théorie suppose que le débit soit élevé. C'est généralement le cas pour le codage haute fidélité. Autrement, c'est une limite asymptotique intéressante pour comparer les performances. Dans cette hypothèse, Zador [107] a montré que la puissance de l'erreur de la quantification vectorielle de dimension N pouvait s'exprimer uniquement en fonction de la densité de probabilité $p_s(x)$:

$$\sigma_Q^2(R, N) = c(N) \|p_s(x)\|_{N/(N+2)} 2^{-2R} \quad (5.5)$$

$$= c(N) \left[\int_{\mathbb{R}^N} p_s(x)^{N/(N+2)} dx \right]^{(N+2)/N} 2^{-2R} \quad (5.6)$$

$c(N)$ étant une constante qui ne dépend que de N . Pour une quantification scalaire, $N = 1$, on retrouve l'expression classique de l'erreur de quantification [15] :

$$\sigma_Q^2(r, 1) = c(1) \left[\int_{\mathbb{R}} p_s(x)^{1/3} dx \right]^3 2^{-2R} \quad (5.7)$$

avec $c(1) = 1/12$. Il est à noter que si x est régi par une loi uniforme de distribution (p_s est alors une fonction constante), alors on retrouve la relation fondamentale $\sigma_Q^2(R, 1) = \sigma_s^2 2^{-2R}$, où σ_s^2 est la variance du signal d'entrée.

Lookabaugh et Gray [108] définissent alors le gain de la quantification vectorielle de dimension N par rapport à la quantification scalaire de même résolution R , comme le ratio des puissances de l'erreur issues des deux processus :

$$G_g(N) = \frac{\sigma_Q^2(R, 1)}{\sigma_Q^2(R, N)} \quad (5.8)$$

En faisant l'hypothèse que le signal d'entrée est stationnaire, la probabilité de chaque composante d'un vecteur du signal est la même probabilité $\tilde{p}_s(x)$, et est dite marginale. De plus, si les composantes du vecteur sont indépendantes entre elles, alors la probabilité $p_s(x)$ du vecteur est égale à la probabilité \bar{p}_s définie comme suit :

$$\bar{p}_s(x) = \prod_{i=1}^N \tilde{p}_s(x_i) \quad (5.9)$$

On obtient alors l'expression suivante du gain :

$$\begin{aligned} G_g(N) &= \frac{c(1)}{c(N)} \frac{\|\tilde{p}_s(x)\|_{1/(3)}}{\|\bar{p}_s(x)\|_{N/(N+2)}} \frac{\|\bar{p}_s(x)\|_{N/(N+2)}}{\|p_s(x)\|_{N/(N+2)}} \\ &= G_{g1}(N)G_{g2}(N)G_{g3}(N) \end{aligned} \quad (5.10)$$

avec

$$\begin{aligned} G_{g1}(N) &= \frac{c(1)}{c(N)} \\ G_{g2}(N) &= \frac{\|\tilde{p}_s(x)\|_{1/(3)}}{\|\bar{p}_s(x)\|_{N/(N+2)}} \\ G_{g3}(N) &= \frac{\|\bar{p}_s(x)\|_{N/(N+2)}}{\|p_s(x)\|_{N/(N+2)}} \end{aligned}$$

Le gain total $G_g(N)$, exprimé en dB, est la somme de trois contributions différentes. Le premier gain $G_{g1}(N)$ caractérise la capacité du quantificateur à couvrir l'espace (*space filling advantage*). Ce gain dépend uniquement de $c(N)$. La Figure 5.1 illustre cet avantage en comparant pour une source bidirectionnelle uniforme le dictionnaire obtenu par un produit cartésien d'un quantificateur scalaire par lui-même avec celui d'une quantification vectorielle de dimension 2 optimale. On note que les régions de Voronoï d'une quantification scalaire uniforme sont représentées en

dimension N pour des hypercubes de dimension N . En dimension 2, on s'aperçoit que pour un même espace couvert, la partition de l'espace par des cellules carrées engendre une distorsion moyenne plus élevée qu'un pavage d'hexagones. Gersho [109] a conjecturé que le coefficient $c(N)$ correspond au moment normalisé d'ordre 2 de l'erreur de quantification au sein d'une région de Voronoï pour une distribution uniforme.

$$c(N) = \frac{1}{N} \frac{1}{\text{vol}(V)^{1+2/N}} \int_V \|x\|^2 dx \quad (5.11)$$

où $\text{vol}(V)$ est le volume de la région de Voronoï V . Pour N donné, on montre que le gain est maximal lorsque le polytope formé par les faces de la cellule de Voronoï se rapproche de la forme d'une N -sphère, c.-à-d. une sphère en dimension N . Le gain est maximal lorsque N tend vers l'infini et s'approche alors de la borne maximum de $\frac{\pi e}{6}$ (1.53 dB) [110].

$$\lim_{N \rightarrow \infty} c(N) = (2\pi e)^{-1} \quad (5.12)$$

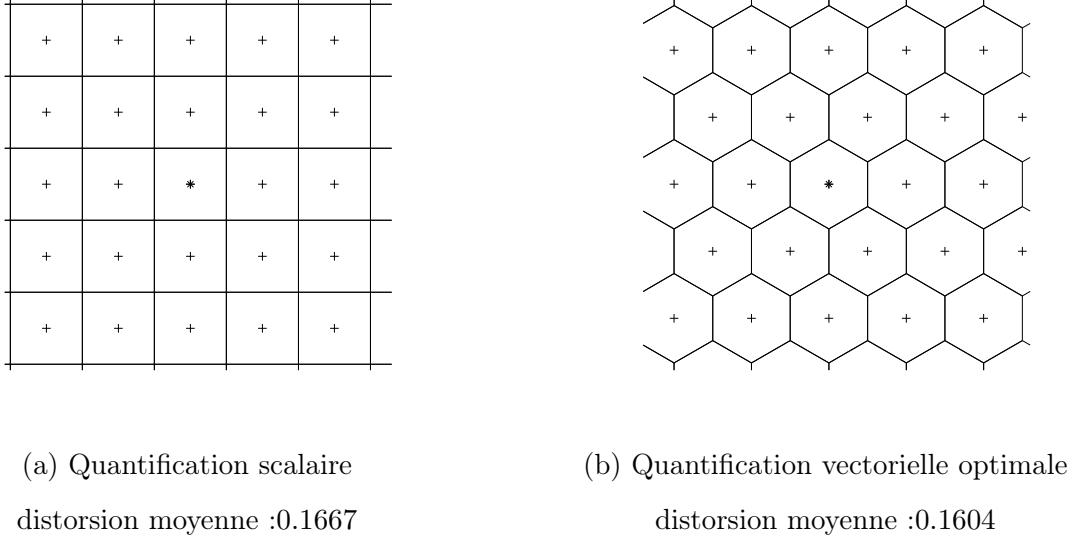


Figure 5.1 Illustration en dimension 2 du gain G_{g1} (ici égal à 0.17 dB) pour une source uniforme.

Contrairement au premier gain $G_{g1}(N)$, le deuxième gain $G_{g2}(N)$ dépend de la distribution du signal. Il peut s'interpréter par l'amélioration de l'adaptation de la région de support du dictionnaire vectoriel à la densité de probabilité de la source par rapport à celle d'un dictionnaire scalaire (*shape advantage*). La Figure 5.2 illustre l'adaptation de la quantification vectorielle à la région de forte densité de probabilité d'un signal gaussien sans mémoire. Il est possible de démontrer que $G_{g2}(N) \geq 1 \quad \forall N$. La quantification vectorielle n'apporte pas de gain $G_{g2}(N)$ si la

distribution est uniforme. Dans le cas gaussien, $G_{g2}(N)$ est égal à :

$$G_{g2}(N) = \frac{3^{3/2}}{\left(\frac{N+2}{N}\right)(N+2)/2} \quad (5.13)$$

ce qui donne 2.4 dB pour $N = 10$ et 2.8 dB pour $N \rightarrow \infty$.

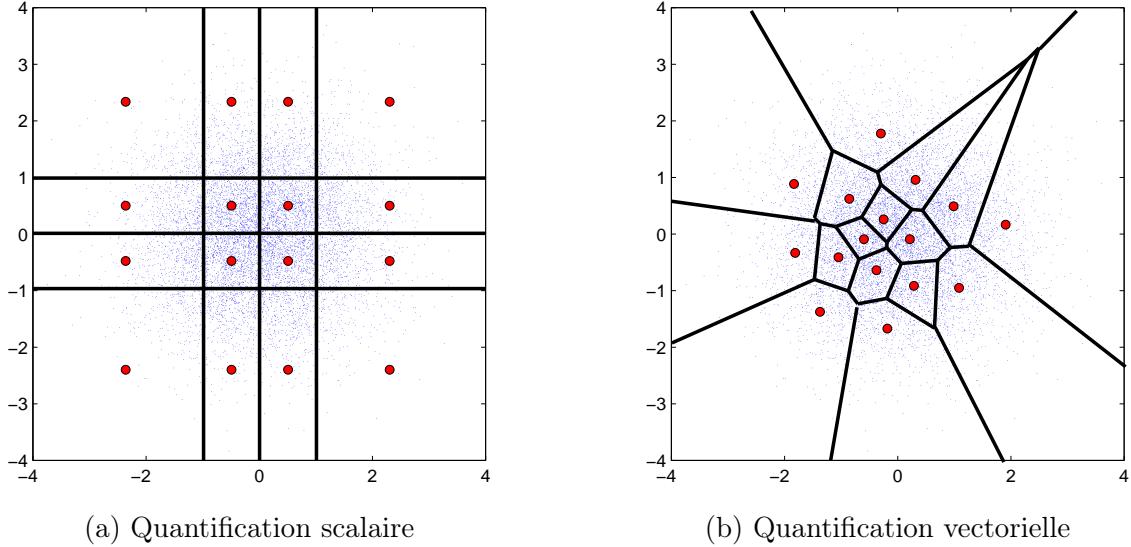


Figure 5.2 Illustration en dimension 2 du gain G_{g2} pour une source gaussienne sans mémoire ($L = 16$).

Le troisième terme $G_{g3}(N)$ tient compte de la corrélation entre les différentes composantes du vecteur, corrélation exploitée par la quantification vectorielle (*memory advantage*). Le rapport est toujours supérieur ou égal à 1 et d'autant plus important que les composantes du vecteur sont corrélées. La figure 5.3 illustre le gain en prenant le cas particulier d'un processus autorégressif d'ordre 1.

Même si on suppose après transformation que les coefficients sont complètement décorrélés ($G_{g3}(N) = 1$) la quantification vectorielle est toujours avantageuse grâce aux gains $G_{g1}(N)$ et $G_{g2}(N)$. À débit variable, les gains $G_{g2}(N)$ et $G_{g3}(N)$ de la quantification vectorielle peuvent être effacés par la quantification scalaire si elle est suivie d'un codage entropique. Par contre, la quantification vectorielle garde toujours l'avantage du gain $G_{g1}(N)$.

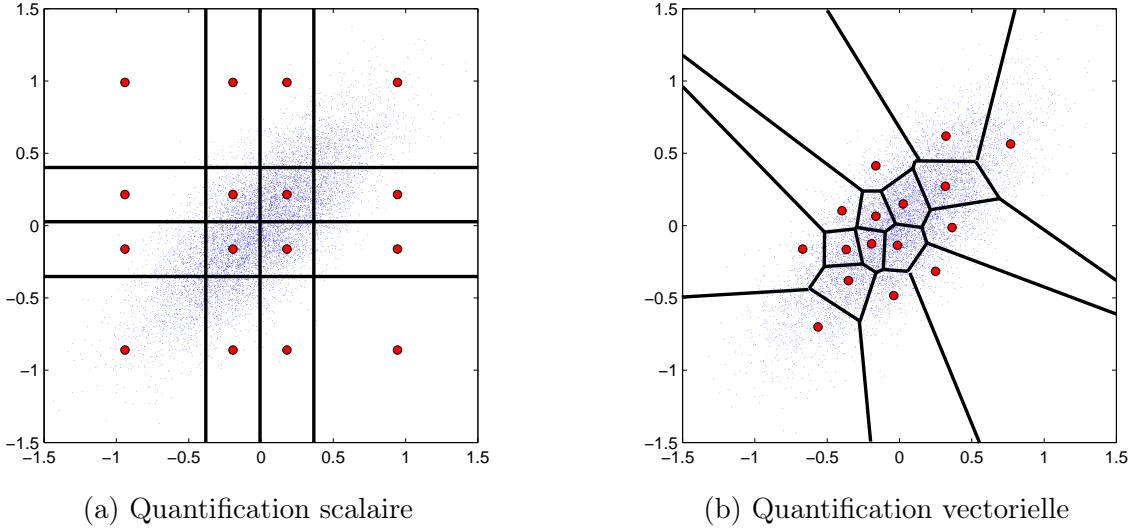


Figure 5.3 Illustration en dimension 2 du gain G_{g3} pour un processus autorégressif ($L = 16$).

5.2.3 Quantification vectorielle par contrainte

On vient de voir que la quantification vectorielle est un outil puissant en compression du signal. Par contre, son utilisation optimale nécessite des dictionnaires importants et des dimensions importantes. La quantification vectorielle stochastique est alors limitée par sa complexité lors de la recherche du meilleur représentant dans son dictionnaire et la mémoire nécessaire pour le stockage du dictionnaire.

L'espace nécessaire en scalaires (ici un scalaire est une composante du vecteur) et la complexité de recherche du plus proche voisin pour un dictionnaire de taille L et de dimension N sont tous les deux proportionnels à NL . La taille L du dictionnaire est strictement liée à la résolution R du quantificateur ainsi qu'à la dimension N du vecteur. On peut donc évaluer la complexité en espace et en temps par la même formule :

$$NL = N2^{(NR)} \quad (5.14)$$

La complexité et la taille augmentent ainsi de façon exponentielle avec la dimension N et le débit alloué R . La solution triviale consistant à réduire la dimension L de la quantification pour s'affranchir de cette barrière sacrifie la possibilité d'atteindre un meilleur gain de codage. Réduire d'autre part le débit associé à chaque composante du vecteur augmente la distorsion de la quantification. La qualité du signal synthétisé risque alors de ne pas être satisfaisante. On estime que la quantification stochastique est typiquement réalisable lorsque $NR \leq 10$ bits [14].

En partant de cette constatation, plusieurs techniques de quantification par contrainte ont été développées. Ces techniques, du fait de leur forte contrainte, ne permettent pas d'atteindre les performances théoriques de la quantification vectorielle stochastique, mais rendent possible la construction de codes efficaces de grande dimension et à débit élevé. Ce compromis est souvent favorable, car pour une minime pénalité au niveau de la distorsion moyenne, il réduit considérablement la complexité. De plus, ces techniques permettent, en exploitant des plages de débits et de dimensions plus élevées, d'obtenir des qualités inatteignables avec des quantificateurs vectoriels non contraints.

On peut citer parmi les quantifications structurées, la quantification vectorielle arborescente, la quantification vectorielle par produit cartésien (la quantification vectorielle de type forme-gain en est un cas particulier), la quantification multiétages, la quantification vectorielle par transformée ou bien la quantification algébrique. Un exposé détaillé de ces différentes méthodes se trouve dans [14]. La quantification par réseau régulier de points fait aussi partie des quantifications par contrainte. Ses propriétés algébriques font d'elle, une quantification ayant une complexité algorithmique et un coût de stockage très faibles. Comme nous allons le voir par la suite, elle peut s'adapter à un codage multidébit d'une source gaussienne ou laplacienne et devient alors très intéressante pour le codage audio.

5.3 Quantification par réseau régulier de points

5.3.1 Réseaux réguliers de points

Un réseau régulier de points Λ dans \mathbb{R}^N est défini comme l'ensemble des points qui s'obtiennent par combinaison linéaire de N vecteurs de base linéairement indépendants, v_1, v_2, \dots, v_N , avec des coefficients de proportionnalité entiers k_i :

$$\Lambda = \{y|y = \sum_{i=1}^N k_i v_i | k_i \in \mathbb{Z}, v_i \in \mathbb{R}, \forall i = 1, 2, \dots, N\} \quad (5.15)$$

Un exemple de réseau régulier de points en dimension 2 est montré à la Figure 5.4. Il est défini par deux vecteurs $v_1 = \{1, 0\}$ et $v_2 = \{0.5, \sqrt{3}/2\}$. Un réseau peut aussi s'exprimer sous forme matricielle :

$$\Lambda = \{y = kM(\Lambda) | k \in \mathbb{Z}^N\} \quad (5.16)$$

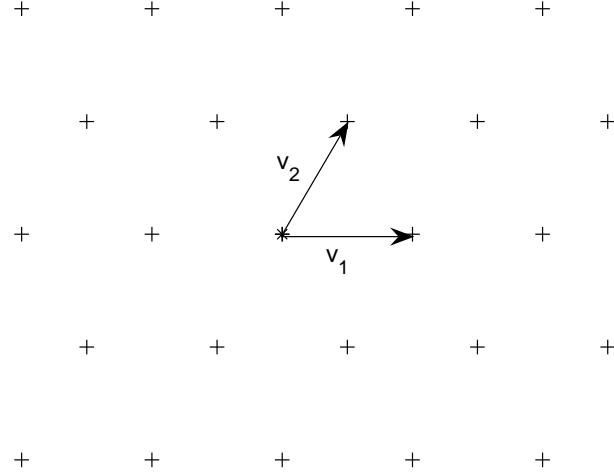


Figure 5.4 Exemple de réseau régulier de points

où k est un vecteur ligne de dimension N formé des coefficients entiers k_i et M une matrice régulière carrée de dimension $N \times N$ formée des vecteurs lignes v_i . $M(\Lambda)$ est appelée la matrice génératrice. Il existe de nombreuses propriétés algébriques qui font des réseaux de points un outil intéressant pour la quantification. On aborde ici seulement quelques définitions, plus de détails pouvant être trouvés dans [111].

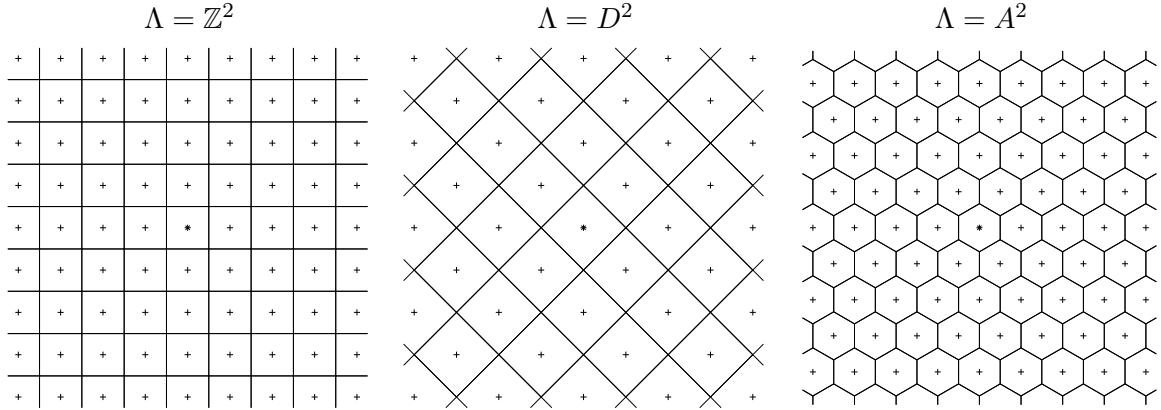
Un réseau de points est souvent défini par sa région de Voronoï à l'origine :

$$V(\Lambda) = \{x \in \mathbb{R}^N \mid \|x\|^2 < \|x - y\|^2 \quad \forall y \in \Lambda - \{0\}\} \quad (5.17)$$

Les régions de Voronoï forment des parallélétopes tous congruents. Ainsi la région relative au point y du réseau est $V(\Lambda) + y$. L'ensemble des régions de Voronoï forme alors un pavage régulier de l'espace \mathbb{R}^N . Le réseau le plus simple est \mathbb{Z}^N , associé au produit cartésien de N quantificateurs scalaires uniformes. Ses régions de Voronoï sont des N -cubes centrés sur une grille cartésienne.

Un coset $\Lambda + c$ d'un réseau Λ est un réseau géométriquement translaté de telle sorte que $c \in \mathbb{R}^N$. Un réseau dual Λ^* consiste à l'ensemble des points de \mathbb{R}^N de telle façon que le produit interne avec tout point de Λ soit entier :

$$\Lambda^* = \{x \in \mathbb{R}^N \mid xy^t \in \mathbb{Z} \quad \forall y \in \Lambda\} \quad (5.18)$$

Figure 5.5 Réseaux \mathbb{Z}^2 , D_2 et A_2 avec leurs régions de Voronoï

Avec ces définitions, il est possible de définir les réseaux les plus importants. On retrouve les familles A_N ($N \geq 2$), D_N ($N \geq 2$) et $2D_N^+$ (N pair ≥ 4), définies comme suit :

$$A_N = \{x \in \mathbb{Z}^{N+1} \mid \sum_{i=1}^{N+1} x_i = 0\} \quad (5.19)$$

$$D_N = \{x \in \mathbb{Z}^N \mid \sum_{i=1}^N x_i \text{ est pair}\} \quad (5.20)$$

$$2D_N^+ = 2D_N \bigcup \{2D_N + (1, \dots, 1)\} \quad (5.21)$$

Les réseaux \mathbb{Z}^N , D_2 et A_2 ainsi que leurs régions de Voronoï associés sont illustrés à La Figure 5.5.

Il est possible de mesurer l'efficacité d'un réseau par son gain granulaire. Pour un signal aléatoire à distribution uniforme, le gain granulaire est défini comme le rapport des moments normalisés d'ordre 2 de l'erreur de quantification d'un signal uniforme au sein d'une région de Voronoï entre le réseau \mathbb{Z}^N et du réseau de points considéré à l'ordre N . On reconnaît alors le premier gain $G_{g1}(N) = c(1)/c(N)$ de l'équation 5.10. Les performances des principaux réseaux sont rapportées au Tableau 5.1. Pour ce critère, les réseaux optimaux sont A_1 , A_2 , A_3^* , D_4 , D_5^* , E_6^* , E_7^* , $RE_8 = 2D_8^+$ (le réseau de Gosset), Λ_{16} et Λ_{24} (réseau de Leech) dans chacune de leur dimension respective. Ces réseaux sont détaillés dans [111]. Par la suite, on utilisera souvent le réseau A_2 en dimension 2 pour illustrer des concepts, alors qu'on utilisera le réseau de Gosset RE_8 en dimension 8 dans l'implémentation réelle.

Réseau (Λ)	$c(N)$	$c(1)/c(N)$ (dB)
\mathbb{Z}^N	$c(1) = 1/12$	0
A_2	0.0802	0.16
D_4	0.0766	0.25
E_8	0.0717	0.65
Λ_{16}	0.0683	0.86
Λ_{24}	0.0658	1.02
N -sphère	$1/2\pi e = 0.0585$ ($N \rightarrow \infty$)	1.53 ($N \rightarrow \infty$)

TABLEAU 5.1 Performances des réseaux importants.

5.3.2 Quantification par réseau régulier de points

La quantification par réseau régulier de points (LVQ) est simplement une quantification vectorielle dont les mots de code sont un sous-ensemble d'un réseau infini de points. Chaque LVQ est alors caractérisée par un réseau de points Λ et une région de support S , ce qui en fait une quantification à dictionnaire de taille finie dont les code-vecteurs sont alors indexables. La LVQ est alors définie comme étant l'intersection de Λ avec S :

$$LVQ(S, \Lambda) = S \cap \Lambda \quad (5.22)$$

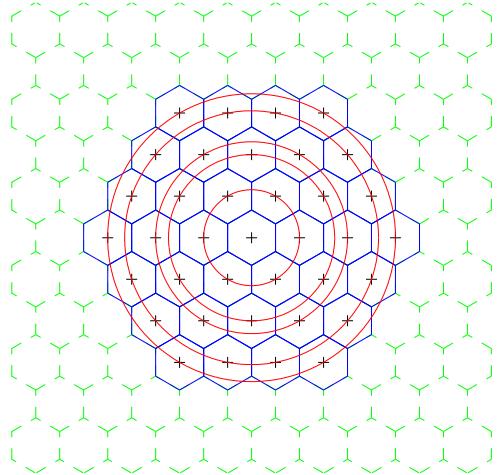
Pour optimiser les performances de la LVQ, la forme de la région de support doit être adaptée aux contours d'équiprobabilité de la source ainsi qu'à la région de plus forte densité de probabilité. De ce fait, la troncature du réseau de points joue un rôle important dans l'efficacité de la LVQ.

Région de support

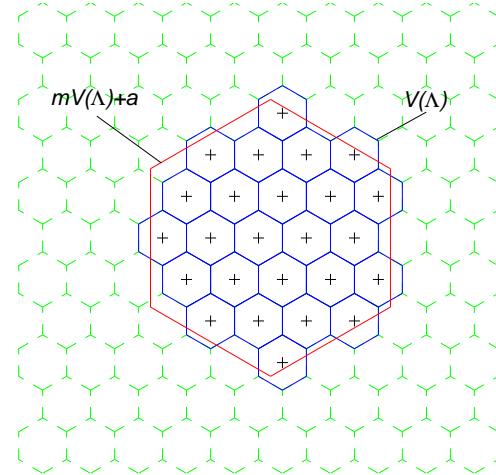
La définition de la région de support, appelée aussi *shaping*, est primordiale pour obtenir en premier lieu un dictionnaire de taille finie, mais aussi un gain de codage optimal. Un dictionnaire de taille finie peut en général être obtenu en retenant une ou plusieurs orbites du réseau ou en le tronquant par une région bornée. Une orbite est un ensemble de points se trouvant à la même distance de l'origine selon une norme définie. La quantification par orbites est une quantification de type gain-forme, dans le sens où elle code une classe de permutations à l'aide d'un vecteur directeur, appelé *leader*, définissant l'orbite, et un rang à l'intérieur de cette classe. On retrouve entre autres des régions de support sphériques et pyramidales selon la forme de l'orbite [112, 113].

Pour les réseaux tronqués, on peut citer le codage de Voronoï proposé par [114]. La zone de troncature du réseau est alors égale à la région de Voronoï dilatée et décalée, $mV(\Lambda) + a$. L'in-

dexation est algorithmique et repose sur l'opération modulo m . La troncature est alors quasi sphérique du fait de la forme du paralléléotope de la région de Voronoï. La quantification scalaire uniforme peut s'interpréter comme une LVQ associant le réseau \mathbb{Z}^N à une troncature cubique, définie par la valeur maximale des amplitudes. D'autres troncatures existent pour des contours quasi elliptiques [115, 116]. La Figure 5.6 illustre pour le réseau A_2 deux dictionnaires à contour sphérique définis par orbite et par troncature.



(a) Codage par orbites sphériques
de rayon de 0 à 3



(b) Troncature par codage de Voronoï
($m = 5$ et $a = 0.25$)

Figure 5.6 Exemples de régions de support pour le réseau A_2 .

La forme des orbites ou de la troncature doit idéalement correspondre à la forme des contours d'équiprobabilité de la source à quantifier [14]. Pour une source sans mémoire, les distributions uniformes, gaussiennes et laplaciennes ont des contours d'équiprobabilité correspondant respectivement à un hypercube, une sphère et une pyramide. Une source gaussienne vectorielle corrélée a un contour d'équiprobabilité de la forme d'une ellipse, alors que les coefficients transformés d'une image ou d'un signal de parole auront plutôt le contour d'une pyramide pondérée [117].

L'ensemble des points du dictionnaire défini par la région de support forme alors une constellation. Si le vecteur à coder se trouve dans une des régions de Voronoï de la constellation, alors la distorsion moyenne est appelée distorsion granulaire D_g . Elle est fonction directe du gain granulaire du réseau :

$$D_g = N \text{vol}(V(\Lambda))^{2/N} c(N) = N \frac{\text{vol}(V(\Lambda))^{2/N}}{c(1)} G_{g1}(N) \quad (5.23)$$

où $vol(V(\Lambda))$ est le volume de la région de Voronoï du réseau Λ . Par contre si le vecteur est en dehors de ces régions, donc de la région de support, le quantificateur est saturé. La distorsion est appelée dans ce cas distorsion de saturation, et est supérieure à la distorsion granulaire. Le dictionnaire est alors surchargé. Les Figures 5.7 (a) et (b) donnent un exemple d'un codage sans saturation et d'un codage avec saturation respectivement.

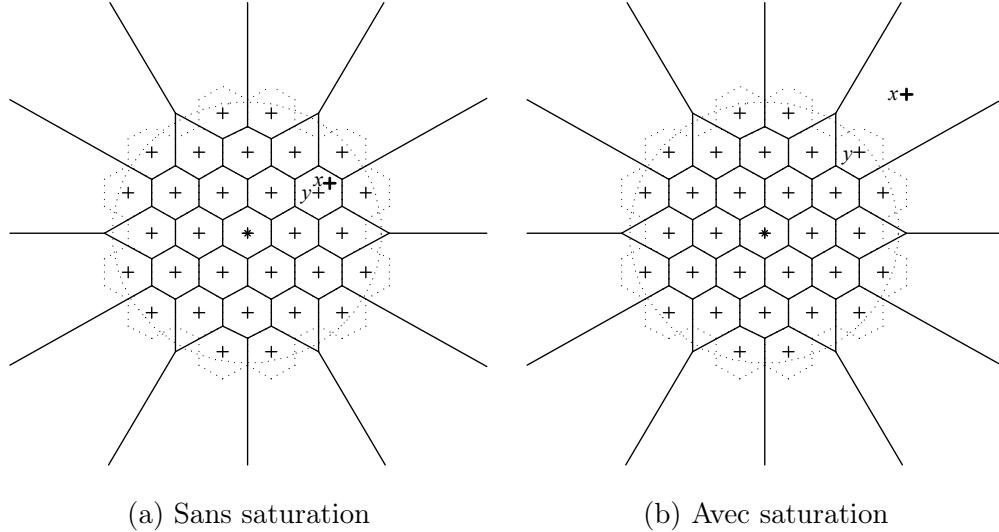


Figure 5.7 Exemple d'une quantification de x par un réseau régulier de points (a) avec saturation et (b) sans saturation.

5.3.3 LVQ en codage

Dès 1979, Gersho [109] conjecture que la quantification vectorielle optimale en haute résolution et pour une contrainte entropique a la forme d'un réseau régulier de points. Mais, c'est qu'à partir de 1981 que la LVQ va être réellement définie par les travaux de Sloane et Conway [118, 119, 114]. Ils définissent en particulier une tabulation des codes sphériques et une méthode rapide de recherche du plus proche voisin ainsi que son indexation.

Son utilisation en codage de source a été initiée en 1984 par Sayood et al. [120] pour des sources uniformes sans mémoire. L'application était alors le codage d'image fixe. Dans le même temps, Adoul et al. [112] proposent des solutions efficaces pour des distributions gaussiennes à l'aide d'une quantification sphérique. Cette quantification est appliquée avec succès au codage bas débit de la parole.

L'utilisation de la LVQ a fait l'objet de nombreux travaux pour la compression de l'image et de la vidéo. On peut signaler entre autres la quantification pyramidale introduite par Fisher [113] pour des distributions laplaciennes. Elle a été aussi employée avec succès par de nombreux chercheurs pour coder les coefficients d'une transformation en ondelettes [121, 122, 123, 124]. Par contre, son utilisation n'est pas aussi fonctionnelle que la quantification scalaire qui peut tirer profit des puissantes méthodes à raffinements successifs en associant le procédé de quantification avec soit une prédiction de structure (*zerotree prediction*) ou bien un codage contextuel par plan de bits (*bitplane coding*) comme le font les codeurs d'images tels EZW [125], SPIHT [126] ou EBCOT [127] de la norme JPEG2000. C'est à partir de cette constatation que de nouvelles techniques de quantification vectorielle à raffinements successifs ont été proposées [128, 129] étendant au cas vectoriel les algorithmes EZW et SPIHT, respectivement. Tous deux projettent chaque vecteur sur plusieurs réseaux de points en utilisant une série de codages de type gain-forme, avec des gains décroissants. Cette technique est assez complexe car elle utilise plusieurs quantifications du même vecteur. D'un autre côté, Mukherjee et Mitra [130] proposent un raffinement successif de chaque vecteur en utilisant de façon récursive un codage de Voronoï d'un réseau de points. Les dictionnaires ont alors la forme d'une région de Voronoï et sont encastrés l'un dans l'autre.

En codage audio, Adoul et al. [112] ont été les premiers à utiliser une quantification par réseau régulier de points. La quantification s'appliquait au signal résiduel de la prédiction d'un codeur parole de type RELP (*Residual Excited Linear Prediction*). Le débit total de 2.4 kbit/s permettait alors d'apporter une alternative aux vocodeurs. La LVQ a été ensuite appliquée à la modélisation par analyse-par-synthèse de l'excitation des codeurs CELP [131]. Son utilisation a permis, au même titre que la technique ACELP [34], de réduire la complexité de codage du schéma CELP originel qui utilisait une quantification stochastique. Elle a été aussi employée avec succès pour le codage de la cible TCX dans [132] à 16 kb/s pour de la parole large bande. La quantification est alors multidébit permettant ainsi de s'adapter à la variation d'amplitude des coefficients spectraux. Pour ce faire, elle utilise plusieurs dictionnaires de dimension 8 encastrés au sein d'une structure appelée EAVQ (*Embedded Algebraic Vector Quantizers*). La technique a été améliorée pour le codage TCX à 32 kbit/s [133] en introduisant l'extension de Voronoï des dictionnaires. Cette nouvelle quantification permet principalement l'utilisation de débits plus élevés en répondant au problème de saturation du quantificateur. L'extension de Voronoï permet d'adapter les dictionnaires aux diverses amplitudes des vecteurs tout en gardant constant le gain granulaire. Cette technologie est alors adaptée aussi bien pour le codage audio que pour le co-

dage de parole pour des débits faibles à moyens. Elle est utilisée dans la nouvelle recommandation AMR-WB+ [134] du 3GPP pour la transmission bas débit de contenu audio sur des réseaux sans fil de 3^egénération.

Dans notre cas, nous souhaitons utiliser une quantification vectorielle dans le domaine transformé pour des débits dans la même gamme que ceux de l'AMR-WB+. L'extension de Voronoï utilisée par ce codeur paraît alors bien adaptée à l'exception que telle qu'introduite, elle ne bénéficie pas d'une description permettant un décodage incrémental. Il est facilement envisageable de permettre un raffinement graduel au niveau de la transmission des vecteurs, comme le montre la Figure 5.8 (a). Dans cet exemple, le spectre de la synthèse au niveau du décodeur est graduellement enrichi par l'envoi progressif des différents code-vecteurs de 1 à 6. Malgré cela, nous souhaitons obtenir une granularité encore plus fine du raffinement, à l'image d'un codage par plan de bits dans le cas scalaire, tout en bénéficiant des avantages de la quantification vectorielle. On se ramène alors au cas de la Figure 5.8 (b). La description de chaque vecteur est alors formée de plusieurs code-vecteurs de différentes signifiances, graduant ainsi l'amplitude du vecteur codé. Les code-vecteurs les plus signifiants sont envoyés en premiers. Dans l'exemple donné, ils sont envoyés dans l'ordre de 1 à 22. Si la graduation des amplitudes est aussi fine que 1 bit/vecteur, on est dans le cas du codage par plan de bits. La résolution du raffinement, c.-à-d. la granularité du train binaire, est alors de 1 bit. Pour des dimensions supérieures à 1, une graduation aussi fine est beaucoup plus difficile à atteindre car il est difficile de décomposer aussi finement un vecteur de dimension $N > 1$. En utilisant les caractéristiques de l'extension de Voronoï, il est possible, comme nous allons le voir par la suite, de tirer profit du principe du raffinement graduel exposé dans le cas du codage d'image [130]. On arrive alors à une graduation des amplitudes d'une précision de N bits pour une dimension N tout en maintenant une performance de codage quasi équivalente à celle de l'extension de Voronoï originale.

5.4 LVQ à raffinements successifs

À partir des constatations précédentes, il est évident que la LVQ peut être un outil intéressant et performant pour coder un signal audio. L'utilisation de LVQ pour le codage TCX [132, 133] a démontré qu'il était possible d'utiliser efficacement une quantification par réseau de points pour le codage audio dans le domaine transformé. Dans un premier temps, on présente la quantification multidébit par extension de Voronoï proposée par [133] et utilisée dans la recommandation AMR-

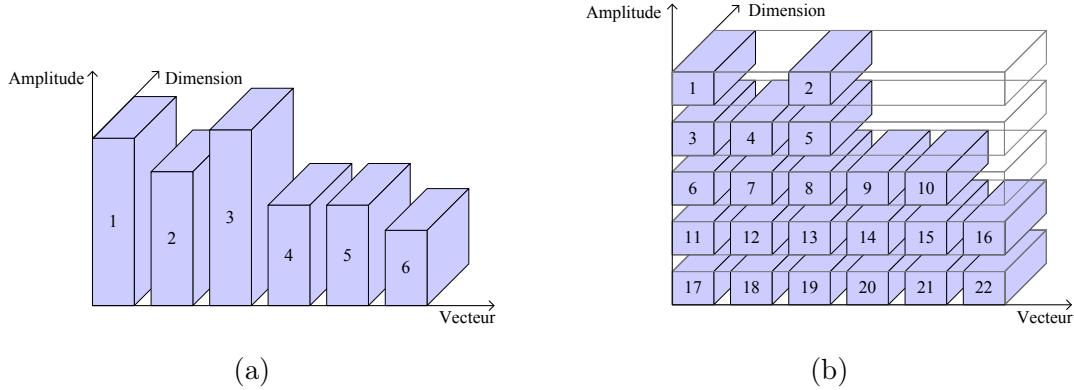


Figure 5.8 Différentes graduations et ordres de transmission du signal quantifié : (a) transmission graduelle des vecteurs du signal, (b) transmission graduelle des vecteurs du signal ainsi que de leurs amplitudes.

WB+. Dans un second temps, on introduit l'extension de Voronoï graduelle permettant de former une quantification à raffinements successifs.

5.4.1 LVQ multidébit

La quantification vectorielle algébrique imbriquée (*Embedded Algebraic Vector Quantization*, EAVQ) introduite par [132] met en œuvre le réseau régulier de points de Gosset de dimension 8, RE_8 . Elle permet un codage multidébit des vecteurs d'entrée. Les vecteurs sont en effet codés par un dictionnaire parmi 6 notés Q_n avec $n = 0, 1, 2, \dots, 5$. Les 6 dictionnaires couvrent un espace \mathbb{R}^N suffisant pour coder à 16 kbit/s la grande majorité des vecteurs d'entrée. Le dictionnaire Q_0 est constitué d'un seul code-vecteur représentant le vecteur nul. Le débit associé à l'index k d'un code-vecteur issu du dictionnaire Q_n est de $4n$ bits par vecteur, c.-à-d. $n/2$ bits par dimension. Le numéro n du dictionnaire employé est transmis en parallèle à l'aide d'un codage sans perte. La Figure 5.9 montre le principe de la quantification multidébit d'un vecteur x de dimension 8 par l'EAVQ.

Les dictionnaires de différentes tailles sont de plus imbriqués les uns dans les autres, $Q_1 \subset Q_2 \subset \dots \subset Q_3$. Chacun d'eux forme une région de support sphérique incluant plus ou moins d'orbites à partir de l'origine. Pour chaque vecteur d'entrée x , le dictionnaire le plus petit n'étant pas saturé est utilisé. L'allocation binaire est donc implicite car dépendante de x .

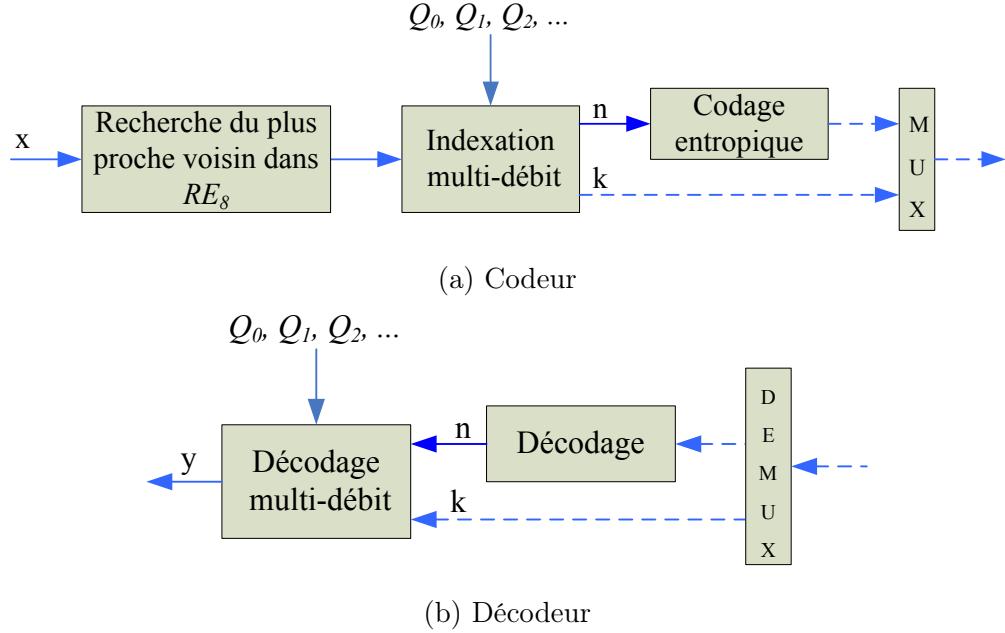


Figure 5.9 Quantification LVQ multi-débit

L’indexation dans un dictionnaire de RE_8 se fait à l’aide de vecteurs directeurs (*leaders*) absolus. Un vecteur directeur absolu représente une classe d’équivalence regroupant les code-vecteurs obtenus par permutation de ses composantes au signe près. Il permet ainsi de représenter une sphère ou une partie d’une sphère du réseau de points. L’indexation d’un code-vecteur se fait en deux étapes. L’index du code-vecteur à l’intérieur de la classe d’équivalence représentée par le vecteur directeur absolu est calculé algébriquement. Par contre, l’indexation du vecteur directeur absolu nécessite le stockage et le parcours d’une table de correspondance [115].

Cette technique de quantification a été employée avec succès pour le codage de la cible TCX à 16 kbit/s d’un signal large bande [132]. Cependant, plusieurs limitations peuvent diminuer les performances et la flexibilité de la quantification :

- La forme géométrique de la région de support est sphérique, or la source n’est pas forcément une gaussienne sans mémoire. Au contraire, même si on suppose que la source est décorrélée après transformation, la distribution réelle des coefficients transformés se rapprochent souvent plus d’une laplacienne que d’une gaussienne.
- L’allocation maximale est seulement de 20 bits par vecteur de dimension 8, ce qui peut engendrer une distorsion importante préjudiciable à la qualité globale. Cette contrainte doit être relaxée.

- La complexité et l'espace de stockage augmentent de façon exponentielle si on rajoute des dictionnaires plus grands au codage multidébit de [132]. Les deux valeurs sont fonction du nombre de vecteurs directeurs absolus utilisés.
- La saturation du quantificateur nécessite un procédé évitant les surcharges du plus grand dictionnaire Q_5 . Cela rajoute une complexité calculatoire supplémentaire au codage.
- L'imbrication des dictionnaires n'aboutit pas une quantification optimale. En effet, l'indexation des code-vecteurs communs au dictionnaire Q_n ainsi qu'aux dictionnaires de taille inférieure, augmente le débit de Q_n sans raison valable. Les vecteurs représentés par ces code-vecteurs sont normalement codés par un des dictionnaires de taille inférieure à Q_n .

5.4.2 Extension de Voronoï

L'extension de Voronoï est une technique de troncature de dictionnaires pouvant s'affranchir des limitations entrevues ci-dessus [133]. Elle permet d'étendre la région de support quasi indéfiniment, tout en gardant une complexité convenable pour la recherche du plus proche voisin dans les dictionnaires, et sans augmenter l'espace de stockage nécessaire pour l'indexation des vecteurs directeurs absolus. De plus, elle maintient constant le gain granulaire des différents quantificateurs générés.

L'extension de Voronoï se base sur les propriétés d'autosimilarité des réseaux réguliers de points et sur la théorie des vecteurs de translation appelés cosets [135]. Un réseau de points Λ peut s'exprimer en fonction du réseau $m\Lambda$ qui correspond au réseau Λ dilaté d'un facteur entier $m \geq 2$ et appelé alors sous-réseau :

$$\Lambda = m\Lambda + V(\Lambda, m\Lambda) = \bigcup_{c \in \Lambda, v \in V(\Lambda, m\Lambda)} mc + v \quad (5.24)$$

$V(\Lambda, m\Lambda)$ est un code de Voronoï de taille m^N . Il existe des algorithmes efficaces d'indexation d'un tel code [114]. Il peut être interprété comme une troncature d'un réseau Λ par une région de Voronoï du réseau dilaté $m\Lambda$ centrée aux alentours de l'origine. où a est un décalage approprié dans \mathbb{R}^N qui assure qu'aucun point de Λ ne soit à la frontière de la région, $V(m\Lambda) + a$. La constante a est souvent choisie proche de l'origine et dans un "trou" du réseau, c.-à-d. une région de \mathbb{R}^N équidistante des points avoisinants du réseau. Le code de Voronoï représente alors l'ensemble des translations (ou cosets) du sous-réseau $m\Lambda$ pour former le réseau Λ [136].

L'extension de Voronoï part du même principe des codes de Voronoï pour étendre un dictionnaire C de résolution R bit/dimension comprenant 2^{NR} code-vecteurs du réseau Λ . L'extension de Voronoï $C^{(r)}$ de C d'ordre r est alors définie comme suit :

$$C^{(r)} = 2^r C + V(\Lambda, 2^r \Lambda) = \bigcup_{c \in C, v \in V(\Lambda, 2^r \Lambda)} 2^r c + v \quad (5.25)$$

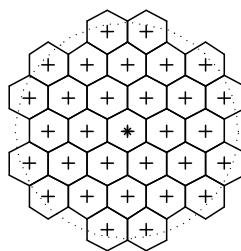
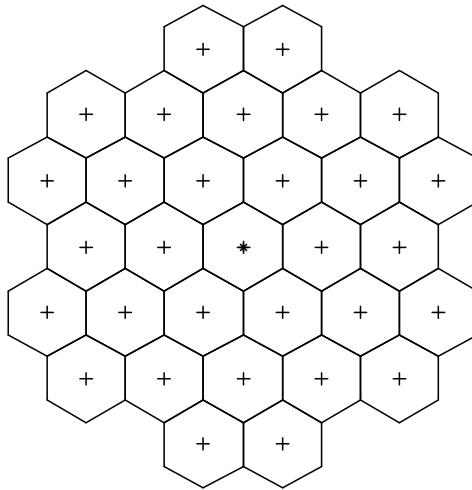
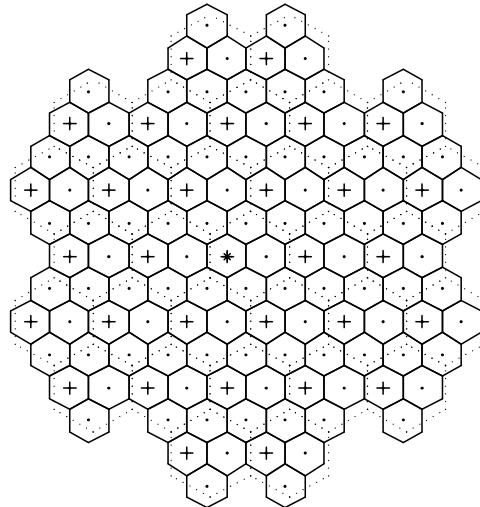
Il peut être déduit que tous les dictionnaires $C^{(r)}$ sont des sous-ensembles de Λ comprenant $2^{N(R+r)}$ code-vecteurs. Ils ont alors un débit de $R + r$ bit/dimension, R bits pour le dictionnaire de C , appelé dictionnaire de base, et r bits pour le code de Voronoï $V(\Lambda, 2^r \Lambda)$. Il est à noter que généralement le décalage a est le même pour tous les ordres r des extensions $C^{(r)}$ de C . La Figure 5.10 donne un exemple concret du processus d'extension dans le réseau A_2 . Le dictionnaire C (Figure 5.10(a)) comprend 31 points ; ses code-vecteurs sont donc indexés sur 5 bits ($R = 2.5$). En cas de saturation comme à la Figure 5.7, une solution consisterait à dilater la constellation C en la multipliant par 2. On obtient alors le dictionnaire $2C$ (Figure 5.10(b)). Les régions de Voronoï sont alors deux fois plus grandes, et la distorsion granulaire augmente d'un facteur 4. Pour maintenir le même distorsion granulaire, on applique une extension de Voronoï d'ordre 1 au dictionnaire C . À partir du dictionnaire $2C$, on ajoute un code de Voronoï $V(A_2, 2A_2)$. Le code comprend 4 points et requiert 2 bits ($r=1$). On obtient le dictionnaire $C^{(1)}$ (Figure 5.10(c)) comprenant $31 \times 4 = 124$ codés sur $5 + 2 = 7$ bits. La résolution de $C^{(1)}$ est donc égale à $R + r = 3.5$ bit/dimension.

La Figure 5.11 illustre l'expansion d'un dictionnaire par l'extension de Voronoï pour le réseau A_2 et pour les ordres 1 et 2. On peut observer que l'extension préserve la granularité du code de base C ainsi que la forme approximative de la troncature originale du réseau.

Un exemple de codage d'un vecteur x par quantification par extension de Voronoï est donné à la Figure 5.12. On remarque que l'extension de Voronoï se prête bien à un raffinement graduel à deux étages. En effet le vecteur quantifié y se décompose en deux code-vecteurs $2^r c$ et v .

$$y = 2^r c + v \quad (5.26)$$

Le vecteur $2^r c$ peut être vu comme une première estimation codée sur NR bits et v un vecteur d'amélioration nécessitant Nr bits supplémentaires. Dans le paragraphe suivant nous allons introduire une nouvelle extension de Voronoï graduelle, permettant une décomposition supplémentaire du code-vecteur v issu du code de Voronoï.

(a) Dictionnaire de base C (b) Dictionnaire dilaté $2C$ (c) Dictionnaire étendu $C^{(1)}$ Figure 5.10 Illustration d'une extension de Voronoï d'ordre 1 du dictionnaire C .

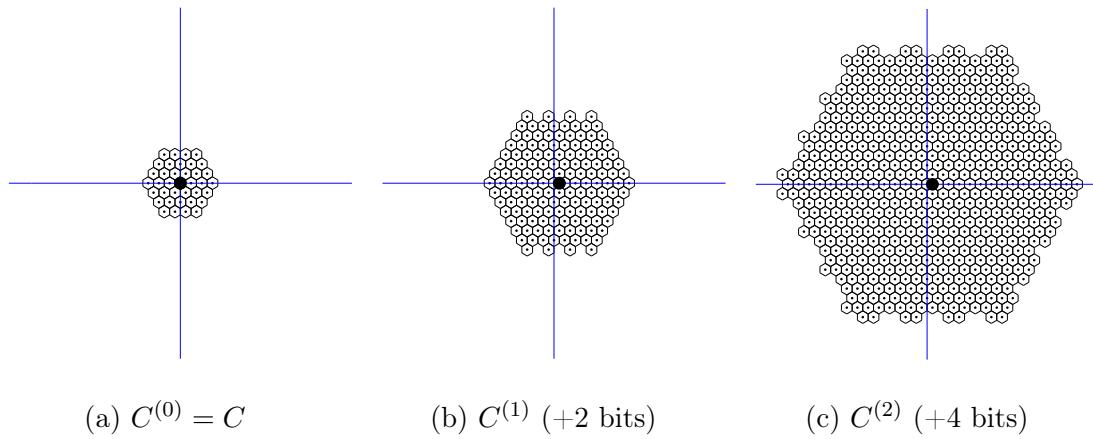


Figure 5.11 Exemple d'extension de Voronoï pour une constellation C du réseau A_2 . Le point noir indique le centroïde de la constellation.

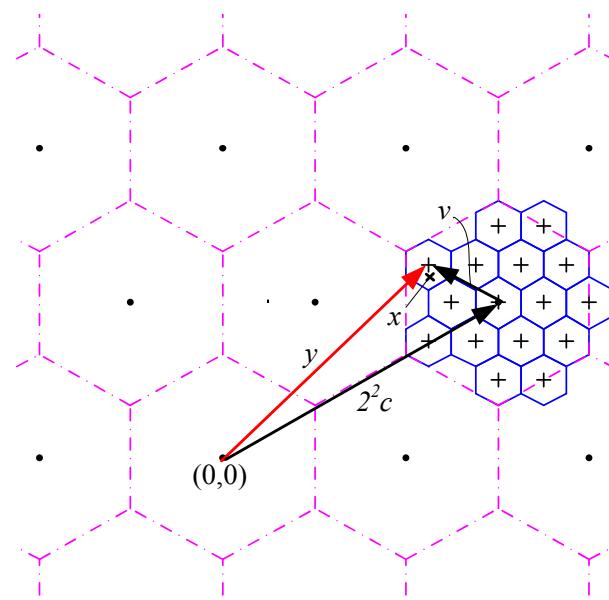


Figure 5.12 Décomposition d'un vecteur y de A_2 par l'extension de Voronoï d'ordre 2.

La recommandation AMR-WB+ utilise une telle extension de Voronoï pour former une quantification multidébit [134]. La quantification comprend 4 dictionnaires de base, Q_0 , Q_2 , Q_3 et Q_4 . Q_0 représente le vecteur nul alors que Q_1 n'est pas utilisé car un vecteur codé par un tel dictionnaire peut être avantageusement remplacé par du bruit. Les vecteurs directeurs (*leaders*) absolus définissant les dictionnaires sont donnés au Tableau 5.2. Pour chaque vecteur directeur absolu $l = \{l_1, \dots, l_8\}$ la norme d'ordre 1 P est définie comme :

$$P = |l_1| + \dots + |l_8| \quad (5.27)$$

P spécifie alors les orbites pyramidales du réseau RE_8 . Ainsi, les dictionnaires de base comprennent des orbites pyramidales successives, ce qui implique qu'ils sont optimisés pour une source laplacienne sans mémoire.

Si le vecteur d'entrée nécessite d'être codé sur des orbites supérieures à celles comprises dans Q_4 , alors une extension de Voronoï d'ordre r est appliquée alternativement aux dictionnaires Q_3 et Q_4 pour former les dictionnaires Q_{3+2r} et Q_{4+2r} respectivement. Par exemple, on a $Q_5 = Q_3^{(1)}$, $Q_6 = Q_4^{(1)}$, $Q_7 = Q_3^{(2)}$ et ainsi de suite. Le débit associé aux dictionnaires Q_n est alors de $4n$ bits ce qui fait un incrément de $1/2$ bit/dimension entre deux dictionnaires successifs. Le décalage $a = \{2, 0, 0, 0, 0, 0, 0, 0\}$ utilisé par le code de Voronoï $V(\Lambda, 2^r\Lambda)$ correspond à un trou profond dans le réseau RE_8 .

5.4.3 Extension de Voronoï Graduelle

Première définition

On introduit maintenant une nouvelle extension de Voronoï, appelée extension de Voronoï graduelle. Au lieu d'utiliser un seul vecteur v issu d'un code de Voronoï $V(\Lambda, 2^r\Lambda)$ comme à l'équation 5.25, on décompose l'information en r vecteurs v_i issus d'un même code de Voronoï $V(\Lambda, 2\Lambda)$. L'extension de Voronoï graduelle $C'^{(r)}$ de C d'ordre r est alors définie comme suit :

$$C'^{(r)} = 2^r C + 2^{r-1}V(\Lambda, 2\Lambda) + \dots + 2V(\Lambda, 2\Lambda) + V(\Lambda, 2\Lambda) \quad (5.28)$$

La Figure 5.13 illustre l'équation 5.28 en visualisant dans le réseau A_2 les contributions des différents termes si on parcourt l'équation de gauche à droite.

Propriétés

Leader absolu	P	Cardinalité	Q_0	Q_2	Q_3	Q_4
0, 0, 0, 0, 0, 0, 0, 0	0	1	•			
2, 2, 0, 0, 0, 0, 0, 0	4	112		•	•	
4, 0, 0, 0, 0, 0, 0, 0	4	16		•	•	
1, 1, 1, 1, 1, 1, 1, 1	8	128		•	•	
2, 2, 2, 0, 0, 0, 0, 0	8	1120			•	
4, 2, 2, 0, 0, 0, 0, 0	8	1344			•	
4, 4, 0, 0, 0, 0, 0, 0	8	112			•	
6, 2, 0, 0, 0, 0, 0, 0	8	224			•	
8, 0, 0, 0, 0, 0, 0, 0	8	16			•	
3, 1, 1, 1, 1, 1, 1, 1	10	1024			•	
2, 2, 2, 2, 2, 0, 0, 0	12	1792				•
3, 3, 1, 1, 1, 1, 1, 1	12	3584				•
4, 2, 2, 2, 2, 0, 0, 0	12	8960				•
4, 4, 2, 2, 0, 0, 0, 0	12	6720				•
4, 4, 4, 0, 0, 0, 0, 0	12	448				•
5, 1, 1, 1, 1, 1, 1, 1	12	1024				•
6, 2, 2, 2, 0, 0, 0, 0	12	4480				•
6, 4, 2, 0, 0, 0, 0, 0	12	2688				•
6, 6, 0, 0, 0, 0, 0, 0	12	112				•
8, 2, 2, 0, 0, 0, 0, 0	12	1344				•
8, 4, 0, 0, 0, 0, 0, 0	12	224				•
10, 2, 0, 0, 0, 0, 0, 0	12	224				•
12, 0, 0, 0, 0, 0, 0, 0	12	16				•
3, 3, 3, 1, 1, 1, 1, 1	14	7168				•
5, 3, 1, 1, 1, 1, 1, 1	14	7168				•
7, 1, 1, 1, 1, 1, 1, 1	14	1024				•
2, 2, 2, 2, 2, 2, 2, 2	16	256				•
3, 3, 3, 3, 1, 1, 1, 1	16	8960				•
4, 2, 2, 2, 2, 2, 2, 0	16	7168				•
8, 8, 0, 0, 0, 0, 0, 0	16	112				•
9, 1, 1, 1, 1, 1, 1, 1	16	1024				•
10, 6, 0, 0, 0, 0, 0, 0	16	224				•
12, 4, 0, 0, 0, 0, 0, 0	16	224				•
14, 2, 0, 0, 0, 0, 0, 0	16	224				•
16, 0, 0, 0, 0, 0, 0, 0	16	16				•
10, 10, 0, 0, 0, 0, 0, 0	20	112				•
12, 8, 0, 0, 0, 0, 0, 0	20	224				•

TABLEAU 5.2 Spécifications de Q_0, Q_2, Q_3 et Q_4 dans RE_8

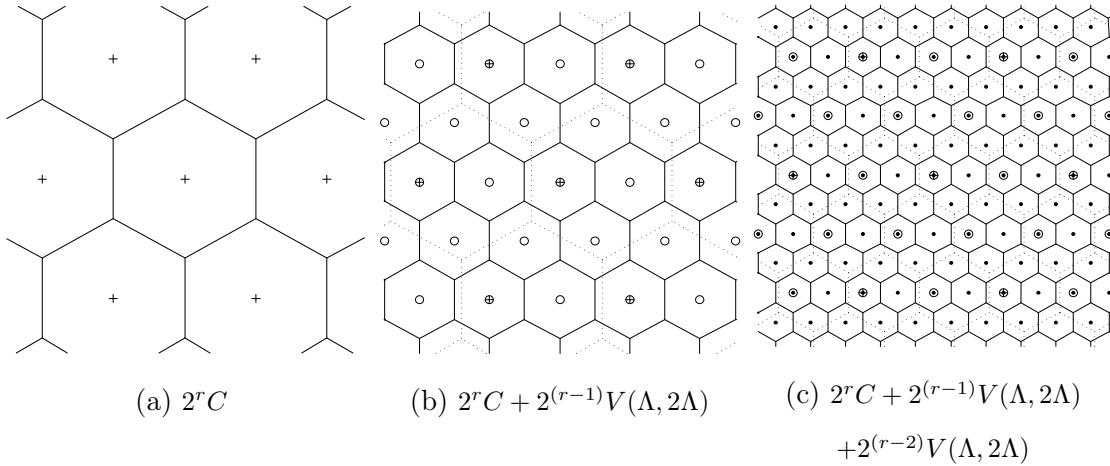


Figure 5.13 Illustration d'une extension de Voronoï graduelle dans le réseau A_2 .

Il est facilement visible que l'extension de Voronoï graduelle forme bien une constellation de Λ si on voit l'extension de Voronoï graduelle comme une succession de r extensions de Voronoï d'ordre 1.

$$C'^{(r)} = 2 \left(2 \left(\dots 2(2C + V(\Lambda, 2\Lambda)) + V(\Lambda, 2\Lambda) \dots \right) + V(\Lambda, 2\Lambda) \right) + V(\Lambda, 2\Lambda) \quad (5.29)$$

Elle hérite alors d'une partie des propriétés de l'extension de Voronoï, à savoir :

1. Elle est valable pour n'importe quel réseau régulier de points Λ .
2. Une constellation étendue $C'^{(r)}$ comprend 2^{Nr} plus de points que la constellation de base C .
3. Si une constellation C de Λ est étendue à l'aide de l'équation 5.28, alors $C'^{(r)}$ est aussi une constellation de Λ .
4. Elle conserve la granularité et la forme approximative de la constellation C .
5. Sous certaines conditions, l'extension graduelle de Voronoï produit une série de codes imbriqués avec $C \subset C'^{(1)} \subset C'^{(2)} \subset \dots$ et asymptotiquement $C'^{(+\infty)} = \Lambda$. Cette propriété est vraie par exemple si la région de troncature de C est suffisamment grande et que le décalage a de $V(\Lambda, 2\Lambda)$ est proche de l'origine.

Par contre une partie des propriétés ne sont plus satisfaites :

1. Les codes de l'extension de Voronoï graduelle ne sont plus systématiquement circonscrits à l'intérieur d'une cellule de Voronoï du réseau dilaté $2^r \Lambda$ comme c'est le cas pour l'extension

de Voronoï de l'équation 5.25. Cette différence a plusieurs incidences que ce soit au niveau des performances ou bien de l'algorithme de codage et de décodage. Nous allons par la suite détailler ces différents aspects.

2. Si C est de moyenne nulle, c'est à dire centré à l'origine, alors le centroïde $p(C'^{(r)})$ des codes $C'^{(r)}$ n'est pas égal au centroïde $p(V(\Lambda, 2^r \Lambda))$ voisin du décalage a définissant le code $V(\Lambda, 2^r \Lambda)$. Comme a est généralement proche de l'origine, alors le centroïde de l'extension de Voronoï originale $p(C^{(r)})$ est presque centré à l'origine. Pour l'extension de Voronoï graduelle le centroïde $p(C'^{(r)})$ est maintenant égal à :

$$\begin{aligned}
 p(C'^{(r)}) &= 2^{(r-1)}p(V(\Lambda, 2\Lambda)) + 2^{(r-2)}p(V(\Lambda, 2\Lambda)) \\
 &\quad + 2p(V(\Lambda, 2\Lambda)) + p(V(\Lambda, 2\Lambda)) \\
 &= (2^{(r-1)} + 2^{(r-2)} + \dots + 2 + 1)p(V(\Lambda, 2\Lambda)) \\
 &= (2^r - 1)p(V(\Lambda, 2\Lambda))
 \end{aligned} \tag{5.30}$$

Il est alors évident que le décalage par rapport à l'origine n'est plus négligeable. La Figure 5.14 illustre cette dérive du centroïde en fonction de l'ordre r de l'extension dans le cas du réseau A_2 .

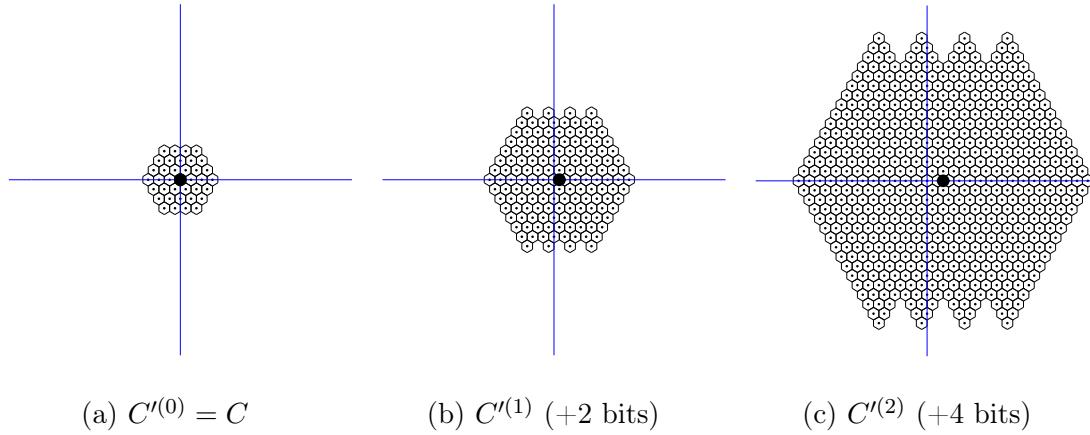


Figure 5.14 Extension de Voronoï graduelle pour une constellation C du réseau A_2 : illustration de la dérive du centroïde (point noir) du dictionnaire.

Comme l'illustre la Figure 5.15 pour le réseau A_2 , l'extension de Voronoï graduelle permet d'obtenir une décomposition plus fine à plusieurs niveaux de raffinement du vecteur y représentant le vecteur original x .

$$y = 2^r c + 2^{(r-1)}v_{r-1} + \dots + 2v_1 + v_0 \tag{5.31}$$

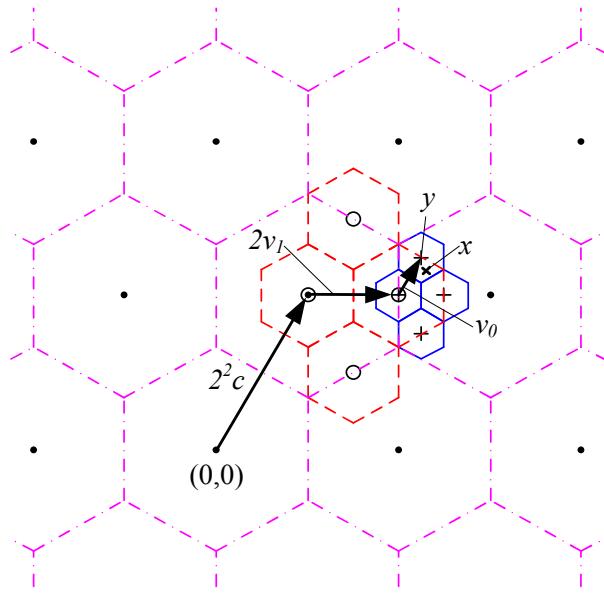


Figure 5.15 Décomposition d'un vecteur y de A_2 par l'extension de Voronoï graduelle d'ordre 2.

Deuxième définition

Pour éviter la dérive du centroïde des codes montrée à la Figure 5.14, on introduit une forme plus générique de l'extension de Voronoï graduelle. Cette fois-ci le code de Voronoï pour chaque résolution peut être différent. On obtient alors l'expression suivante :

$$C'^{(r)} = 2^r C + 2^{r-1} V_{(r-1)}(\Lambda, 2\Lambda) + \dots + 2V_1(\Lambda, 2\Lambda) + V_0(\Lambda, 2\Lambda) \quad (5.32)$$

Pour éviter une dérive du centroïde, on définit une condition particulière sur les différents codes de Voronoï de l'équation 5.32. La condition impose que l'équation 5.32 fasse intervenir seulement deux différents codes :

$$\begin{cases} V_{(r-1)}(\Lambda, 2\Lambda) = \Lambda \bigcap (V(2\Lambda) + a) \\ V_{(r-2)}(\Lambda, 2\Lambda) = \dots = V_0(\Lambda, 2\Lambda) = \Lambda \bigcap (V(2\Lambda) - a) \end{cases} \quad (5.33)$$

La Figure 5.16 montre en dimension 2 deux différents codes de Voronoï vérifiant la condition 5.32.

Les deux codes sont alors liés par la relation remarquable sur leur centroïde :

$$p(\Lambda \bigcap (V(2\Lambda) + a)) = -p(\Lambda \bigcap (V(2\Lambda) - a)) \quad (5.34)$$

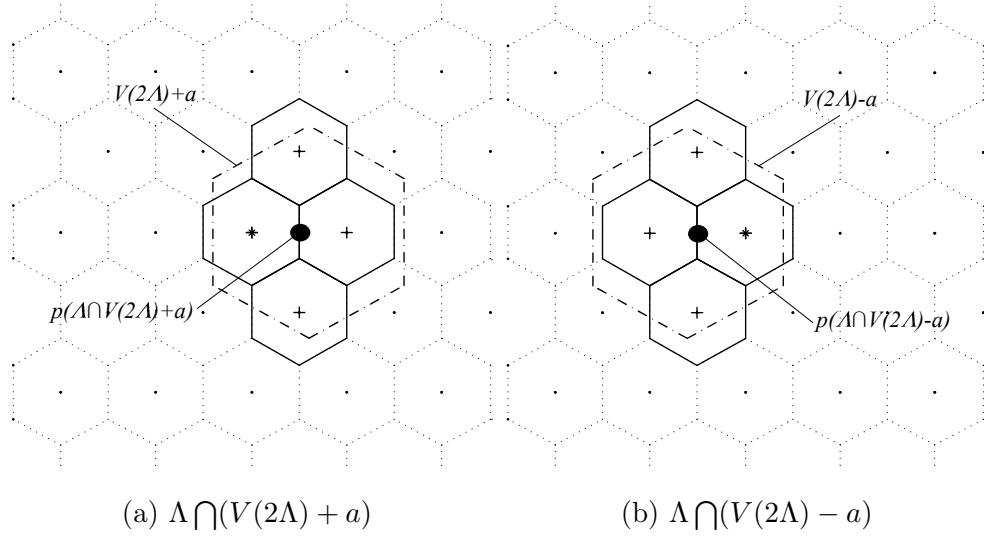


Figure 5.16 Codes de Voronoï dans A_2 répondant à la condition de l'équation 5.32 ($a = 0.6$).

En utilisant cette nouvelle extension de Voronoï graduelle, on obtient alors de nouvelles constellations $C'^{(r)}$, comme l'illustre la Figure 5.17 dans le cas de la dimension 2. La forme des constellations ne changent pas, mais le centroïde des codes s'écrit maintenant de la façon suivante :

$$\begin{aligned} p(C'^{(r)}) = & 2^{(r-1)}p(V_{(r-1)}(\Lambda, 2\Lambda)) + 2^{(r-2)}p(V_{(r-2)}(\Lambda, 2\Lambda)) + \\ & \dots + p(V_0(\Lambda, 2\Lambda)) \end{aligned} \quad (5.35)$$

Dans le cas particulier de l'équation 5.33, on obtient alors :

$$\begin{aligned} p(C'^{(r)}) = & 2^{(r-1)}p(V_{(r-1)}(\Lambda, 2\Lambda)) - 2^{(r-2)}p(V_{(r-1)}(\Lambda, 2\Lambda)) - \\ & \dots - p(V_{(r-1)}(\Lambda, 2\Lambda)) \\ = & p(V_{(r-1)}(\Lambda, 2\Lambda)) \end{aligned} \quad (5.36)$$

On obtient alors une constellation C'^r dont le centroïde est proche de l'origine pour tous ordres r comme les constellations $C^{(r)}$ issues de l'extension de Voronoï originale.

La Figure 5.18 montre la décomposition obtenue avec cette nouvelle extension de Voronoï graduelle toujours avec l'exemple du codage du vecteur x en dimension 2. On observe que le vecteur codé y est décomposé différemment qu'à la Figure 5.15. Le choix des codes de Voronoï a donc une influence sur la décomposition en code-vecteurs. Au décodage, si la description de parcours menant à y est tronquée, on obtiendra des approximations différentes du vecteur y selon les codes de Voronoï choisis. En effet, dans l'exemple donné, si le débit n'est pas suffisant pour transmettre les vecteurs v_1 et v_0 , l'approximation 2^2c de y est différente dans les deux cas. Par la suite,

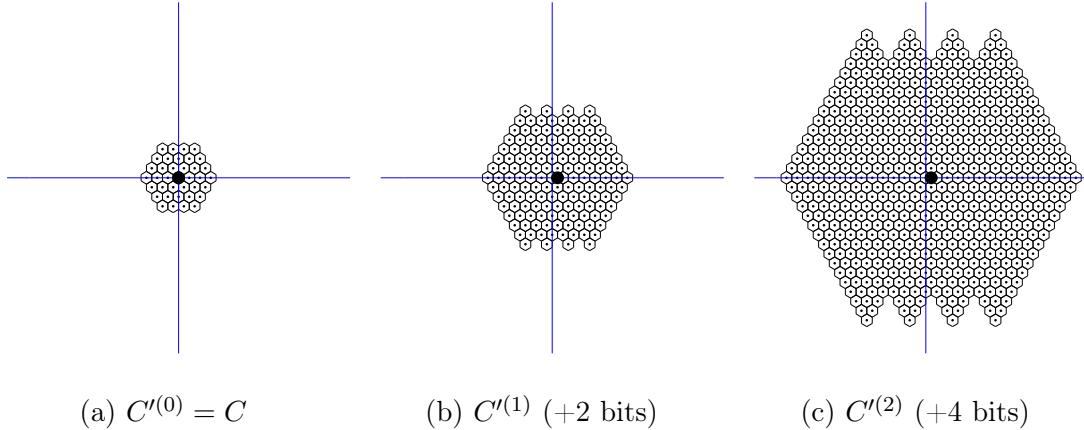


Figure 5.17 Nouvelle extension de Voronoï graduelle pour une constellation C du réseau A_2 . Le point noir indique le centroïde de la constellation.

nous verrons comment améliorer l'approximation en cas de troncature du parcours menant à y connaissant les caractéristiques des codes de Voronoï utilisés.

Algorithme de codage

Le codage par extension de Voronoï graduelle se fait selon l'Algorithme 2 de la page suivante. Il permet d'obtenir l'index du dictionnaire de base I ainsi que les différents index des codes de Voronoï k_i .

Algorithme 2 Algorithme de codage par extension de Voronoï graduelle

Entrées: $x \in \mathbb{R}^N$, $C \subset \Lambda$

Sorties: $I \in \mathbb{N}$, $k_{(r-1)}, \dots, k_0 \in \mathbb{N}$ et $r \in \mathbb{N}$

Chercher le plus proche voisin y de x dans Λ

Initialisation : $r = 0$, $c \leftarrow y$

while $c \notin C$ **do**

Chercher $v_r \in V_r(\Lambda, 2\Lambda)$ de telle façon que $c = 2\Lambda + v_r$

Calculer l'index k_r de v_r dans $V_r(\Lambda, 2\Lambda)$

$c \leftarrow \frac{1}{2}(c - v_r)$

$r \leftarrow r + 1$

end while

Calculer l'index I de c dans C

Retourner I , $k_{(r-1)}, \dots, k_0$ et r .

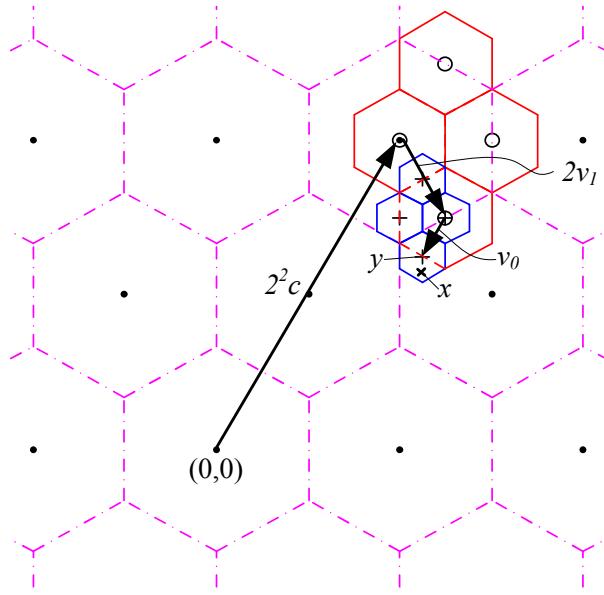


Figure 5.18 Décomposition d'un vecteur y de A_2 par la nouvelle extension de Voronoï graduelle d'ordre 2.

Il existe plusieurs algorithmes de décodage d'un réseau de points Λ permettant de trouver le plus proche voisin d'un point x quelconque de \mathbb{R}^N dans le réseau infini Λ [119]. L'appartenance de c à la constellation C et son indexation peuvent se faire par un codage par orbite en passant par des vecteurs directeurs [137]. L'indexation se fait en grande partie de façon algébrique mais passe aussi par des tables comme celle donnée en exemple au Tableau 5.2. Les algorithmes d'indexation des codes de Voronoï sont explicités dans [114]. Pour finir, il est important de noter que C peut être un ensemble de plusieurs dictionnaires de base $C = \{Q_0, Q_1, \dots\}$ comme c'est le cas pour la quantification de la recommandation AMR-WB+ [134] utilisant l'extension de Voronoï originale. L'index $I = \{n_c, k_c\}$ est alors un ensemble d'index regroupant le numéro n_c du dictionnaire de base ainsi que l'index k_c au sein du dictionnaire utilisés tous deux pour coder c . Il est alors possible de former un numéro de dictionnaire propre à l'extension de Voronoï graduelle regroupant l'information sur le dictionnaire de base choisi ainsi que son ordre d'extension $n = \{n_c, r\}$. Le même procédé est réalisable pour les index des différents code-vecteurs du dictionnaire de base et des codes de Voronoï. On obtient alors un index global $k = \{k_c, k_{(r-1)}, \dots, k_0\}$. Le couple (n, k) définit ainsi complètement le codage du vecteur x dans le réseau Λ , k étant décodable graduellement.

Algorithme de décodage

Le décodage incrémental amène une difficulté supplémentaire par rapport au décodage conventionnel. En effet, lors d'une quantification conventionnelle, l'index représente le point de reconstruction unique qui se trouve être le centroïde de la cellule de Voronoï dans laquelle se trouve le vecteur à coder. Or dans le cas de la quantification par extension de Voronoï graduelle, les différents index représentent le parcours vers le meilleur représentant y du vecteur x dans le réseau Λ . Ainsi, les points du parcours ne sont pas forcément les centroïdes des code-vecteurs qu'ils desservent. En effet, on a déjà mentionné que les codes de l'extension de Voronoï graduelle ne sont pas forcément circonscrits à l'intérieur d'une cellule de Voronoï du réseau dilaté $2^r\Lambda$. À chaque étape de décodage, il est donc nécessaire de calculer le centroïde des points atteignables du réseau à partir de l'état d'avancement à l'intérieur du parcours. On définit alors un entier R_T représentant le débit de troncature qui permet de connaître la position de la troncature à l'intérieur du parcours. R_T répond à la condition suivante :

$$0 \leq R_T \leq R + r \quad (5.37)$$

où r est l'ordre de l'extension et R la résolution du dictionnaire de base. On pose alors $r_T = R_T - R$. Si $r_T < 0$, alors le vecteur nul est décodé, aucune approximation de y peut être obtenue. Sinon, on décode un représentant $y^{(r_T)}$ associé au débit r_T qui est une approximation du vecteur y . Il s'écrit alors de la façon suivante :

$$y^{(r_T)} = \begin{cases} 0 & \text{si } r_T < 0 \\ 2^r c + 2^{(r-1)} p(V_{(r-1)}(\Lambda, 2\Lambda)) + \dots + p(V_0(\Lambda, 2\Lambda)) & \text{si } r_T = 0 \\ 2^r c + 2^{(r-1)} v_{(r-1)} + \dots + 2^{(r-r_T)} v_{(r-r_T)} + \\ 2^{(r-(r_T+1))} p(V_{(r-(r_T+1))}(\Lambda, 2\Lambda)) + \dots + p(V_0(\Lambda, 2\Lambda)) & \text{si } r_T > 0 \end{cases} \quad (5.38)$$

où c représente le code-vecteur issu de C , v_i et $p(V_i(\Lambda, 2\Lambda))$ le code-vecteur et le centroïde du code $V_i(\Lambda, 2\Lambda)$, respectivement. Il est à noter que $y^{(r)} = y$. Dans notre cas particulier répondant à la condition 5.33, l'expression se simplifie de la façon suivante :

$$y^{(r_T)} = \begin{cases} 0 & \text{si } r_T < 0 \\ 2^r c + p(V_{(r-1)}(\Lambda, 2\Lambda)) & \text{si } r_T = 0 \\ 2^r c + 2^{(r-1)} v_{(r-1)} + \dots + 2^{(r-r_T)} v_{(r-r_T)} - \\ (2^{(r-r_T)} - 1) p(V_{(r-1)}(\Lambda, 2\Lambda)) & \text{si } r_T > 0 \end{cases} \quad (5.39)$$

Les approximations successives $y^{(r_T)}$ se composent d'une part des composantes du parcours au sein du réseau Λ et d'autre part du centroïde des points du réseau atteignables à partir du parcours préalablement décodé.

$$y^{(r_T)} = y'^{(r_T)} + y''^{(r_T)} \quad (5.40)$$

avec

$$y'^{(r_T)} = \begin{cases} 0 & \text{si } r_T < 0 \\ 2^r c & \text{si } r_T = 0 \\ 2^r c + 2^{(r-1)} v_{(r-1)} + \dots + 2^{(r-r_T)} v_{(r-r_T)} & \text{si } r_T > 0 \end{cases} \quad (5.41)$$

et

$$y''^{(r_T)} = \begin{cases} 0 & \text{si } r_T \leq 0 \\ p(V_{(r-1)}(\Lambda, 2\Lambda)) & \text{si } r_T = 0 \\ -(2^{(r-r_T)} - 1)p(V_{(r-1)}(\Lambda, 2\Lambda)) & \text{si } r_T > 0 \end{cases} \quad (5.42)$$

$y''^{(r_T)}$ est appelé ajustement de décodage. La Figure 5.19 illustre le décodage en dimension 2 mettant en jeu les deux vecteur $y'^{(r_T)}$ et $y''^{(r_T)}$ afin de trouver l'approximation $y^{(r_T)}$.

Le décodage des extensions de Voronoï graduelles se fait selon l'Algorithme 3. Il comprend deux phases. La première permet de retrouver $y'^{(r_T)}$, un point du réseau Λ grâce aux index décodés. La deuxième calcule l'ajustement correspondant au centroïde $y''^{(r_T)}$ des codes de Voronoï qui découlent du point $y'^{(r_T)}$.

5.4.4 Quantification multidébit par extension de Voronoï graduelle

En codage du signal, il est généralement souhaité de minimiser l'erreur de quantification dans un domaine où le critère est l'erreur quadratique. En codage audio, ce domaine peut être un domaine fréquentiel ou temporel auquel on a appliqué un modèle perceptuel. Par exemple pour le codage TCX [62], la quantification de la cible obtenue à la suite d'une transformation et d'une pondération spectrale minimise l'erreur quadratique. Cette optimisation est faite la plupart du temps localement au sein d'une trame d'analyse. Les coefficients à quantifier sont alors regroupés en trame de N_v vecteurs de dimension N pour laquelle un budget fixe B_{tot} est alloué. Si la distribution des vecteurs est de moyenne nulle et en plus gaussienne, la fonction débit-distorsion

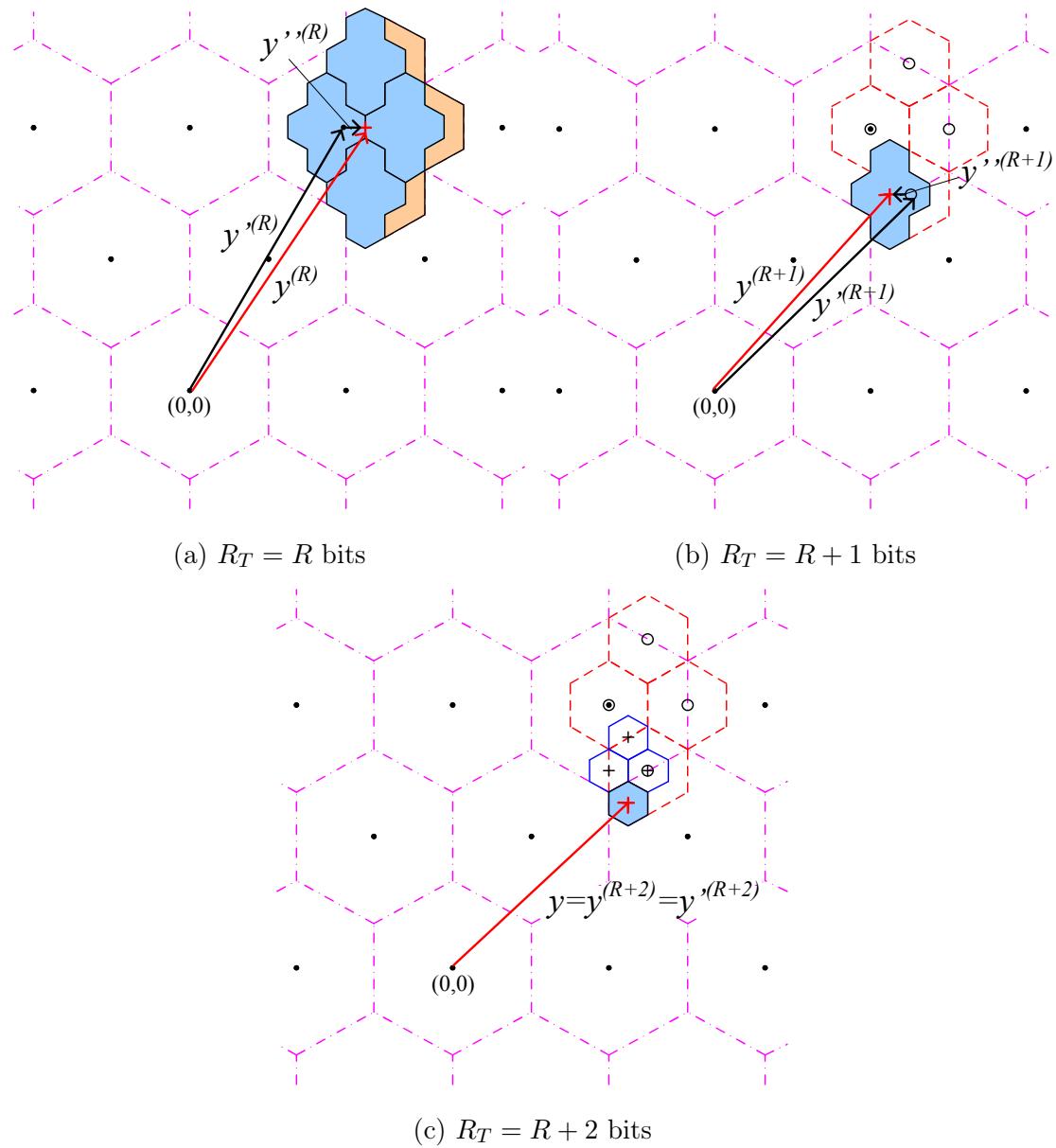


Figure 5.19 Décodage par raffinements successifs d'une extension de Voronoï graduelle dans le réseau A_2 .

Algorithme 3 Algorithme de décodage des extensions de Voronoï graduelles

Entrées: $I \in \mathbb{N}$, $k_{(r-1)}, \dots, k_{(r-r_T)} \in \mathbb{N}$, $r_T \in \mathbb{N}$ et $r \in \mathbb{N}$

Sorties: $y \in \mathbb{Z}^N$

if $r_T < 0$ **then**

$y \leftarrow \{0, \dots, 0\}$

else

 Chercher le mot de code $c \in C$ d'index I

$y \leftarrow 2^r c$

$i \leftarrow 1$

while $i \leq r_T$ **do**

 Chercher le code-vecteur $v_{(r-i)} \in V_{(r-i)}(\Lambda, 2\Lambda)$ d'index $k_{(r-i)}$

$y \leftarrow y + 2^{(r-i)} v_{(r-i)}$

$i \leftarrow i + 1$

end while

while $i \leq r$ **do**

$y \leftarrow y + 2^{(r-i)} p(V_{(r-i)}(\Lambda, 2\Lambda))$

$i \leftarrow i + 1$

end while

end if

est minimale lorsque la distorsion totale est distribuée suivant le principe du remplissage inverse des eaux (*reverse water-filling*) [138]. La distorsion D_i associée à chaque vecteur X_i devra alors être de la forme, $D_i = \min(\lambda, \sigma_{X_i}^2)$, où λ est le “niveau d’eau” et $\sigma_{X_i}^2$ la variance du vecteur X_i . Un exemple est donné à la Figure 5.20. Si la puissance de la source X_i dépasse le seuil λ , alors la distorsion est fixé à λ . Sinon, aucun bit n’est alloué à cette source si bien que la distorsion est équivalente à la variance de la source $\sigma_{X_i}^2$.

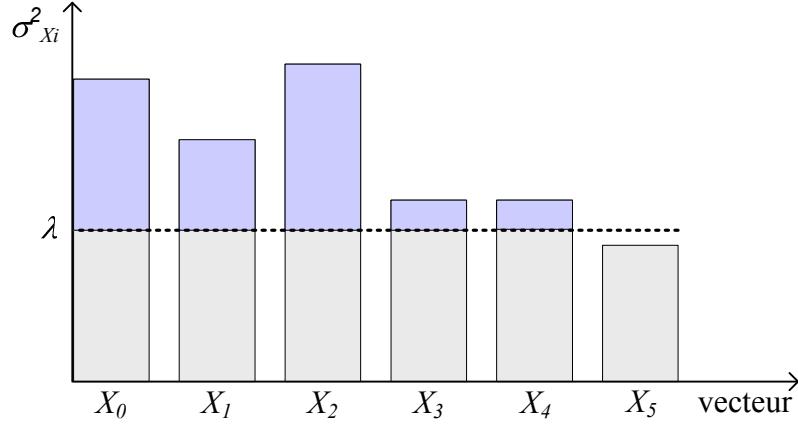


Figure 5.20 Allocation des ressources selon le remplissage inverse des eaux.

Dans notre cas on cherche à minimiser λ de telle façon que le débit total R_{tot} soit contraint par le budget alloué B_{tot} :

$$R_{tot} = \sum_{i=1}^{N_v} (NR(k_i) + R(n_i)) = \sum_{i=1}^{N_v} (N(R_i + r_i) + R(n_{ci}, r_i)) \leq B_{tot} \quad (5.43)$$

où k_i sont les index de la quantification, n_i est le numéro du dictionnaire et l’ordre de l’extension, et $R(k_i)$ et $R(n_i)$ leur débit associé. $R(k_i)$ est formé du débit R_i propre au dictionnaire de base et du débit r_i produit par les codes de Voronoï. $R(n_i) = R(n_{ci}, r_i)$ est le débit nécessaire pour transmettre le numéro n_{ci} du dictionnaire utilisé dans C ainsi que l’ordre de l’extension r_i de Voronoï.

En partant du principe du remplissage inverse des eaux, nous allons présenter deux techniques d’allocation du budget de bit applicable à l’extension de Voronoï graduelle. La première a été introduite pour la quantification par extension de Voronoï dans le standard AMR-WB+ [67] et peut s’appliquer de la même façon à notre quantification. Elle consiste à fixer λ par l’intermédiaire d’un gain global g . Ce gain contrôle alors l’allocation et la distorsion. La quantification est réalisée

après normalisation des vecteurs par le gain global g . La seconde solution est propre à l'extension de Voronoï graduelle. Elle consiste à quantifier par l'extension de Voronoï graduelle les vecteurs non-normalisés, puis à tronquer leur description selon le remplissage inverse des eaux pour rentrer dans le budget de bits.

L'allocation par gain global

L'allocation par un gain global permet de contrôler implicitement l'allocation binaire ainsi que la distorsion engendrée grâce à une quantification gain-forme. Dans la recommandation AMR-WB+ [67], un facteur d'échelle $1/g$ est appliqué au codeur à la cible à quantifier X pour former une cible normalisée X' . C'est cette nouvelle cible qui est quantifiée dans le réseau RE_8 . Le facteur inverse g est appliqué au décodeur à la cible normalisée reconstruite \hat{X}' . Plus g est grand, plus le débit associé est faible. Il est estimé d'après les énergies des N_v vecteurs de telle façon que ces derniers soient codés avec le budget de bit total B_{tot} . Bien sûr g n'étant qu'une estimation, il se peut que le budget soit dépassé. Dans ce cas, un certain nombre de vecteur est mis à zéro jusqu'au moment où le nombre de bits total généré R_{tot} soit égal ou en dessous du budget B_{tot} . On peut quand même dire que l'allocation a lieu avant la quantification comme le montre la Figure 5.21.

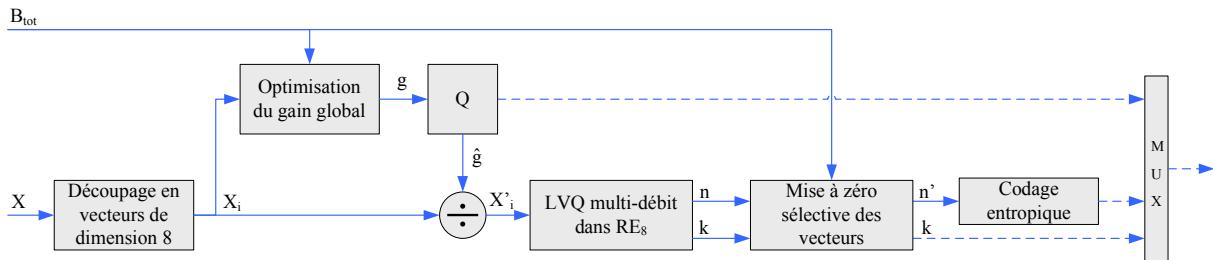


Figure 5.21 Contrôle par un gain global du débit de sortie du codage LVQ multidébit.

L'allocation par troncature

L'allocation par troncature est un procédé de gestion du budget de bits qui se fait au contraire à la suite de la quantification comme le montre la Figure 5.22. Dans un premier temps, la cible non-normalisée X est quantifiée par une LVQ par extension de Voronoï graduelle. C'est dans un second temps, que l'allocation est effectuée par troncature des différents code-vecteurs générés.

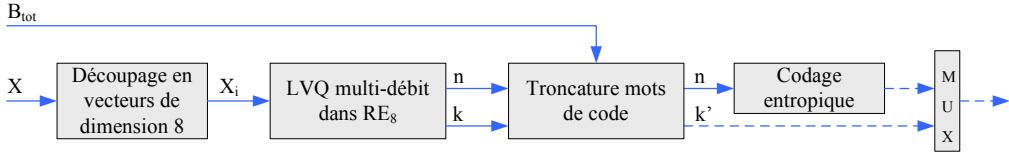


Figure 5.22 Contrôle par troncature du débit de sortie du codage LVQ multi-débit

L’allocation consiste en particulier à calculer un point de troncature des code-vecteurs générés pour que le débit total R_{tot} rentre dans le budget alloué B_{tot} . Le procédé est possible du fait de la propriété encastrée du train binaire. La cible est initialement codée par la quantification par extension de Voronoï graduelle avec un débit $R_i + r_i$ bit/dimension pour chaque vecteur X_i . On cherche alors le débit minium r_λ , représentant le niveau des eaux, devant être retiré à chaque descriptions des vecteurs initialement codés, pour rentrer dans le budget total B_{tot} .

$$\min_{r_\lambda \in \mathbb{R}_+} \sum_{i=0}^{N_v-1} [N(f(R_i) + f(r_i - g(r_\lambda)) + R(f(n_{ci}), f(r_i))] \leq B_{tot} \quad (5.44)$$

avec

$$f(x) = \begin{cases} x & \text{si } r_i - g(r_\lambda) \geq 0 \\ 0 & \text{sinon} \end{cases} \quad (5.45)$$

et

$$g(r_\lambda) = \begin{cases} \lfloor r_\lambda \rfloor & \text{si } i < N_v - \lceil N_v(r_\lambda - \lfloor r_\lambda \rfloor) \rceil \\ \lceil r_\lambda \rceil & \text{sinon} \end{cases} \quad (5.46)$$

$R(f(n_{ci}), f(r_i))$ est le débit nécessaire pour coder le numéro du dictionnaire de base ainsi que son ordre d’extension, sachant que le couple $(0, 0)$ correspond au vecteur nul. Le débit niveau des eaux r_λ peut être trouvé par une simple recherche par dichotomie. La transmission du r_λ obtenu n’est pas obligatoire, car le décodeur gère automatiquement toute troncature du train binaire.

La fonction $g()$ de l’équation 5.46 permet de définir le nombre de bits à tronquer aux différentes descriptions des vecteurs X_i . Elle permet entre autres de gérer les valeurs fractionnaires de r_λ . Si r_λ est à valeur entière alors $N \times r_\lambda$ bits seront retirés à toutes les descriptions des N_v vecteurs de dimension N . Si r_λ est fractionnaire, alors $N \times \lfloor r_\lambda \rfloor$ bits seront retirés à toutes les descriptions des N_v vecteurs, et en plus N bits supplémentaires seront retirés aux derniers $\lceil N_v(r_\lambda - \lfloor r_\lambda \rfloor) \rceil$ vecteurs de la trame. La Figure 5.23 illustre avec un exemple, la procédure d’allocation par

troncature. Toutes les descriptions au-dessus de la limite $g(r_y)$ sont envoyées dans l'ordre indiqué par les numéro de 1 à 9. L'ordre de transmission est entièrement déterminé par la fonction $g()$ si on augmente progressivement r_λ . Il est possible de définir d'autres fonctions $g()$ que celle présentée à l'équation 5.46. Les descriptions des vecteurs X_i se trouvant en-dessous de la limite $g(r_y)$, ne sont pas émises pour que le débit total R_{tot} rentre dans le budget B_{tot} . De plus, toutes les descriptions au-dessus de la limite r_i correspondent à des code-vecteurs issus du dictionnaire de base C . Les autres descriptions correspondent à des code-vecteurs issus de codes de Voronoï. Enfin il à noter que si $r_{Ti} = r_i - g(r_\lambda) < 0$, alors aucune description n'est envoyée pour le vecteur X_i , comme il est indiqué à l'équation 5.38.

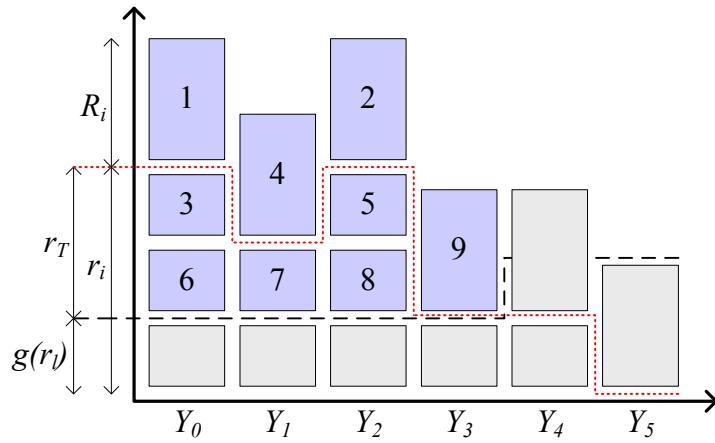


Figure 5.23 Exemple d'allocation par troncature avec détermination de l'ordre de transmission.

Choix du dictionnaire et de l'allocation

Le choix du ou des dictionnaires C a une grande influence sur les performances de la quantification par extension de Voronoï graduelle. Plus le dictionnaire est grand, plus la région de support du réseau Λ est susceptible d'être bien définie et adaptée à la source. Par contre l'index des code-vecteurs du dictionnaire C ne peut être fractionnable. Ainsi, plus le dictionnaire est grand, plus la granularité du train binaire est grossière. Dans l'autre cas extrême, où C représente uniquement l'origine, le débit alloué au dictionnaire de base est alors nul et le train binaire constitué uniquement de code-vecteurs issus de codes de Voronoï a une granularité aussi fine que 1 bit/dimension sur toute sa longueur. Par contre dans ce cas-ci, la région de support ne dépend plus que des codes de Voronoï, et est la plupart du temps mal adaptée à la source, ce qui diminue

la performance globale. Le compromis entre l'efficacité et la granularité va dicter le choix du ou des dictionnaires de base C .

Le choix de l'allocation va dépendre en grande partie du choix du dictionnaire de base C . Si C est de grande taille, le nombre de bits $g(r_\lambda)$ à tronquer calculé par l'allocation par troncature risquent d'empiéter sur l'index du dictionnaire de base C . On perd alors beaucoup d'information, car cet index n'est pas fractionnable. Cette raison impose pour une allocation par troncature quasi optimale, de choisir un dictionnaire de base C de très petite taille, voire réduit au point d'origine. Il à noter aussi que l'allocation par troncature n'est pas optimale au sens du remplissage inverse des eaux lorsque r_λ n'est pas à valeur entière, du fait que la distorsion n'est pas uniforme sur l'ensemble des vecteurs de la trame. L'utilisation de l'allocation par gain global assure dans tous les cas une allocation optimale. Par contre il est nécessaire de transmettre au décodeur le gain utilisé.

5.4.5 Génération du train binaire

La génération du train binaire telle que faite dans le standard AMR-WB+ [116] ne se prête pas du tout au décodage incrémental de la source. Il est donc nécessaire de modifier la construction du train binaire afin de favoriser la transmission graduelle des amplitudes des vecteurs codés Y_i . La fonction $g()$ définie préalablement par l'équation 5.46 est un exemple d'ordonnancement possible. Nous présentons ici un ordonnancement du train binaire plus évolué permettant de transmettre progressivement ainsi bien les index k de la quantification que les numéros de dictionnaire n , toujours en suivant le principe du remplissage inverse des eaux.

Un exemple de train binaire produit est montré à la Figure 5.24 lorsqu'on utilise les mêmes dictionnaires de base que ceux utilisés par le standard AMR-WB+, Q_0, Q_1, Q_2, Q_3 et Q_4 (cf. Tableau 5.2). Uniquement Q_3 et Q_4 sont utilisés par l'extension de Voronoï graduelle, et ce alternativement pour former les dictionnaires Q_{3+2r} et Q_{4+2r} . Dans cet exemple le spectre est composé de $N_v = 8$ vecteurs X_i de dimension $N = 8$. Les vecteurs codés sont transmis dans l'ordre décroissant du numéro n du dictionnaire utilisé. Le train binaire transmet les descriptions du dictionnaire le plus grand $n = n_{\max} = \max_{i=\{0, \dots, N_v\}}(n_i,)$ au dictionnaire le plus petit $n = 2$. La valeur n_{\max} est la première valeur transmise. Par la suite et pour chaque numéro de dictionnaire $n = n_c + 2r$, on code tout d'abord la position et le nombre de vecteurs X_i qui sont codés par le dictionnaire n . On utilise pour ce faire des vecteurs de position ainsi qu'un vecteur d'arrêt, le

vecteur nul dans notre cas. Ensuite, on transmet tous les index k_c des code-vecteurs c issus du dictionnaire de base Q_{n_c} à la résolution 2^r , ainsi que les index $k_{(r)}$ des code-vecteurs $v_{(r)}$ issus du code de Voronoï $V_{(r)}(\Lambda, 2\Lambda)$ à la résolution 2^r et provenant d'une extension d'ordre supérieur à r du dictionnaire Q_{n_c} . Les positions des index $k_{(r)}$ des codes de Voronoï dans le spectre sont déduits des positions des index k_c préalablement codées dans les étages n supérieurs. Il est à noter que les vecteurs codés par Q_0 ne sont pas explicitement transmis mais déduit des positions restantes du spectre non codées après l'étage $n = 2$. Les vecteurs de position et le vecteur d'arrêt peuvent être codés par un codage entropique, comme le codage de Huffman.

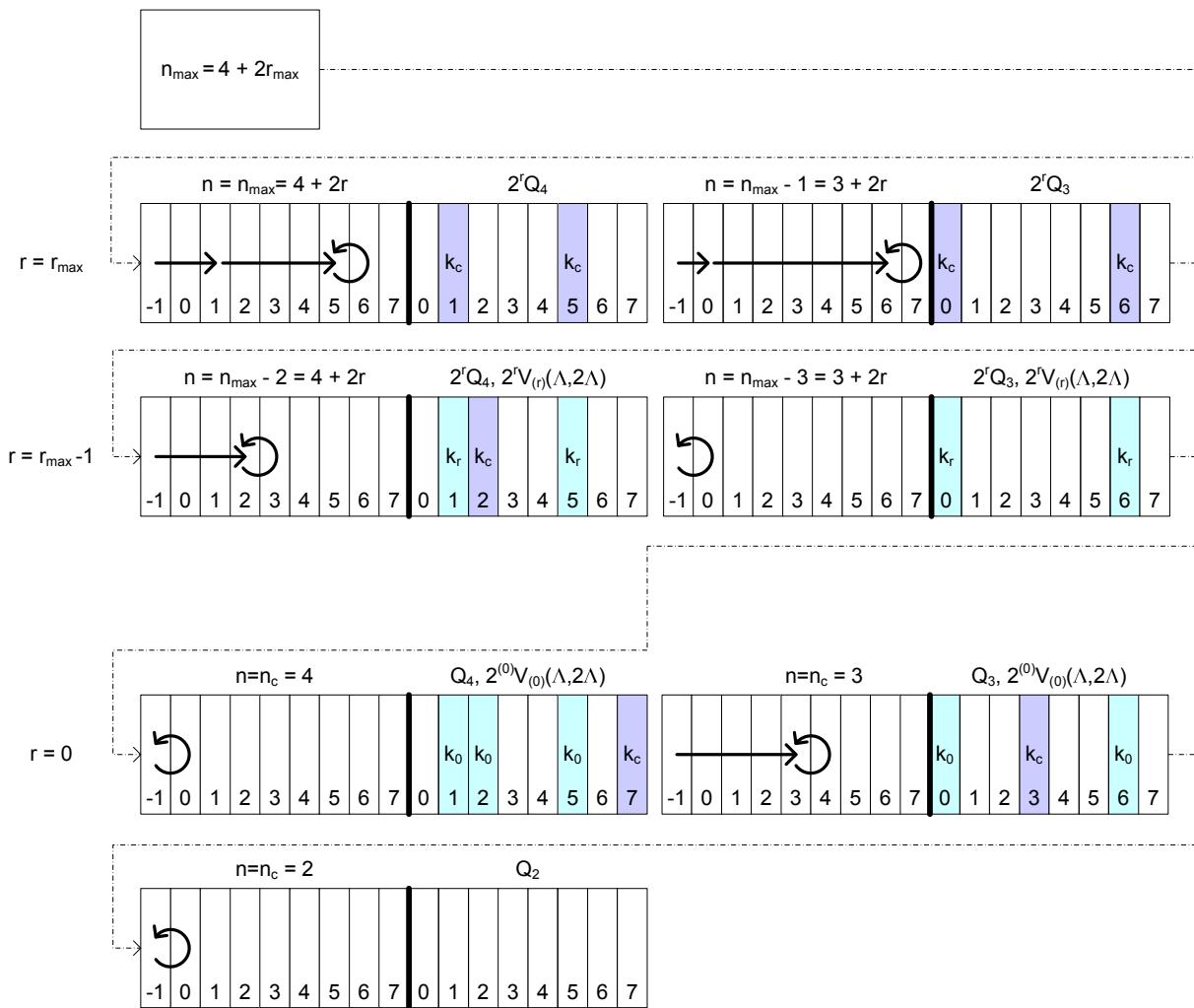


Figure 5.24 Organisation du train binaire généré pour une transmission graduelle des amplitudes des vecteurs codés par une quantification par extension de Voronoï graduelle.

5.5 Performances des quantifications multidébit

5.5.1 Cas d'une source gaussienne

Nous avons évalué dans un premier temps la quantification à raffinements successifs proposée à la section 5.4 dans le cas d'une source gaussienne discrète et sans mémoire (i.i.d), de moyenne nulle et de variance unité. L'évaluation s'est faite en dimension 8 à l'aide du réseau régulier de points RE_8 . Afin de tester l'influence du choix des dictionnaires de base, nous avons évalué deux jeux de dictionnaires. Le premier jeu utilise simplement le dictionnaire de base nul, $C = \{0, 0, 0, 0, 0, 0, 0, 0\}$. L'espace est entièrement recouvert par des codes de Voronoï de 1 bit/dimension. Le codage multidébit est alors à débits entiers par dimension. Le second jeu de dictionnaires de base utilise les mêmes dictionnaires que la quantification multidébit du standard AMR-WB+. Ainsi, quatre dictionnaires de base Q_0 , Q_2 , Q_3 et Q_4 sont utilisés. Seulement Q_3 et Q_4 sont étendus par l'extension de Voronoï graduelle, et de façon alternative, afin d'obtenir cette fois-ci des débits avec un incrément de 1/2 bit/dimension.

On utilise le système de quantification gain-forme décrit à la Figure 5.25, où le vecteur x de dimension 8 est quantifié en vecteur y . Le gain global g permet de contrôler le débit de codage généré. Le facteur g est supposé être connu au codeur comme au décodeur, il n'est donc pas transmis. Le débit généré est alors fluctuant d'une trame à l'autre, mais g est fixé pour obtenir un débit moyen désiré. De plus, le débit de décodage pour être contrôlé au décodeur par la procédure de troncature $g()$ de l'équation 5.46 à l'aide du débit niveau des eaux r_λ .

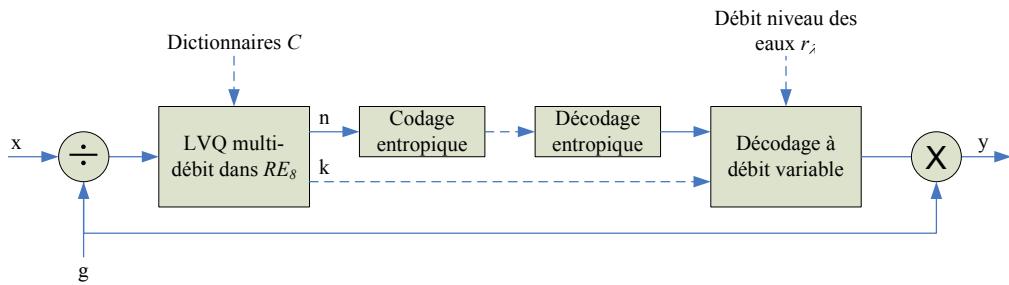


Figure 5.25 Système de quantification LVQ multi-débit dans RE_8 avec décodage à débit variable.

Dans chacun des cas, deux mesures sont comparées : les performances à débit fixe et les performances à débit variable. Pour le débit fixe, r_λ est mis à 0, et le débit de codage et de décodage est le même. Les différents débits sont alors atteignables par modification du gain global g . Il

faut autant de codages de la source que de débits. Pour le débit variable, le gain global est fixe ($g = 1/40$), et le même train binaire est utilisé pour obtenir les différents débits de décodage en faisant varier r_λ . Dans ce cas, un seul codage de la source suffit.

Pour chaque mesure, 10^6 vecteurs source x de dimension 8 sont quantifiés. Le débit moyen par dimension $R = \frac{1}{8}E[R(n)+R(k)]$, où $R(n)$ et $R(k)$ sont les nombres de bits utilisés pour représenter n et k respectivement, est alors associé à la distorsion moyenne $D = \frac{1}{8}E[||x - y||^2]$. Cette mesure est comparée à la borne inférieure débit-distorsion $D(R)$ dans le cas gaussien, appelée limite de Shannon :

$$D(R) = \sigma_x^2 2^{-2R} \quad (5.47)$$

$$D_{SNR}(R) = 20R \log_{10} 2 \approx 6.02R \quad (5.48)$$

On considère que les numéros de dictionnaire n sont codés par un codage entropique générant un débit minimal $R(n)$ dont l'espérance est égale à :

$$E[R(n)] = - \sum_{n=0}^{n_{max}} p(n) \log_2(p(n)) \quad (5.49)$$

où $p()$ est la fonction de probabilité associée aux numéros de dictionnaire et n_{max} le numéro de dictionnaire maximal.

Performance avec un dictionnaire de base nul

On utilise seulement le dictionnaire nul se limitant au vecteur nul en plus des codes de Voronoï $V_i(\Lambda, 2\Lambda)$. On obtient alors une quantification multi-débit à débits entiers par dimension. La description des vecteurs peut se décomposer entièrement en une série de codes de 1 bit/dimension. Il est à noter que cette quantification ne nécessite aucun stockage.

Les Figures 5.26 de (a) à (c) illustrent les résultats de l'extension de Voronoï graduelle lorsqu'elle est utilisée à débit fixe ($r_\lambda = 0$) et à débit variable ($g = 1/40$). Le débit fixe est à débit équivalent la plus performante des solutions car elle correspond pour les dictionnaires données à la quantification optimale. L'écart des performances entre le débit fixe et la limite de Shannon est quasi-constante pour l'ensemble des débits aux alentours de 3.7 dB. La Figure 5.26 (a) montre les performances du débit variable sans réajustement y'' lors du décodage. On voit bien la perte de performance par rapport au décodage avec réajustement à la Figure 5.26 (b). Dans ce cas les performances du débit variable se rapproche du débit fixe, pour venir même être équivalentes

pour $R < 2$ bit/dimension. L'écart s'explique par la forme de la cellule regroupant les régions de Voronoï des différents points pouvant être atteints à la suite de la troncature. Comme il est illustré à la Figure 5.19, la cellule diverge de la forme d'une région de Voronoï, et ne correspond pas à une partition selon le principe du plus proche voisin. Le gain granulaire résultant est alors plus faible que celui du réseau gRE_8 . On observe aussi une oscillation des performances du débit variable qui est encore plus visible à la Figure 5.26 (c). Cette oscillation est la conséquence de la troncature à l'aide du débit niveau des eaux r_λ . Les maxima de performance sont atteints lorsque r_λ est entier, c'est à dire qu'un même nombre de bits est retiré à toutes les descriptions. La distorsion ajoutée est alors équivalente sur l'ensemble des vecteurs, on respecte alors la condition du remplissage inverse des eaux. Lorsque r_λ est fractionnaire, la condition n'est plus respectée et les performances baissent. Pour finir, on peut observer que pour le débit fixe et pour $R > 1$ bit /dimension, la distorsion moyenne du débit fixe correspond à la distorsion granulaire D_g du réseau gRE_8 , qui peut s'exprimer en fonction du moment normalisé d'ordre 2 $c(N)$ du réseau :

$$D_g(g\Lambda) = \frac{1}{vol(g\Lambda)} \int_{vol(g\Lambda)} \|x\|^2 dx = Ng^2 vol(\Lambda)^{2/N} c(N) \quad (5.50)$$

où $vol(\Lambda)$ est le volume de la région de Voronoï $V(\Lambda)$ de Λ . Pour RE_8 , $V(\Lambda) = 256$ et $c(N) \approx 0.0717$ (cf. Tableau 5.1), on a alors la relation $D_g(gRE_8) \approx 2.2944g^2$. Par contre lorsque $R < 1$ bits/dimension, le gain g tendant vers l'infini, la distorsion converge alors vers la variance du signal x , c.-à-d. 1, et s'éloigne de D_g .

Performance avec les dictionnaires de base l'AMR-WB+

On utilise cette fois-ci les dictionnaires de l'AMR-WB+ donnés au Tableau 5.2. L'incrément en débit est de 1/2 bits/dimension ce qui rend la quantification plus adaptable à la source et augmente ainsi les performances comme le montre les Figures 5.27 de (a) à (d). L'écart entre les performances du débit fixe et la limite de Shannon est maintenant aux alentours de 2.14 dB. La Figure 5.27 (a) démontre principalement que le codage unaire des numéros de dictionnaire utilisé dans la recommandation AMR-WB+ est sous-optimal principalement à haut débit du fait que les codes ne sont plus adaptés à la distribution des numéros. En comparant les Figures 5.27 (b) et (c), on observe encore l'apport très significatif de l'ajustement fait au décodage. L'oscillation est toujours présente pour les mêmes raisons évoquées précédemment. Par contre, on observe qu'autour de 2 bit/dimension, la distorsion moyenne s'écarte anormalement de la performance du débit fixe. Ceci est dû aux dictionnaires de base de taille supérieure à 1 bit/dimension qui ne

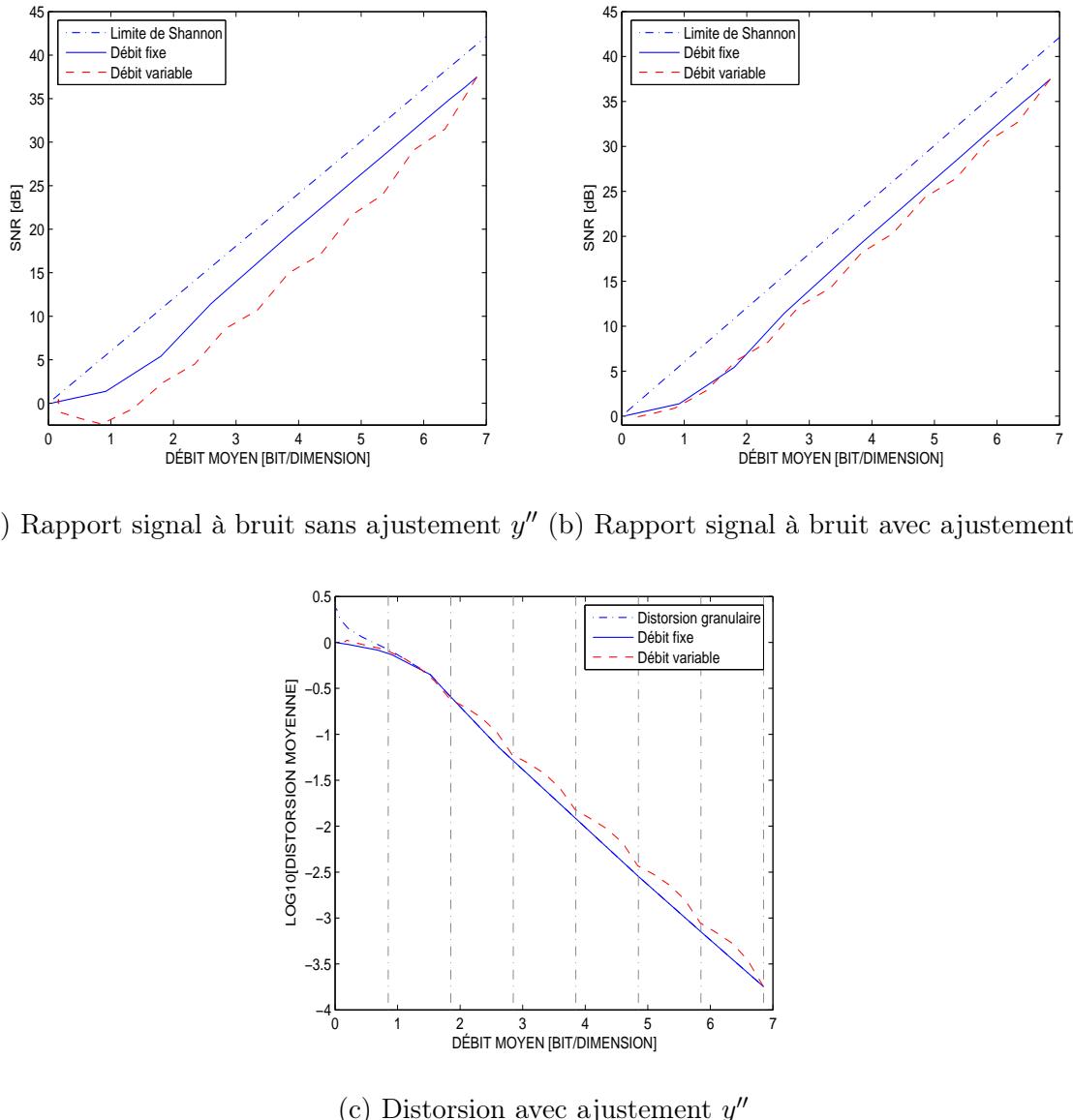


Figure 5.26 Performances de l'extension de Voronoï graduelle pour le cas gaussien avec un dictionnaire de base nul.

peuvent pas contrairement aux codes de Voronoï $V(\Lambda, 2\Lambda)$ se décomposer en série de descriptions de 1 bit/dimension.

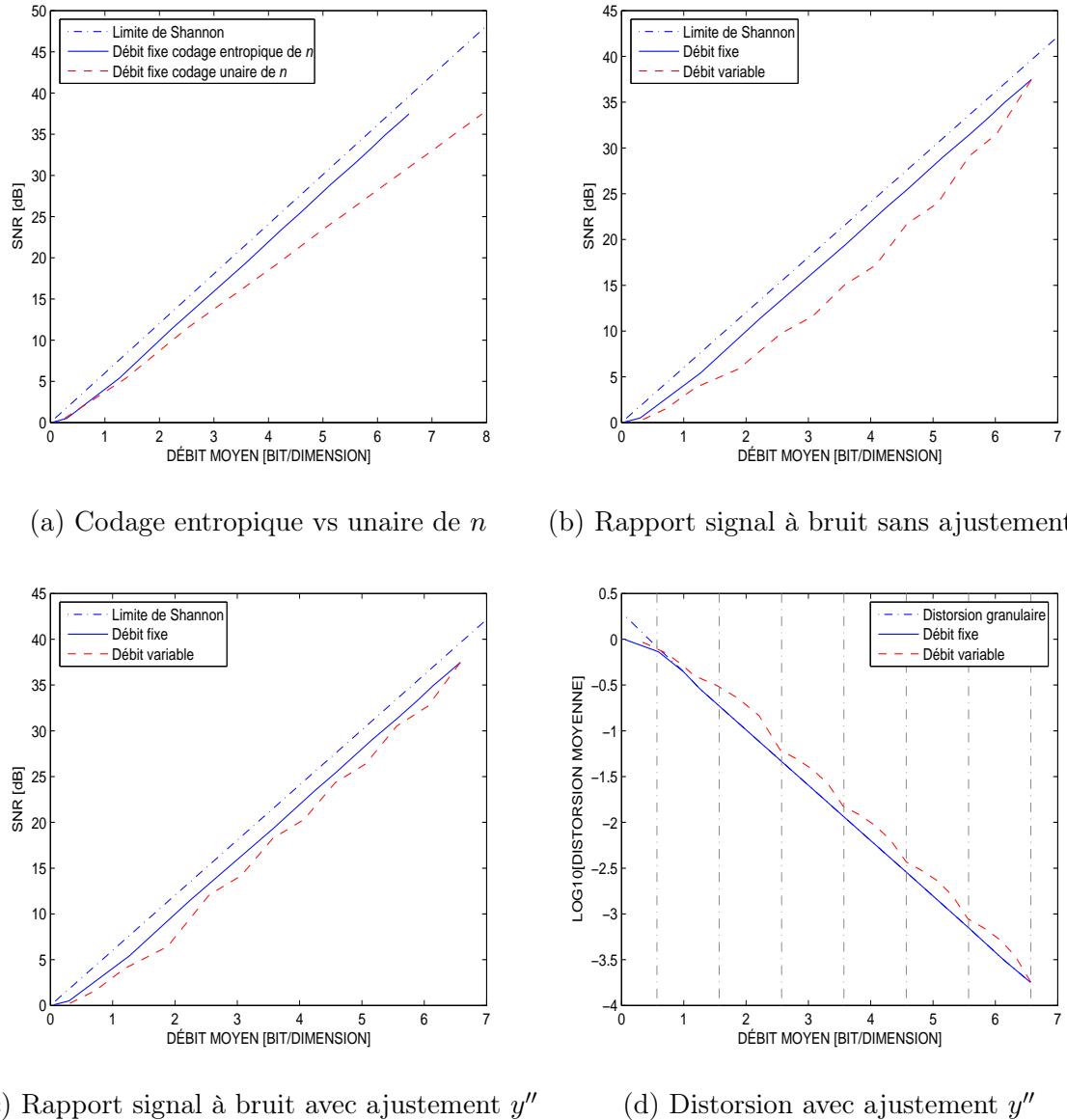


Figure 5.27 Performances de l'extension de Voronoï graduelle pour le cas gaussien avec les dictionnaires de base de l'AMR-WB+.

5.5.2 Codage d'une cible TCX

Dans cette section nous comparons les performances du quantificateur de l'AMR-WB+ employant l'extension de Voronoï originale [133] avec le quantificateur à raffinements successifs utilisant

l'extension de Voronoï graduelle. On compare les deux quantifications dans le cas réel du codage d'une cible TCX d'une séquence audio de 2 minutes très hétérogène mélangeant de la musique et de la parole. Dans les deux cas, on utilise les mêmes dictionnaires de base. Le gain global g est cette fois-ci codé sur une échelle logarithmique sur 7 bits. Le débit de codage n'est plus fluctuant, car g s'adapte automatiquement pour chaque trame. On mesure donc le débit instantané et non plus le débit moyen.

Pour le train binaire issu de la quantification de l'AMR-WB+, nous ordonnons les numéros de dictionnaire n , codés par un codage unaire, et les index k de telle façon qu'on puisse avoir un décodage graduel du spectre, vecteur par vecteur, comme l'illustre la Figure 5.8 (a). Pour la quantification à raffinements successifs utilisant l'extension de Voronoï graduelle, on génère le train binaire comme il est expliqué à la Figure 5.24. On obtient alors en plus du décodage graduel du spectre, un raffinement graduel des amplitudes des vecteurs codés comme l'illustre la Figure 5.8 (b).

La Figure 5.28 (a) donne les résultats de la comparaison d'un codage à débit fixe avec un décodage à débit variable à partir d'un train binaire à 20 kbit/s. La première constatation est que notre source n'est pas une gaussienne sans mémoire. Elle s'en rapproche, mais d'une part elle est plutôt laplacienne et d'autre part les coefficients ne sont pas entièrement décorrélés même après transformation. La quantification vectorielle exploite bien cette dernière caractéristique d'où le dépassement de la limite de Shannon pour le cas du codage à débit fixe. Dans le cas du décodage à débit variable, les deux solutions de décodage graduel sont comparables en terme de performance. Cela s'explique par l'utilisation d'ordres peu élevés par les deux différentes extension de Voronoï.

Par contre, pour des débits et des ordres d'extensions plus élevés, l'avantage de l'extension de Voronoï graduelle devient plus évidente. La Figure 5.28 (b) illustre cet avantage lorsque que le décodage à débit variable provient d'un train binaire à 100 kbit/s. On distingue alors les différentes caractéristiques des deux quantificateurs. La première constatation est que le décodage graduel par vecteur obtenu avec l'extension de Voronoï originale est sous-optimal selon le critère $D(R)$. Comme il a été dit à la section 5.4.4, c'est le remplissage inverse des eaux qui est optimal au sens débit-distorsion. La quantification à raffinements successifs utilisant l'extension de Voronoï graduelle permet de se rapprocher par la décomposition de vecteurs en plusieurs code-vecteurs issus de codes de Voronoï. La Figure 5.28 met aussi en évidence le fait que le codage unaire des numéros de dictionnaires est sous-optimal pour des débits élevés. La description des numéros de

dictionary par des vecteurs de position montre alors son potentiel, et la quantification par extension de Voronoï graduelle dépasse les performances de la quantification de la recommandation AMR-WB+. Finalement à très haut débit lorsque le gain global g approche de l'unité, l'erreur de codage sur le gain n'est plus négligeable par rapport à la distorsion granulaire du réseau $\hat{g}RE_8$. Les performances tendent alors vers une limite asymptotique. Il faudrait donc allouer plus que 7 bits au codage du gain car l'allocation devient très sous-optimale à haut débit.

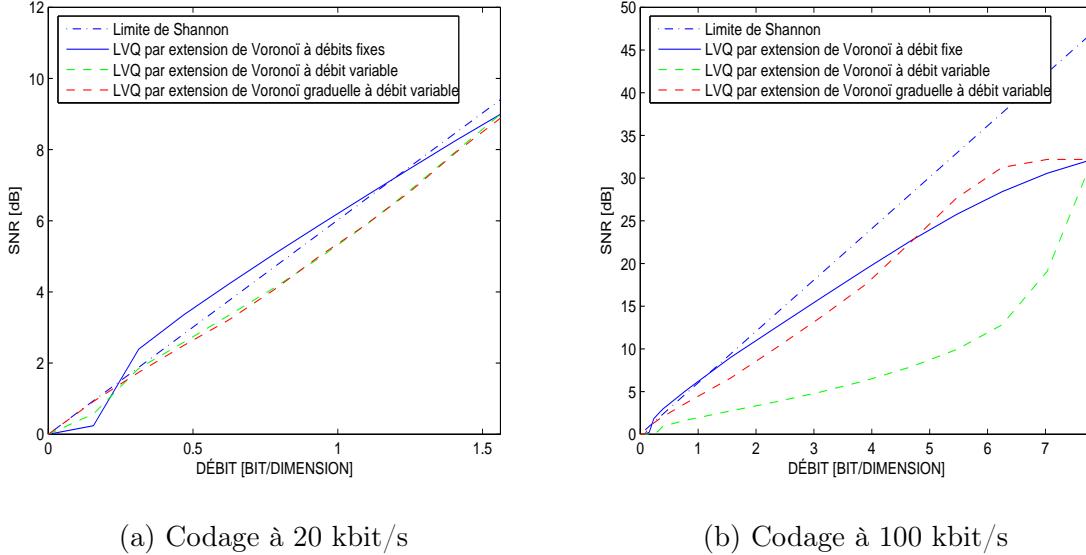


Figure 5.28 Performances de différentes stratégies de quantifications à raffinements successifs de la cible TCX dans RE_8 comparées à une quantification à débit fixe.

5.6 Conclusion

Nous venons de présenter une nouvelle technique de quantification vectorielle algébrique appelée quantification par extension de Voronoï graduelle. Elle repose sur les codes de Voronoï comme l'extension de Voronoï originale proposée par [133], mais en permettant un décodage incrémental de fine granularité des vecteurs codés. Elle s'adapte particulièrement bien à la conception d'une quantification à raffinements successifs. Nous avons entre autres présenté une quantification multidébit et à raffinements successifs se basant sur les dictionnaires de base du standard AMR-WB+.

Des tests ont montré que les performances de cette dernière quantification est proche de la limite de Shannon pour le cas d'une source gaussienne sans mémoire lorsque qu'on utilise le même débit

pour le codage et le décodage. En outre, même lors d'un décodage à débit variable à l'aide d'une troncature d'un même train binaire, la qualité de restitution reste très proche à débit équivalent de celle du codage optimisé pour un bit fixe. Ceci démontre que les raffinements successifs suivent bien une fonction débit-distorsion quasi optimale.

De plus la quantification proposée offre des avantages non-négligeables, comme le faible coût de stockage, des débits virtuellement illimités et l'adaptation à un très grand nombre de sources par l'optimisation des dictionnaires de base. Néanmoins, l'avantage le plus attrayant reste le décodage à débit variable qui permet de décomposer le train binaire avec une très fine granularité. Un incrément aussi faible que 1 bit/dimension permet d'améliorer la description d'un signal.

Nous allons dans la suite de la thèse nous servir de cette quantification à raffinements successifs pour concevoir nos solutions de codage audio hiérarchique.

CHAPITRE 6

Codage Hiérarchique en Bande Élargie

6.1 Introduction

On propose dans ce chapitre un codeur hiérarchique se basant sur le principe du codage à raffinements successifs. Un tel type de codage est formé de plusieurs couches, où l'erreur de codage d'une couche est prise en compte par la couche suivante. Notre structure est formée de deux couches et fait suivre un codeur de parole de type CELP par un codage par transformée. Cette approche est assez populaire pour obtenir un codage mixte parole-musique [68, 69, 70], comme nous l'avons vu au chapitre 3. Elle est plus robuste et polyvalente que le posttraitement fréquentiel proposé au chapitre 4, car elle permet de transmettre explicitement les erreurs du codeur de parole. Mais C'est aussi une approche plus coûteuse en termes de débit et de complexité algorithmique.

Dans la première section du chapitre, nous présentons la structure hiérarchique. Elle met en jeu le codeur de parole AMR-WB à 12.65 kbit/s avec un codage par transformée utilisant la quantification algébrique à raffinements successifs introduite dans le chapitre 5. On obtient un débit de codage total de 24 kbit/s, et un débit de décodage variable de 12.65 à 24 kbit/s. Les performances du codeur hiérarchique proposé sont présentées au travers un test subjectif. L'analyse des résultats permet de constater que le codeur hiérarchique a une qualité de reproduction très homogène quelque soit la nature du signal à coder, mais qu'il reste encore quelques insuffisances pour certains types de signaux.

Dans la seconde section, des optimisations perceptuelles sont apportées au codeur hiérarchique afin de répondre aux problèmes entrevus dans la première section. On intègre ainsi à la structure deux outils supplémentaires. Le premier est le posttraitement fréquentiel du chapitre 4 pour améliorer la qualité de la synthèse des signaux musicaux. Le second est une commutation du domaine de codage permettant d'annuler le préécho du codage par transformée lors de fortes transitoires. Un test subjectif confirme les améliorations apportées.

6.2 Structure hiérarchique du codeur

6.2.1 Principe

Le schéma de principe 6.1 donne une vision globale de la structure hiérarchique du codeur. Le codeur de parole est suivi d'une transformation T et d'une quantification Q . Le codeur de parole, dans notre cas l'AMR-WB, produit une première synthèse \hat{x}_{base} en transmettant un sous-flux de base à 12.65 kbit/s. Le codage par transformée prend en compte l'erreur du codeur de parole et la quantifie pour former le sous-flux d'amélioration à 11.35 kbit/. Le débit total est alors de 24 kbit/s. Le sous-flux d'amélioration peut être décodé à différents débits, car il provient d'une quantification à raffinements successifs. La synthèse \hat{x}_{enh} est alors une version améliorée de la synthèse \hat{x}_{base} .

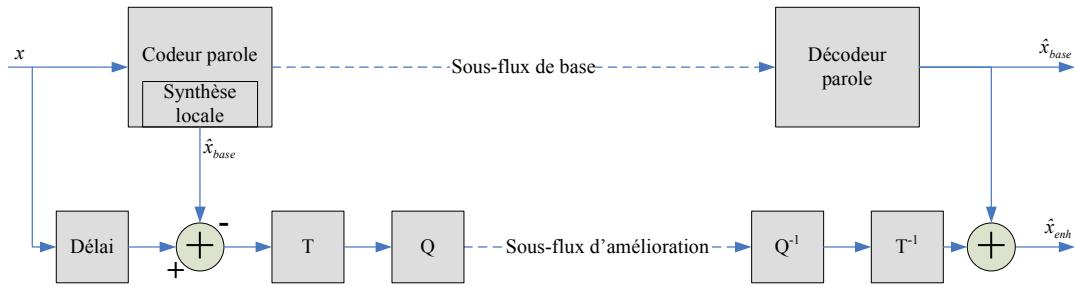


Figure 6.1 Schéma de principe du codage hiérarchique.

Le codeur et le décodeur sont détaillés aux Figures 6.2 et 6.3 respectivement. Le codage par transformée s'effectue dans le domaine de la MDCT par la LVQ à raffinements successifs. Le codeur de parole, l'AMR-WB, et le codage par transformée sont interfacés par un modèle perceptuel qui permet aux deux couches de tirer profit au maximum l'une de l'autre. Le modèle perceptuel consiste en une procédure de masquage et de mise en forme par le filtre $W(z)$ du signal de différence provenant du codeur de parole. Les blocs fonctionnels MDCT et ODFT intègrent le fenêtrage, la gestion de la mémoire ainsi que le recouvrement à 50%. Les différentes parties de codeur hiérarchique sont détaillées ci-dessous.

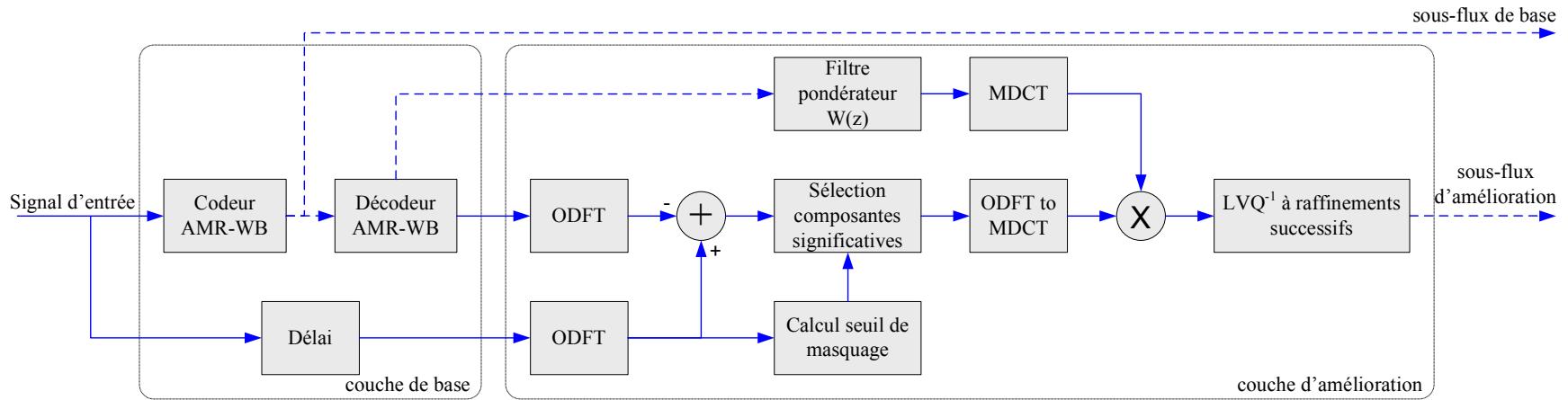


Figure 6.2 Schéma bloc du codeur hiérarchique.

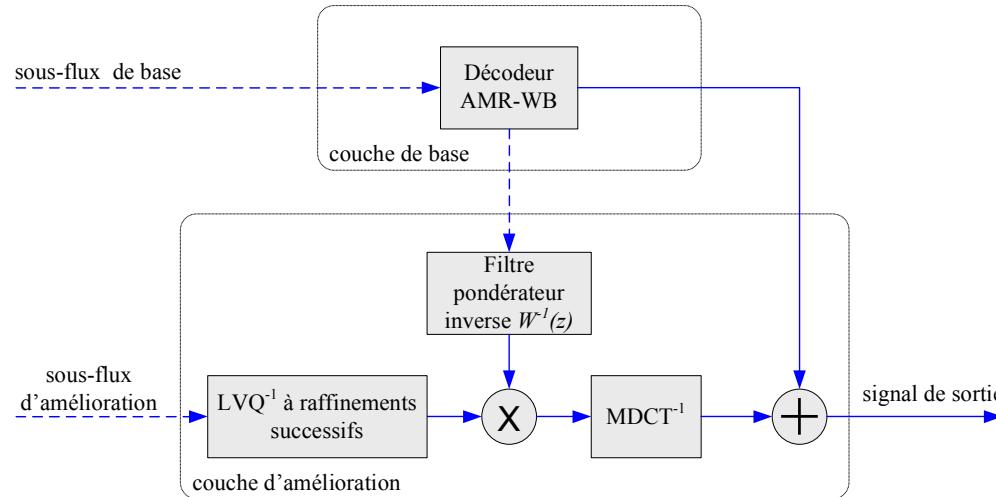


Figure 6.3 Schéma bloc du décodeur hiérarchique.

6.2.2 Couche de base

Nous utilisons dans ce chapitre comme codeur de base le codeur de parole large bande AMR-WB [7], traitant le signal d'entrée dans le domaine temporel. Le signal est tout d'abord décomposé en deux bandes de fréquence 50-6.4 kHz et 6.4-7 kHz. La bande haute subit un codage hautement paramétrique alors que la bande basse fait l'objet d'un codage ACELP. Nous nous intéressons à améliorer principalement la bande basse dans la couche supérieure. La restitution de la bande haute reste alors une simple estimation. Ainsi dans tout ce qui suit, le signal traité est échantillonné à 12.8 kHz. Le mode à 12.65 kbit/s de l'AMR-WB est privilégié.

6.2.3 Couche d'amélioration

La couche d'amélioration traite le signal de différence entre le signal original et la synthèse du codeur de parole. Le signal de différence est codé dans le domaine fréquentiel par une MDCT. Les coefficients de la MDCT sont calculés par l'intermédiaire d'une transformée discrète de Fourier impaire (*Odd Distrete Fourier Transform*, ODFT) [139] sur deux trames successives de 20 ms de l'AMR-WB. L'ensemble de deux trames, formant alors une trame de 40 ms de $N = 512$ échantillons. On rappelle que les coefficients réels de la MDCT sont donnés par l'expression suivante :

$$X_M(k) = \sum_{i=0}^{N-1} x(n)h(n) \cos\left(\frac{2\pi}{N}(k + \frac{1}{2})(n + n_0)\right) \quad (6.1)$$

où $n_0 = \frac{1}{2} + \frac{N}{4}$. Les coefficients de l'ODFT sont quant à eux donnés par :

$$X_O(k) = \sum_{i=0}^{N-1} x(n)h(n) \exp\left(j\frac{2\pi}{N}(k + \frac{1}{2})n\right) \quad (6.2)$$

Il est alors facile de montrer que les coefficients de la MDCT peuvent être déduits à partir de ceux de l'ODFT par l'expression suivante :

$$X_M(k) = |X_O(k)| \cos\left(\frac{2\pi}{N}(k + \frac{1}{2})n_0 - \angle X_O(k)\right) \quad (6.3)$$

On utilise comme filtre prototype $h(n)$ la fenêtre KBD (*Kaiser Bessel Derived*) comme filtre prototype. Elle vérifie les conditions de reconstruction parfaite de l'équation 2.19 et est définie

par l'équation suivante :

$$h(n) = \begin{cases} \sqrt{\frac{\sum_{i=0}^n w(i)}{\sum_{i=0}^{N/2} w(i)}} & \text{si } 0 \leq n < N/2 \\ \sqrt{\frac{\sum_{i=0}^{N-1-n} w(i)}{\sum_{i=0}^{N/2} w(i)}} & \text{si } N/2 \leq n < N \\ 0 & \text{sinon} \end{cases} \quad (6.4)$$

avec

$$w(n) = \begin{cases} \frac{I_0(\pi\alpha\sqrt{1-(2n/(N/2+1)-1)^2})}{I_0(\pi\alpha)} & \text{si } 0 \leq n \leq N/2 \\ 0 & \text{sinon} \end{cases} \quad (6.5)$$

$w(n)$ est la fenêtre de Kaiser alors que I_0 est la fonction de Bessel de premier type [140]. Le paramètre α influence la forme de la fenêtre et donc directement l'importance du recouvrement. Dans notre implémentation nous avons choisi $\alpha = 4$ pour un taux de recouvrement de 50%. La Figure 6.4 illustre la reconstruction parfaite grâce au recouvrement de 50% de deux trames successives fenêtrées après la transformation MDCT direct et inverse par $h^2(n)$. On obtient alors après la transformation inverse et une procédure de recouvrement et d'addition (*overlap-add*), une trame de sortie de 20 ms de $N/2 = 256$ échantillons avec un délai généré de 20 ms.

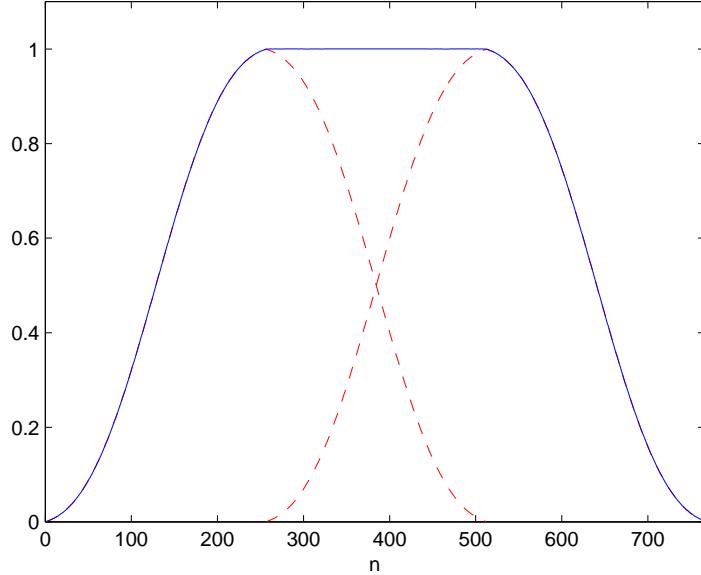


Figure 6.4 Illustration de la reconstruction parfaite à l'aide de deux fenêtres KBD au carré $h^2(n)$ successives avec $\alpha = 4$.

Le principal inconvénient de l'utilisation d'une MDCT est qu'une seule transformation isolée n'est pas orthogonale. En effet, elle l'est seulement après l'annulation du repliement temporel (*Time Domain Aliasing Cancellation*, TDAC) réalisée à l'aide du procédé d'*overlap-adding*. Si une altération a lieu dans le domaine transformé, c.-à-d. une erreur de codage dans notre cas, alors l'annulation du repliement temporel n'est plus parfaite et des artefacts temporels viendront se rajouter à la synthèse. Dans notre cas si la couche d'amélioration n'a pas de débit alloué alors le signal de sortie sera égal à la synthèse du codeur parole. Si le débit est suffisant alors la sortie sera égale ou presque égale au signal original. Pour les débits intermédiaires la qualité de la restitution devrait se trouver entre les deux qualités évoquées ci-dessus, c.-à-d. celle de la synthèse du codeur parole et la qualité transparente. Néanmoins, il est possible d'introduire des artefacts dus entre autres au repliement temporel non annulé. Dans notre cas, ce genre d'artefacts est très peu audible du fait que la forme d'onde de la synthèse du codeur parole est quand même proche de celle du signal original. De plus, le fenêtrage a tendance à atténuer ce genre de problème.

6.2.4 Modèle perceptuel

On utilise deux types de méthodes pour prendre en compte les caractéristiques de l'appareil d'auditif humain.

Le masquage

Les codeurs perceptuels essayent d'obtenir une erreur de codage en dessous du seuil d'audition et de masquage de l'appareil auditif. De cette manière, le bruit de codage injecté est inaudible tout en garantissant un gain de compression élevé. Un tel procédé est généralement réalisé en adoptant une allocation binaire appropriée pour le codage des différentes composantes spectrales en se basant sur un modèle auditif. Dans notre cas, le faible débit alloué à la seconde couche (11.35 kbit/s) ne garantit pas que toute l'erreur de quantification soit inaudible ou masquée. L'objectif est alors d'économiser le maximum de ressources pour les composantes les plus significatives selon la perception auditive.

Les bits alloués à la couche d'amélioration ne doivent donc être assignés qu'aux composantes audibles du signal de différence. Un seuil de masquage est alors calculé à partir du signal original dans le domaine de l'ODFT en se basant sur le modèle introduit par Johnston [38]. Il permet de déterminer si une composante spectrale du signal de différence est signifiante ou non : les composantes signifiantes ont leur énergie au-dessus du seuil et sont donc audibles, alors que

les composantes non signifiantes ont leur énergie en dessous du masque et sont inaudibles. Les composantes non signifiantes ne nécessitent pas de description supplémentaire et sont mises à zéro. Les autres composantes sont codées par la LVQ à raffinements successifs. La Figure 6.5 illustre le procédé de masquage dans le cas d'un segment de parole voisé. Le masquage permet aussi, mais dans une moindre mesure, d'alléger le processus de quantification du signal de différence en diminuant l'énergie totale du signal à quantifier.

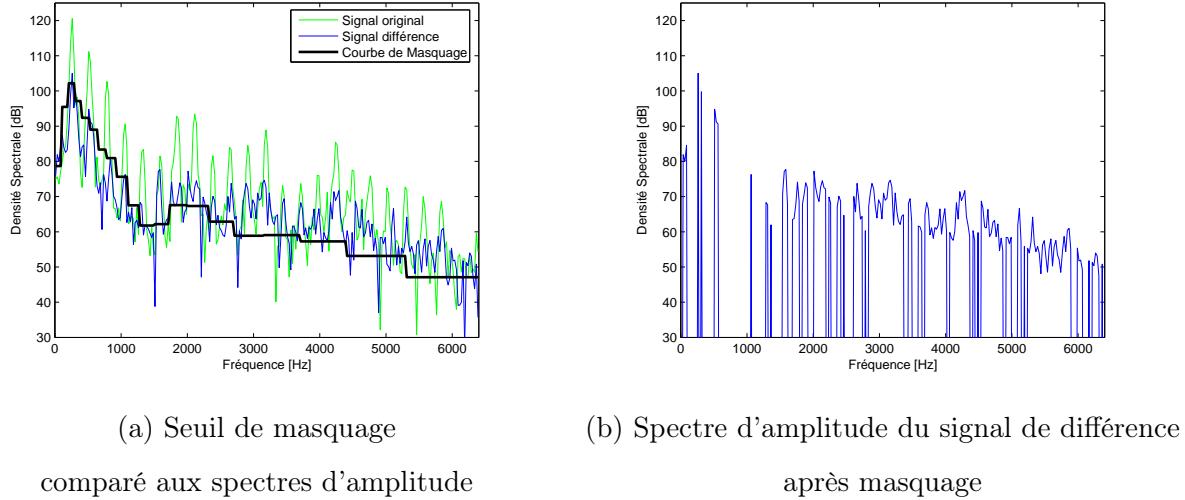


Figure 6.5 Illustration du processus de masquage du signal de différence.

Mise en forme du bruit de codage

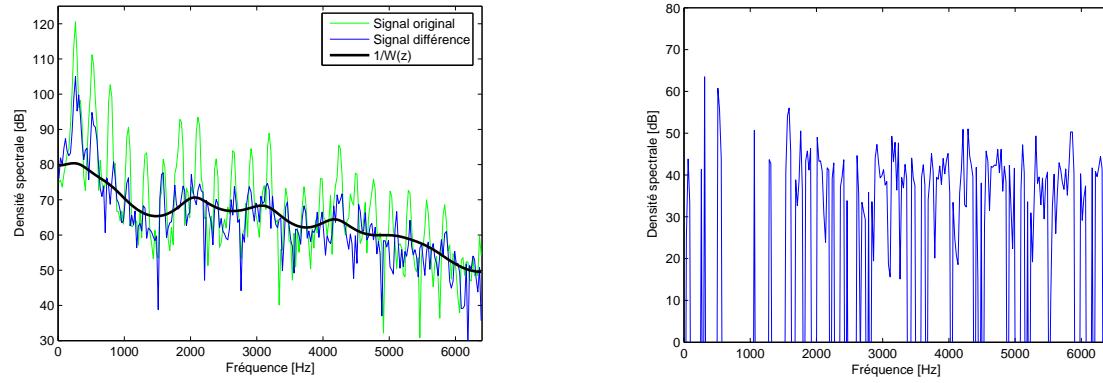
Dans les codeurs de parole par analyse-par-synthèse, le résidu de la prédiction linéaire est usuellement codé de façon à minimiser l'erreur quadratique moyenne entre le signal original et la synthèse dans un domaine perceptuel dit pondéré. La pondération perceptuelle est effectuée à l'aide d'un filtre pondérateur $W(z)$:

$$W(z) = \hat{A}(z/\gamma_1)/\hat{A}(z/\gamma_2) \quad 0 \leq \gamma_2 < \gamma_1 \leq 1 \quad (6.6)$$

où $\hat{A}(z)$ est le filtre issu des coefficients quantifiés de la prédiction linéaire, et γ_1, γ_2 des facteurs contrôlant la pondération spectrale. L'AMR-WB utilise les paramètres $\gamma_1 = 0.92$ et $\gamma_2 = 0$. Le bruit de codage de l'AMR-WB est donc mis en forme par le filtre $W^{-1}(z) = 1/\hat{A}(z/\gamma_1)$ comme l'illustre la Figure 6.6 (a).

L'utilisation du même filtre pondérateur $W(z)$ à la sortie du codeur de parole dans la couche d'amélioration permet de blanchir le signal de différence. Les principales redondances du signal de

différence sont alors retirées. Dans la couche d'amélioration, les paramètres du filtre pondérateur sont extraits de la couche de base et convertis en des coefficients MDCT de pondération. Les coefficients MDCT du signal de différence sont ainsi normalisés par les coefficients pondérateurs obtenus. Ce sont ces coefficients pondérés du spectre blanchi qui sont alors quantifiés. Un exemple de spectre d'amplitude blanchi du signal de différence est donné dans le domaine MDCT à la Figure 6.6 (b).



(a) Bruit de codage à la sortie l'AMR-WB comparé à la réponse fréquentielle de $W^{-1}(z)$ (b) Spectre d'amplitude MDCT blanchi du signal de différence avant quantification

Figure 6.6 Illustration du blanchiment du signal de différence.

De plus, la normalisation du spectre d'amplitude met en forme la nouvelle erreur de codage introduite dans la couche d'amélioration en utilisant le même critère perceptuel que celui de la couche de base. On se ramène ainsi au cas d'un codage de type TCX. Mais contrairement au codage TCX classique, on code ici un signal de différence au lieu du signal original. Par la suite, on appelle les coefficients MDCT sélectionnés et pondérés du signal de différence, la cible TCX.

6.2.5 Codage de la cible TCX

La cible TCX est codée par la quantification algébrique (LVQ) à raffinements successifs introduite dans le chapitre 5. Les coefficients de la cible TCX sont regroupés en vecteur de 8 et projetés sur le réseau régulier de points RE_8 . La quantification est mutidébit et utilise plusieurs dictionnaires Q_n de débit de $n/2$ bits par dimension, Q_0 représentant le vecteur nul. Les dictionnaires Q_n sont optimisés pour une source laplacienne pour $n < 5$. Pour $n > 5$ ils sont générés à partir des dictionnaires Q_3 et Q_4 alternativement par une extension de Voronoi graduelle. Ainsi, les vecteurs

codés par la quantification sont décodables graduellement avec une granularité d'amélioration pouvant être aussi fine que la longueur des codes de Voronoï $V_r(2RE_8)$, c.-à-d. 8 bits ou encore 1 bit par dimension. Il est à noter que la prise en compte d'un bit supplémentaire par trame lors du décodage permet d'améliorer en moyenne la qualité globale de restitution étant donné que la troncature ne tombe pas forcément à la même position d'une trame à l'autre. Q_1 n'est pas utilisé car n'étant pas assez précis. Les vecteurs devant être codés par Q_1 sont mis à zéro et sont codés par Q_0 libérant ainsi des ressources pour les vecteurs plus énergétiques. L'allocation binaire ainsi que la distorsion sont contrôlées implicitement par un gain global g optimisé pour chaque trame et codé sur 7 bits sur une échelle logarithmique. On utilise le même processus que dans la recommandation AMR-WB+ [67] pour optimiser g .

La cible TCX après normalisation par le gain global g et sans procédure de masquage, a une distribution se rapprochant d'une distribution laplacienne. La Figure 6.7 (a) montre que la densité de probabilité de la cible TCX normalisée pour une séquence audio de 23 minutes mélangeant de la parole et de la musique. Il est possible de voir que la densité de probabilité associée à la cible est plus proche d'une distribution laplacienne ($\gamma = 1$) que gaussienne ($\gamma = 2$), car elle peut être modélisée par une gaussienne généralisée de paramètre $\gamma = 1.25$. L'utilisation des dictionnaires ayant des contours optimisés pour une source laplacienne est alors justifiée. Lorsqu'on utilise le processus de masquage, on force un certain nombre de valeurs à zéro. Ces valeurs étant sous le seuil de masquage sont généralement de faibles amplitudes. On remarque grâce à la Figure 6.7 (b) que la distribution change de caractéristique. Le pic de la densité de probabilité est beaucoup plus prononcé en zéro aux dépens des valeurs avoisinantes qui ont une densité plus plate. Les vecteurs ayant des composantes mises à zéro peuvent alors être codés par des dictionnaires plus petits libérant ainsi des ressources pour les composantes significatives. Dans notre cas, on utilise toujours les mêmes dictionnaires optimisés pour une source laplacienne. Il serait possible d'optimiser les dictionnaires pour la nouvelle source afin d'exploiter davantage les nouvelles caractéristiques statistiques.

La Figure 6.8 compare l'histogramme des numéros de dictionnaire utilisés pour la cible TCX avec et sans processus de masquage. Lors de l'utilisation du masquage, on observe que le dictionnaire Q_0 est favorisé par rapport aux dictionnaires Q_2 et Q_3 , ce qui permet une utilisation plus importante des dictionnaires Q_n pour $n > 3$. Néanmoins, le processus de masquage n'a qu'une influence limitée sur cette statistique. Si on regarde la moyenne des numéros de dictionnaire utilisés par

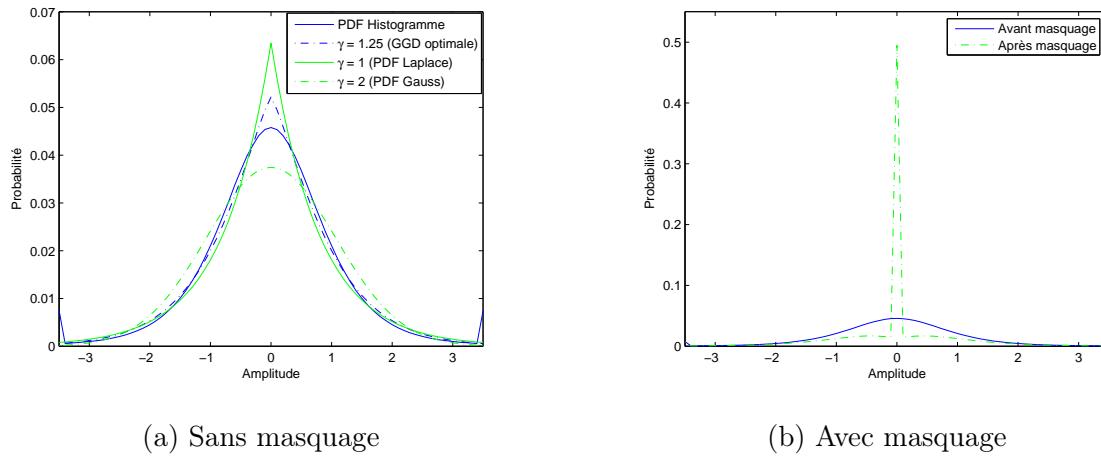


Figure 6.7 Densité de probabilité de la cible TCX normalisée avec et sans masquage (les valeurs à -3.5 et 3.5 représentent les probabilités cumulées pour les valeurs < -3.5 et > 3.5 respectivement).

vecteur à la Figure 6.9, on observe, malgré le blanchiment du spectre, que les composantes de la cible TCX ont une plus forte énergie dans les basses fréquences et se trouvent ainsi privilégiés pour ces fréquences. Le masquage permet de contrebalancer cette tendance en reportant une partie des ressources originellement allouée aux basses fréquences vers les hautes fréquences.

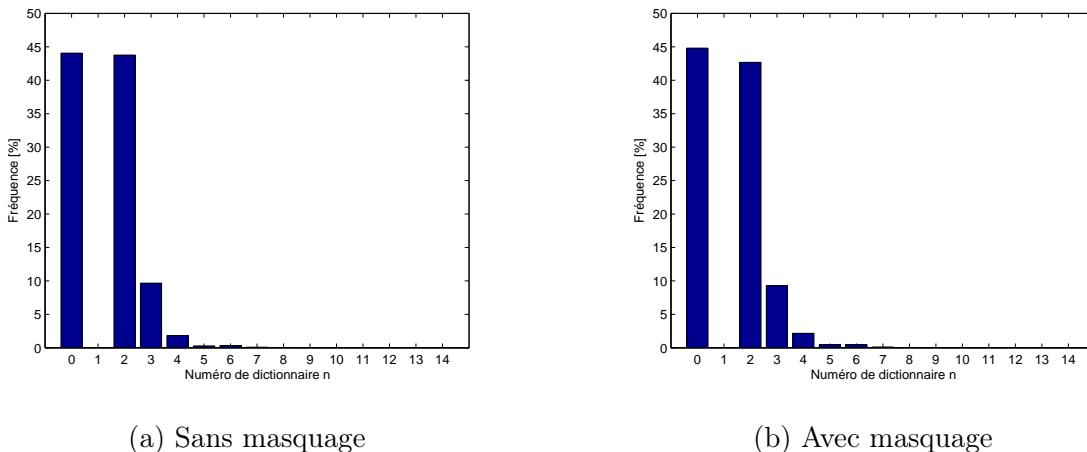


Figure 6.8 Fréquence d'utilisation des dictionnaires de la LVQ.

6.2.6 Performances du codeur hiérarchique

Mesures objectives

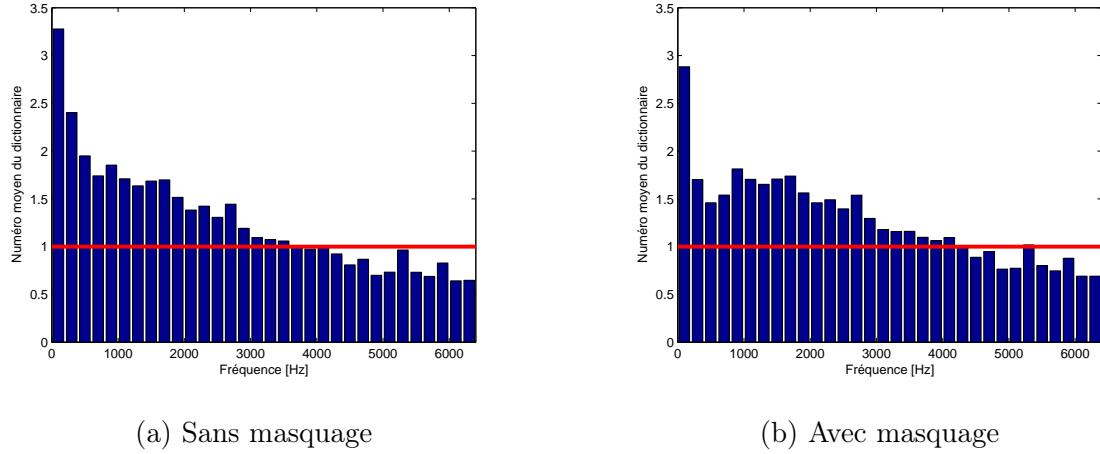


Figure 6.9 Numéro moyen de dictionnaire utilisé par vecteur de dimension 8 selon les bandes de fréquence.

Nous avons utilisé le rapport signal à bruit segmental (segSNR) pour mesurer les performances de notre codeur hiérarchique. La mesure a été faite tout les 20 ms dans le domaine pondéré de la cible TCX après le filtrage $W(z)$. On a utilisé une séquence de 2 minutes très hétérogène mixant de la parole, du bruit de fond et de la musique. Nous avons comparé les performances de notre solution à 24 kbit/s, et pour différents débits de décodage allant de 12.65 à 24 kbit/s, avec les différents modes de l'AMR-WB au dessous de 12.65 kbit/s. Les résultats sont reportés à la Figure 6.10.

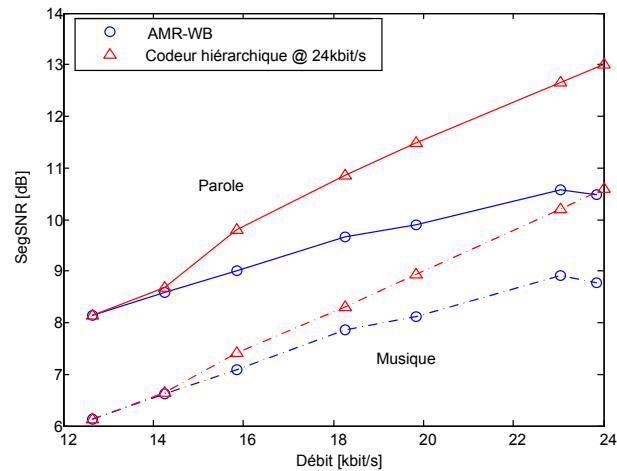


Figure 6.10 segSNR du codeur hiérarchique comparé à l'AMR-WB.

Pour ce critère, le codeur hiérarchique est meilleur que l'AMR-WB pour tous les débits. De plus, le segSNR du codeur hiérarchique est une fonction linéaire du débit de décodage ce qui démontre que les raffinements successifs de la synthèse sont bien graduels. Le domaine pondéré utilisé ici, est un modèle psychoacoustique très simpliste, et ne reflète pas forcément la qualité auditive absolue de la synthèse. C'est pour cette raison que nous avons conduit en sus un test subjectif. Néanmoins, pour les deux codeurs, l'AMR-WB et le codeur hiérarchique, l'erreur de codage est minimisée dans ce domaine pondéré. On peut en déduire que la quantification du codeur hiérarchique, la LVQ à raffinements successifs dans le domaine transformé, est en moyenne plus performante que la quantification ACELP.

Mesures subjectives

Nous avons conduit un test d'écoute à l'aide de la méthode MUSHRA [103] sur 11 auditeurs entraînés. Le test comprenait 4 séquences de parole de voix de femme et d'homme en anglais, 4 séquences de divers styles musicaux et 4 séquences de parole mixée à de la musique ou du bruit de fond. Notre codeur hiérarchique à 24 kbit/s a été comparé à 3 autres codecs, l'AMR-WB à 12.65 et à 23.05 kbit/s et le G.722.1 à 24 kbit/s. L'AMR-WB à 23.05 kbit/s et le G.722.1 à 24 kbit/s servent de références pour le traitement des signaux de parole et de musique respectivement. La référence cachée a été aussi insérée dans le test afin de valider les résultats des auditeurs. Les séquences ont été filtrées passe-haut à 7kHz, pour que les auditeurs se concentrent sur la qualité de la synthèse plutôt que sur la plénitude fréquentielle des sons.

Les résultats sont reportés à la Figure 6.11 qui distingue les 3 différentes catégories de sons. Il apparaît que le codeur hiérarchique a la qualité la plus homogène sur l'ensemble des séquences. On peut alors affirmer qu'il est le plus polyvalent. Par contre, pour chaque catégorie, il se positionne moins bien que les codeurs optimisés, que ce soit pour la parole (AMR-WB) ou pour la musique (G.722.1). La forte contrainte de la structure hiérarchique est pénalisante par rapport aux codeurs non hiérarchiques pouvant faire une optimisation globale pour un débit de décodage fixe. En outre, la polyvalence de notre codeur défavorise ses performances pour des signaux caractéristiques contrairement aux codeurs de parole comme l'AMR-WB faisant une forte hypothèse sur la nature du signal d'entrée. Malgré les deux compromis intrinsèques à notre codeur hiérarchique, ces performances sont surtout pénalisées par deux artefacts clairement audibles. Le premier est dû à l'étalement spectral surtout perceptible lors d'attaques pour les signaux de parole. Le second vient de l'incapacité à défaire la signature très caractéristique du codeur parole lors du traitement des

signaux audio autres que de la parole. Dans la section suivante du chapitre, nous allons apporter deux optimisations au codeur hiérarchique afin d'atténuer ces deux problèmes.

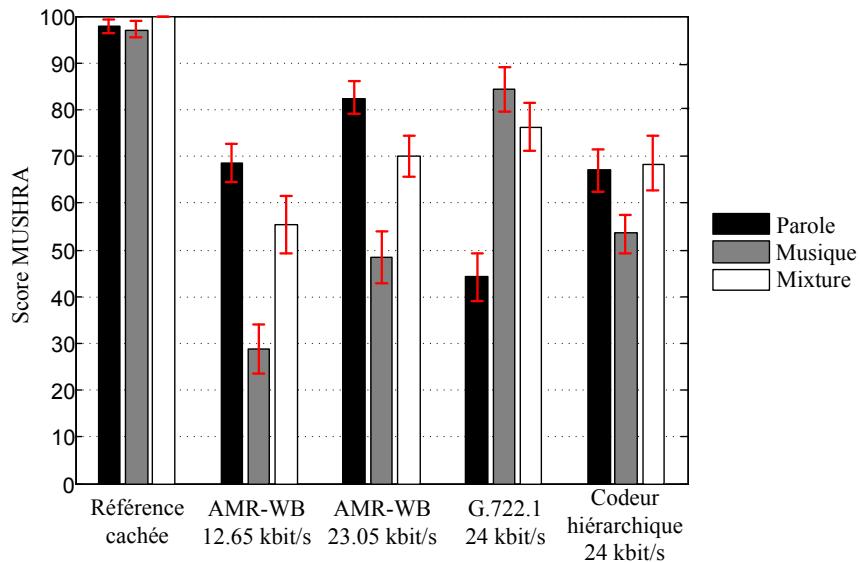


Figure 6.11 Résultats du test MUSHRA évaluant notre codeur hiérarchique.

6.3 Optimisations perceptuelles

6.3.1 Principe

À partir du schéma de codage hiérarchique préalablement décrit, on se propose de rajouter deux outils. Le premier est le posttraitement fréquentiel introduit au chapitre 4. Il va essayer d'améliorer les performances du codeur hiérarchique spécialement pour des sons n'étant ni du bruit ni monoharmoniques et ne répondant donc pas au modèle source-filtre du codeur de parole. Le deuxième outil est un commutateur adaptatif du domaine de codage de la couche d'amélioration. Il permet lors de fortes transitoires de changer temporairement de domaine de codage en passant dans le domaine temporel afin d'éviter l'effet intempestif de l'étalement spectral.

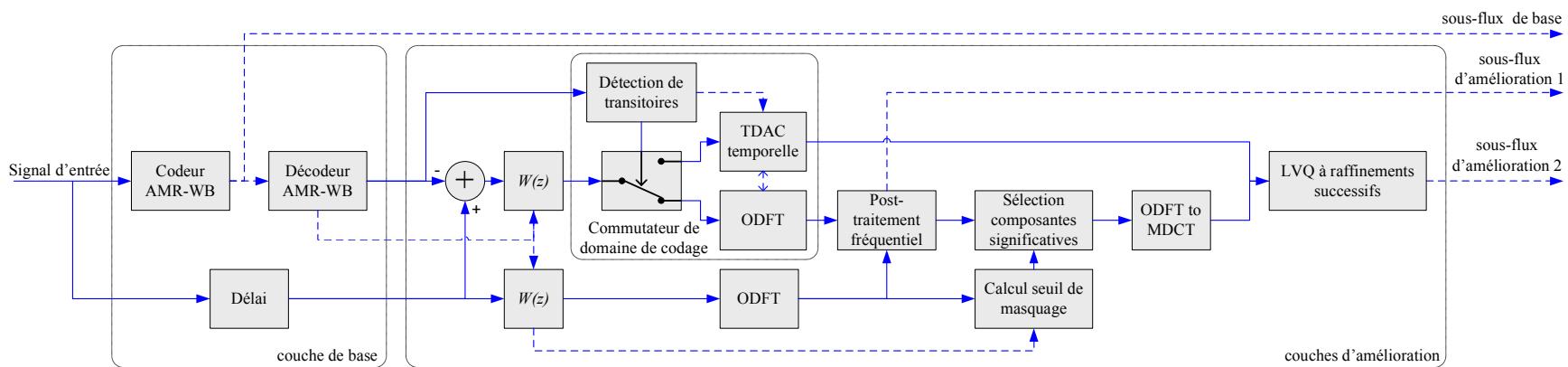


Figure 6.12 Schéma bloc du codeur hiérarchique optimisé

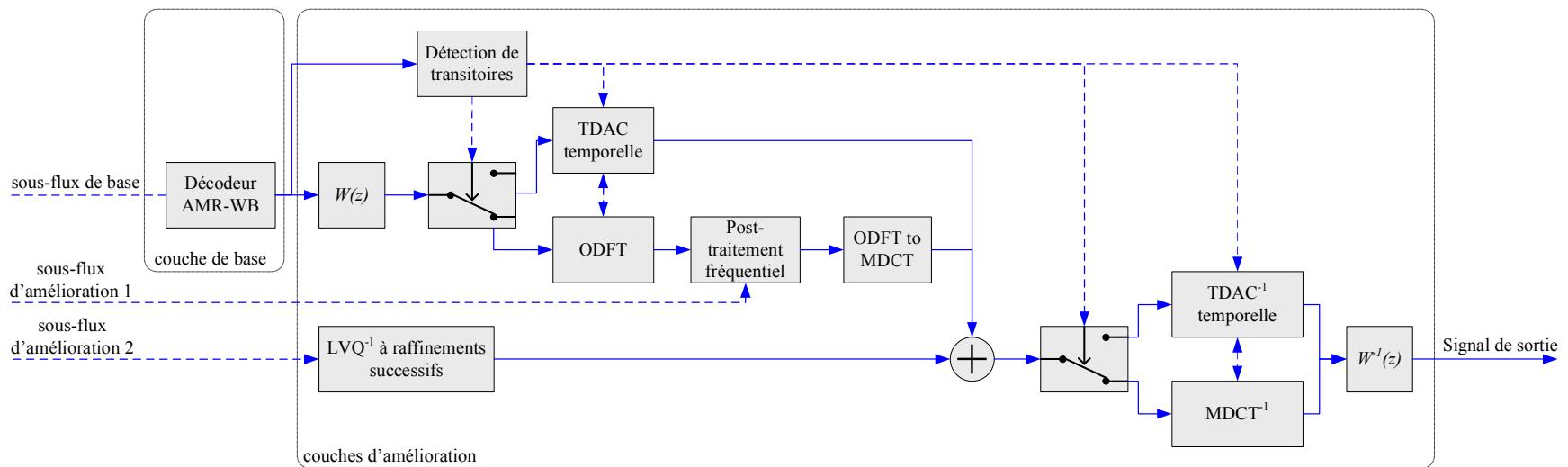


Figure 6.13 Schéma bloc du décodeur hiérarchique optimisé

Les Figures 6.12 et 6.13 représentent respectivement le codeur et le décodeur hiérarchiques optimisés. On peut apercevoir le commutateur du domaine de codage entre le domaine temporel (TDAC temporelle) et le domaine fréquentiel (ODFT). La TDAC (*Time Domain Alias Cancellation*) temporelle, explicitée en détail dans le paragraphe suivant, est la duale de la MDCT dans le domaine temporel. Le domaine est choisi en fonction de la détection ou non d'une transitoire. Si une transitoire est détectée, le signal temporel, après la TDAC temporelle, est directement quantifié. Sinon, le signal est transformé dans le domaine de l'ODFT où le posttraitement fréquentiel est appliqué. Un premier sous-flux d'amélioration est généré. Dans un second temps et dans le domaine de la MDCT, le signal de différence entre le signal original et la synthèse posttraitée est quantifié. Le reste du débit des 11.35 kbit/s qui n'a pas été utilisé par le posttraitement est alloué à la quantification. Un second sous-flux d'amélioration est généré. Comme précédemment les blocs fonctionnels MDCT, ODFT et TDAC temporelle intègrent le fenêtrage, la gestion de la mémoire ainsi que le recouvrement à 50%.

6.3.2 Commutation du domaine de codage

L'étalement spectral est un problème récurrent du codage par transformée qui plus est à bas débit. Il est perceptible surtout lors de transitoires, et plus spécifiquement lors d'attaques. Plusieurs outils existent pour le réduire comme le fenêtrage adaptatif [55] ou bien la mise en forme temporelle du bruit de codage (*Temporal Noise Shaping*, TNS) [58]. Le fenêtrage adaptatif permet de changer de résolution temporelle et fréquentielle durant le processus de codage. La résolution temporelle peut être ainsi augmentée aux dépens de la résolution fréquentielle lors de la détection d'une transitoire. Néanmoins, elle nécessite l'utilisation de deux fenêtres de transition (de début et de fin) entre les deux résolutions pour assurer une reconstruction parfaite comme le montre la Figure 6.14. Le changement de fenêtrage engendre alors un délai supplémentaire égal au moins à la moitié de la durée d'une trame d'entrée, ce qui engendre dans notre cas un délai supplémentaire d'au moins 10 ms.

Le TNS consiste en une prédiction linéaire des coefficients spectraux intratrame. Le TNS est le dual dans le domaine fréquentiel du codage par prédiction linéaire (LPC). Il permet ainsi de coder l'enveloppe temporelle de l'enveloppe d'Hilbert du signal au carré à l'aide d'un filtre de synthèse tout pôle. C'est le résidu de la prédiction fréquentielle qui est alors quantifié. L'erreur de codage est ainsi mise en forme par le filtre de synthèse issu de la prédiction. Pour une LPC

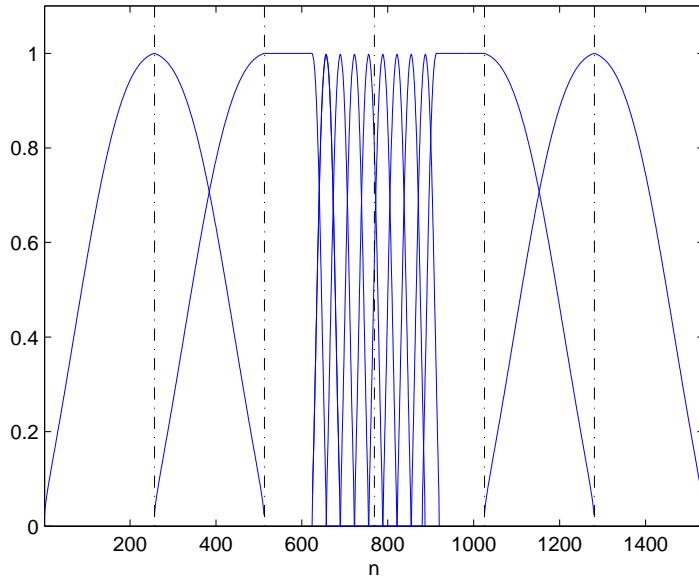


Figure 6.14 Fenêtrage adaptatif entre une résolution temporelle normale et une résolution huit fois plus grande.

l'ordre dépend de la structure en formants de la parole. On a donc en général un ordre de 10 pour la parole bande étroite (5 formant au maximum) et 16 pour la bande élargie (8 formants au maximum). Dans le domaine fréquentiel, l'ordre va correspondre aux variations temporelles, qu'on veut décrire. Si on veut modéliser les variations grossières, l'ordre devra être petit aux alentours de 0.1 pôle/ms. Si au contraire on souhaite modéliser les impulsions glottales, l'ordre devra être plus important. Par exemple, pour modéliser une fréquence élevée de pitch à 300Hz, il faut au moins 1 pôle et son conjugué tous les 1/300s, ce qui correspond à 0.66 pôle/ms. La prédiction linéaire dans le domaine fréquentiel a été plus amplement étudiée dans [141] pour des trames plus longues (de l'ordre de la seconde). Il est alors évident que la prédiction n'est pas la même selon la bande de fréquence considérée. De plus, si comme dans notre cas, on traite l'information à l'aide d'une transformée à échantillonnage critique en utilisant des fenêtres à recouvrement, alors le TNS subira lui aussi l'effet du repliement temporel hérité de la MDCT. On code ainsi l'enveloppe temporelle du signal replié par effet miroir à la moitié du segment gauche et du segment droit de la fenêtre d'analyse. Même si cet effet est atténué par la pondération de la fenêtre d'analyse, il reste tout de même gênant et nécessite l'utilisation lors l'activation du TNS d'un autre jeu de fenêtres ayant un recouvrement moins important. Dans ce cas, la

justesse de l'analyse fréquentielle et, par conséquent, le gain de codage seront affectés. C'est ce compromis qui est adopté dans le codeur AAC-LD [58]. L'autre inconvénient de la TNS est la complexité rajoutée lors du calcul des coefficients de la prédition. Cependant, le TNS se prête particulièrement bien au codage hiérarchique. En effet, comme l'enveloppe temporelle de la synthèse de la couche de base est fortement corrélée avec celle du signal original, il est possible de se passer de l'envoi des coefficients de la prédition et de les calculer au codage comme au décodage sur la synthèse du codeur de base. Dans notre cas, comme on traite principalement de la parole, on veut minimiser la possibilité d'introduire des artefacts, même minimes, lors des transitoires très fréquentes et perceptuellement importantes. Or à bas débit, le TNS peut quand même conserver de l'étalement spectral perceptible, voire rajouter certains artefacts.

On introduit alors une nouvelle méthode pour traiter l'étalement spectral : le commutateur du domaine de codage. Il consiste simplement à commuter entre le domaine fréquentiel et le domaine temporel. Le domaine fréquentiel est pertinent lorsqu'il s'agit de signaux ou de segments pseudostationnaires, ce qui représente la grande majorité des sons. Par contre, lorsqu'il y a une forte transitoire, la résolution temporelle est très importante. Le signal est alors généralement plus décorrélé dans le domaine temporel que dans le domaine fréquentiel. On décide d'après ces considérations d'utiliser le domaine temporel comme domaine de codage dans ces cas précis. La commutation entre les deux domaines peut être vue comme un cas particulier du fenêtrage adaptatif, où la taille des fenêtres est réduite à un seul échantillon lors des transitoires. On n'a alors plus aucune résolution fréquentielle mais une résolution temporelle maximale. Ce cas particulier du fenêtrage adaptatif est illustré à la Figure 6.15. La fenêtre en pointillé est codée dans le domaine temporel, alors que les fenêtres sont codées dans le domaine fréquentiel. Il est toujours nécessaire d'utiliser des fenêtres de transition pour passer d'un domaine à l'autre, ce qui génère le même délai supplémentaire d'au moins une demi-trame. De plus, leur segment rectangulaire réduit la qualité de l'analyse fréquentielle donc le gain de codage. Pour ces raisons, on introduit le TDAC temporel, un processus d'antirepliement temporel à l'instar de la MDCT mais directement appliqué dans le domaine temporel. Cela permet de passer directement du domaine fréquentiel au domaine temporel (et vice versa) d'une fenêtre à l'autre sans passer par des fenêtres de transition.

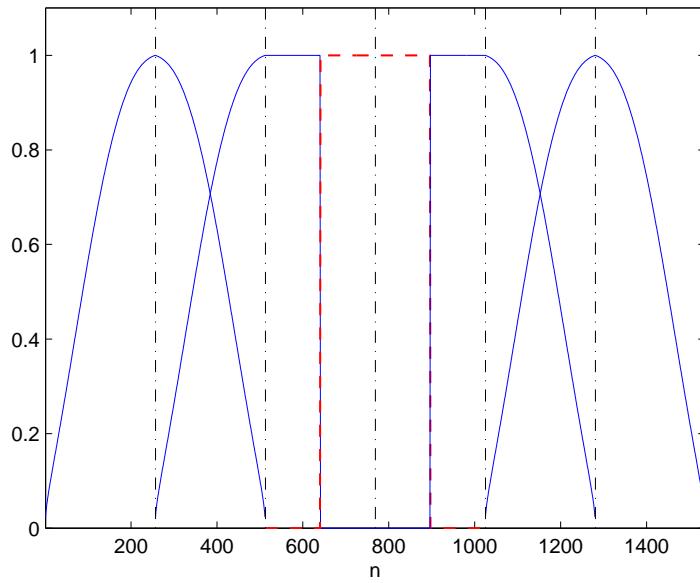


Figure 6.15 Cas particulier du fenêtrage adaptatif où la résolution temporelle devient maximale. Les fenêtres en traits continus sont codées dans le domaine fréquentiel alors que la fenêtre en pointillé est codée dans le domaine temporel.

Détection de transitoires

La détection des transitoires est la première étape avant la commutation de domaine de codage. On utilise un critère introduit dans [142], l'entropie de l'énergie segmentale du signal temporel. L'entropie est calculée en divisant chaque trame de 20 ms en segments de 2.5 ms. L'énergie normalisée σ_i^2 est calculé dans chacun des 8 segments en normalisant l'énergie du segment par l'énergie totale de la trame. L'entropie est alors calculée sur ces variables comme suit :

$$H = - \sum_{i=1}^8 \sigma_i^2 \log_2(\sigma_i^2) \quad (6.7)$$

Cette valeur permet de mesurer les brusques variations d'énergie de l'enveloppe temporelle. Si l'énergie est quasi constante sur l'ensemble de la trame, alors l'entropie H est élevée, et au contraire lors de transitions soudaines, H est petite. On utilise dans notre codeur un seuil unique de 2.5 pour distinguer les trames stationnaires des trames avec transitoires. La détection de transitoires est effectuée sur la synthèse du codeur de base au codage comme au décodage, ce qui préserve d'envoyer un bit indicateur.

TDAC temporelle

L'utilisation d'une MDCT nécessite l'emploi d'un recouvrement entre les fenêtres d'analyse et d'une addition à la synthèse pour satisfaire la condition de reconstruction parfaite. Le recouvrement de deux fenêtres successives et leur addition (*overlap-add*) permettent principalement de sommer deux descriptions d'un même coefficient partagé entre les deux fenêtres par des pondérations complémentaires afin de recouvrir le coefficient original ou du moins de s'en rapprocher. Cependant, le recouvrement et l'addition servent aussi à annuler le repliement temporel inséré dans le domaine fréquentiel par la MDCT. C'est ce repliement qui permet d'obtenir une transformation à échantillonnage critique. La Figure 6.16 explicite le processus du repliement temporel effectué par la MDCT en mettant en évidence la relation avec une DCT de taille $N/2$. La fenêtre d'analyse est divisée en quatre segments a, b, c et d de $N/4$ échantillons. Les segments a_R, b_R, c_R et d_R représentent les segments retournés des segments originaux. Si dans la chaîne il n'y a aucune erreur de codage insérée, alors on retrouve le signal original après le processus de recouvrement et d'addition avec la fenêtre précédente pour retrouver les $N/2$ premiers échantillons comme le montre la Figure 6.17, et avec la fenêtre suivante pour retrouver les $N/2$ derniers échantillons.

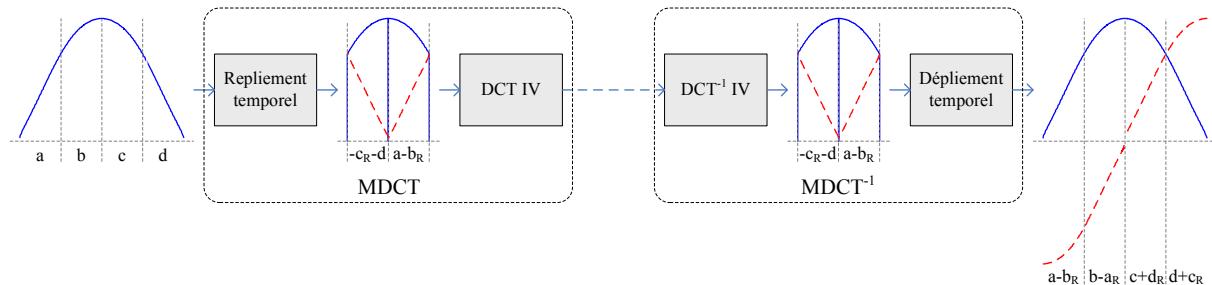


Figure 6.16 Illustration du repliement temporel lors d'une MDCT.

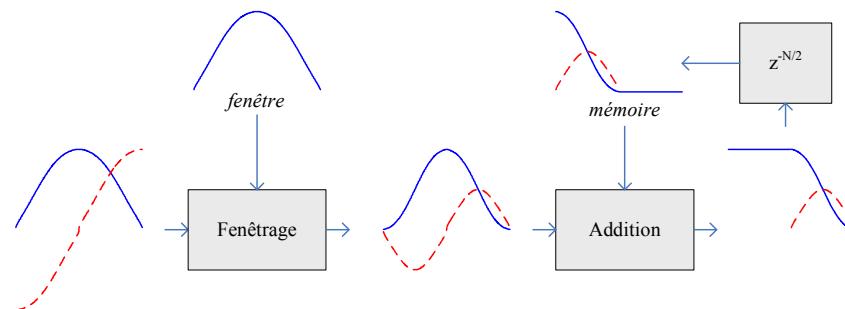


Figure 6.17 Illustration de la reconstruction parfaite à l'aide de la TDAC.

Pour éviter d'introduire un délai, on souhaite pouvoir alterner entre les domaines fréquentiel et temporel sans utiliser de fenêtres de transition dans le domaine fréquentiel. Afin d'assurer une reconstruction parfaite, il est nécessaire que les fenêtres d'analyse temporelle adjacentes à une fenêtre d'analyse fréquentielle puissent annuler le repliement temporel introduit par la MDCT. Il faut donc effectuer dans le domaine temporel le même repliement et dépliement temporel que la MDCT et la MDCT inverse. Le processus de codage dans le domaine temporel est décrit à la Figure 6.18. On constate que pour toute fenêtre de taille N , seulement $N/2$ coefficient sont codés, ce qui assure un échantillonnage critique. Ce processus, appelé TDAC temporel, permet de mettre en jeu dans le domaine temporel tout fenêtrage répondant aux conditions de reconstruction parfaite de la MDCT de l'équation 2.19. Pour la synthèse, on utilise le même procédé de recouvrement et d'addition de deux fenêtres successives que la MDCT en partageant la même mémoire.

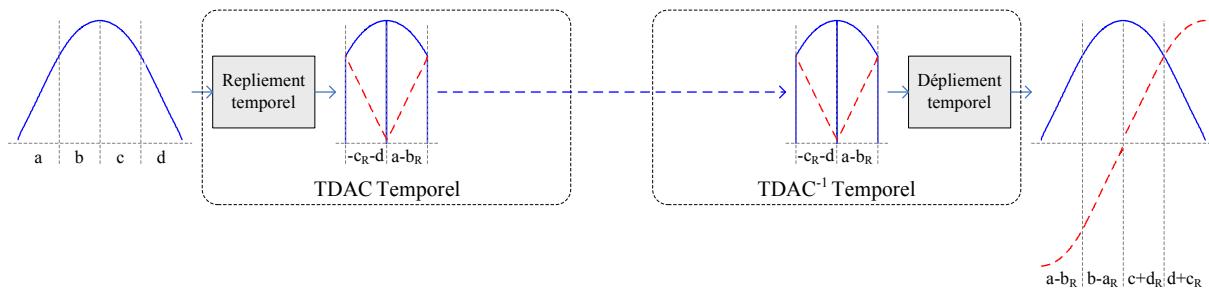
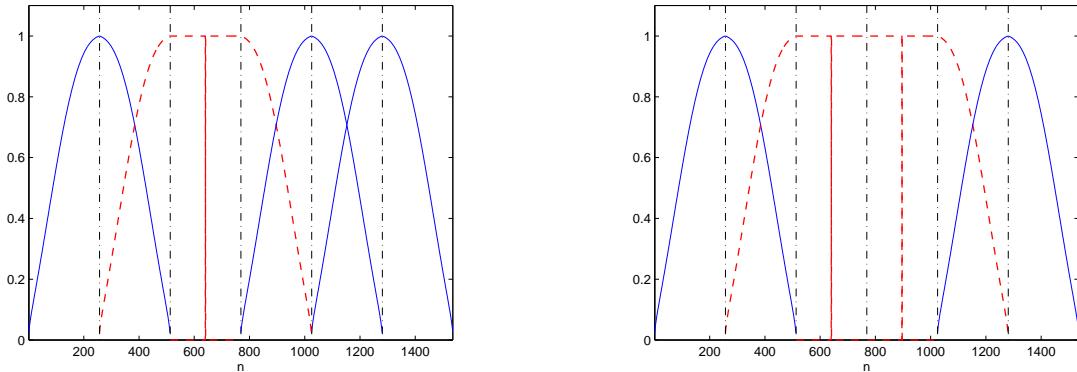


Figure 6.18 Repliement temporel dans le domaine temporel (TDAC temporelle) permettant la comptabilité avec la MDCT.

Dans notre codeur hiérarchique on utilise le fenêtrage décrit aux Figure 6.19 (a) et (b). Cette fois-ci, il n'y a plus de fenêtre de transition dans le domaine fréquentiel mais dans le domaine temporel. La présence de ces fenêtres est justifiée par le fait que dans le domaine temporel le recouvrement des fenêtres n'est pas nécessaire contre les effets de bloc. La fenêtre rectangulaire est alors la plus adaptée pour l'analyse temporelle. Les fenêtres de transition, de début et de fin d'analyse temporelle permettent alors de passer d'un recouvrement à 50% à une fenêtre rectangulaire sans recouvrement. S'il y a une seule trame isolée avec transitoires, alors on utilise seulement la fenêtre de début suivie de la fenêtre de fin (Figure 6.19 (a)). Par contre s'il y a plusieurs trames successives avec transitoires, on utilise entre les deux des fenêtres rectangulaires

(Figure 6.19 (b)). Pour les fenêtres rectangulaires, aucun repliement temporel n'a lieu, et les coefficients temporels sont directement codés sans pondération.



(a) Une trame de transitoires isolée

(b) Deux trames successives de transitoires

Figure 6.19 Commutation du domaine de codage avec des fenêtres de transition directement dans le domaine temporel.

6.3.3 Posttraitement fréquentiel

Pour réduire les artefacts très caractéristiques du modèle source-filtre du codeur de parole de la couche de base, on utilise avant le codage explicite des coefficients MDCT du signal de différence, le posttraitement fréquentiel proposé dans le chapitre 4. Comme on l'a déjà vu, ce traitement apporte un gain significatif lors du traitement des sons ne répondant pas au modèle de production de la parole. Pour des raisons de repliement temporel, le traitement est effectué dans le domaine de l'ODFT au lieu de celui de la MDCT. Il est seulement actif lors des trames stationnaires lorsque le codage a lieu dans le domaine fréquentiel. Le masque de l'équation 4.1, nécessaire au décodage, est remis en forme par l'algorithme 1 afin d'obtenir des statistiques plus favorables pour le codage des longueurs de plages. Ces derniers sont codés par un codage de Huffman à longueur variable pour former le sous-flux d'amélioration 1. Afin de contrôler le débit sortant, un codage multimodal du masque est en plus utilisé. Le débit de sortie est alors limité à 6.4 kbit/s. Si le codage entropique du masque reformaté a un débit supérieur à 6.4 kbit/s, alors on utilise simplement une décimation par 2 du masque original avant de le transmettre sans codage entropique comme le montre la Figure 4.17. Les codes de Huffman sont optimisés lors d'une séquence d'entraînement appliquée plusieurs fois afin de faire converger les codes après le

classement effectué par le codage multimodal. Le débit restant après le posttraitement fréquentiel est alloué à la quantification des coefficients spectraux provenant soit du signal original ou bien du signal de différence selon la valeur du masque. Ainsi, les sous-flux d'amélioration 1 et 2 forment un sous-flux d'amélioration total de 11.35 kbit/s. Le Tableau 6.1 donne les débits moyens alloués aux différentes parties du codage pour des signaux de nature différente. Il est à noter que le débit total à 24 kbit/s n'est pas une restriction, mais simplement le cas de figure étudié dans le chapitre.

Type de signaux	Couche de base AMR-WB	Couches d'amélioration		Total
		Post-traitement	LVQ	
Parole	12.65	1.52	9.83	24
Musique	12.65	4.85	6.5	24
Mixage parole et musique	12.65	2.55	8.8	24

TABLEAU 6.1 Débits moyens alloués aux différentes couches (kbit/s).

6.3.4 Performances avec optimisations

Les optimisations proposées au codeur hiérarchique ont été apportées à la suite de réflexions sur les mesures subjectives de la Figure 6.11. C'est donc au travers de nouvelles mesures subjectives que le codeur hiérarchique optimisé est évalué.

Mesures subjectives

Nous avons conduit le même test subjectif que dans la section 6.2 mais cette fois-ci en incorporant le codeur hiérarchique optimisé. Les résultats du test sont donnés à la Figure 6.20. On observe que le codeur hiérarchique optimisé se comporte beaucoup mieux pour de la parole et devient alors presque aussi performant que l'AMR-WB à 23.05 kbit/s. Cet apport est principalement dû au commutateur du domaine de codage réduisant grandement les problèmes d'étalement spectral. Pour la musique, le gain, bien qu'étant moins important, est quand même significatif avec presque 10 points de plus que le codeur hiérarchique non optimisé de la section 6.2. Ce gain mesure presque directement l'apport bénéfique du posttraitement fréquentiel. Néanmoins, les performances pour cette catégorie de signaux restent quand même significativement en dessous du G.722.1. Ceci s'explique par le faible débit alloué au codage par transformée (11.35 kbit/s). Un débit supérieur

alloué aux couches d'amélioration pourrait améliorer les performances du codeur hiérarchique par rapport à un codage par transformée classique.

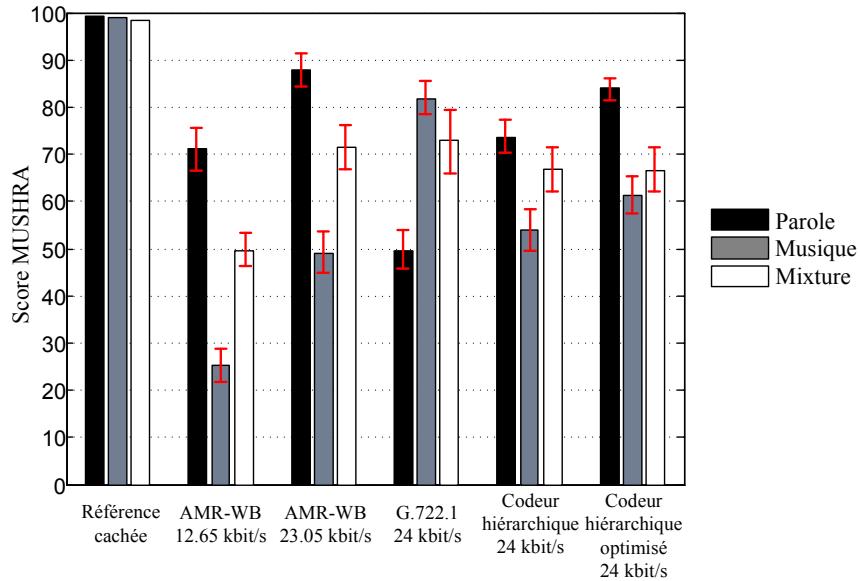


Figure 6.20 Résultats du test MUSHRA évaluant les optimisations apportées au codeur hiérarchique.

Le codeur hiérarchique optimisé est le codeur le plus universel des solutions comme l'atteste les Figures 6.21 (a) et (b). En effet, c'est le codage qui a la performance globale la plus haute, et qui plus est, affiche l'écart type général le plus faible. Les optimisations apportées au codeur hiérarchique permettent de gagner aux alentours de 10 points Mushra. La couche d'amélioration à 11.35 kbit/s, quant à elle, fait gagner en moyenne plus de 20 points Mushra au codeur de base AMR-WB à 12.65 kbit/s.

6.4 Conclusion

Nous avons présenté dans ce chapitre une structure globale de codage hiérarchique à base d'un codeur de parole large bande. Cette structure se repose sur l'imbrication d'un codeur de parole avec un codage par transformée. L'association d'un codage temporel avec un codage fréquentiel n'est pas nouvelle, mais la cohabitation n'est pas forcément aisée.

Dans une structure imbriquée à raffinements successifs, le codage par transformée traite un signal de différence. L'information supplémentaire à coder doit pouvoir améliorer la qualité audible de

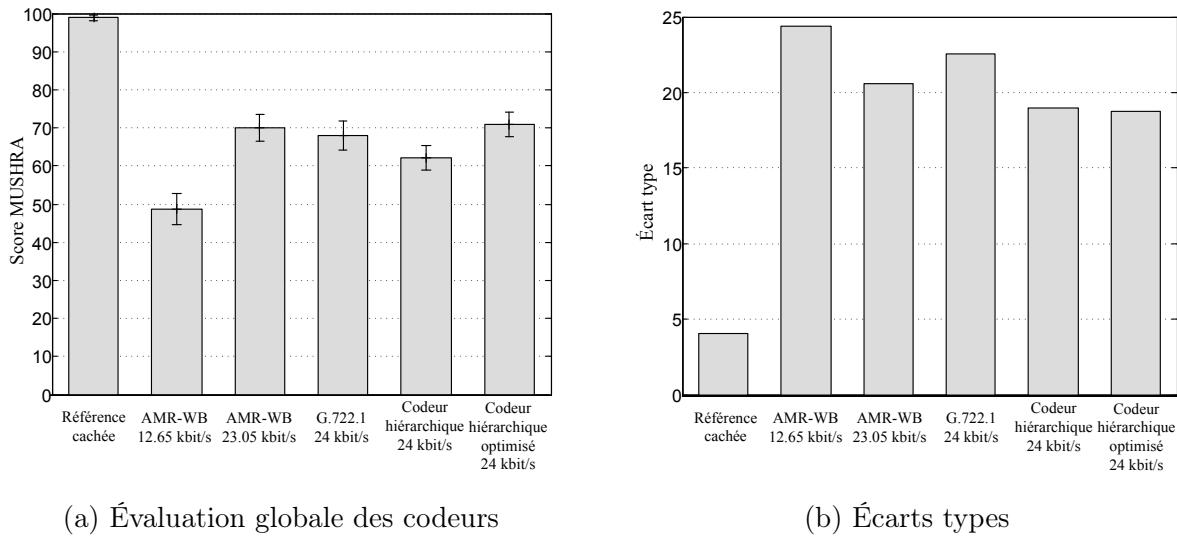


Figure 6.21 Résultats globaux sur l'ensemble des signaux.

la synthèse du codeur de base. Pour cette raison, on a proposé de mettre en série un procédé de masquage et une mise en forme du signal de différence afin de quantifier l'information la plus pertinente selon le modèle perceptuel.

D'autre part, le codage d'amélioration doit, dans la mesure du possible, rattraper les artefacts préalablement introduits par le codeur de base tout en évitant d'en ajouter d'autres. C'est pour ces raisons que nous avons introduit deux outils d'optimisation supplémentaires au codeur hiérarchique. Le premier, le commutateur du domaine de codage (section 6.3.2), permet d'éviter l'étalement spectral introduit généralement par le codage par transformée. Le deuxième, le post-traitement fréquentiel du chapitre 4, aide à rattraper le handicap des codeurs de parole pour certains sons musicaux.

En tenant compte de ces considérations, nous avons obtenu un codeur hiérarchique à 24 kbit/s à base du codeur AMR-WB à 12.65 kbit/s. Entre 12.65 et 24 kbit/s, le débit de décodage est complètement variable et jouit d'une granularité aussi fine que 1 bit. Le décodage graduel est assuré par la LVQ à raffinements successifs introduite au chapitre 5. Au final, le codeur hiérarchique proposé, malgré la forte contrainte sur son train binaire et sa structure, est une solution concurrentielle à débits équivalents par rapport aux solutions existantes à débits fixes et souvent dédiées à un seul type de signal. De plus, la structure est très flexible et peut supporter divers codeurs de parole dans la couche de base ainsi que d'autres débits de codage.

CHAPITRE 7

Codage Hiérarchique avec Extension de Bande

7.1 Introduction

L'objectif de ce chapitre est de proposer une solution de codage pour pouvoir améliorer la qualité de restitution d'un codeur de base deux façons différentes. D'une part, on veut pouvoir raffiner graduellement la description du signal dans la bande de fréquence originellement transmise, comme il a été fait dans le chapitre 6. D'autre part, on veut étendre la largeur de bande de la synthèse. Dans ce chapitre, on considère en particulier l'extension d'un codeur de parole bande étroite en un codeur audio large bande. À partir du standard G.729 à 8 kbit/s, on construit un codeur hiérarchique pouvant dès 10 kbit/s obtenir une première synthèse large bande. La qualité de la synthèse est alors améliorée pour incrément du débit jusqu'à 32 kbit/s.

Comme on souhaite obtenir un raffinement graduel de la qualité de restitution en fonction du débit de décodage, nous optons pour une extension de la bande transmise faisant intervenir en même temps un codage et une estimation des composantes de la bande manquante à transmettre. En effet, pour un débit supplémentaire donné par rapport au débit du codeur de base, seules certaines composantes de la bande manquante peuvent être décrites explicitement. La description des autres composantes n'est alors pas disponible au décodage. Deux approches sont alors possibles. Dans la première approche, les composantes non transmises ne sont pas prises en compte à la synthèse. Dans ce cas, le spectre de la synthèse s'enrichira au fur et à mesure que le débit disponible augmentera. Dans la seconde approche, le décodeur estime les composantes non transmises afin de jouir dès les plus faibles débits d'une synthèse pleine bande. Dans ce cas, le taux de composantes codées prendra le dessus sur le taux de composantes estimées au fur et à mesure que le débit augmentera. Nous optons pour cette dernière approche sachant que la largeur de bande de la synthèse est un facteur prédominant de la qualité perceptuelle [79]. De plus, il existe des méthodes efficaces pour estimer la bande manquante à partir de la bande déjà transmise appelée bande de base (cf. section 3.2). L'extension de bande est alors hybride conjuguant de façon adaptable un codage et une estimation de la bande manquante selon le débit disponible au décodeur. Un

même type de codage hybride a déjà été présenté pour étendre le codeur de parole G723.1 à 6.4 kbit/s [70].

L'extension de bande hybride, faisant intervenir à la fois codage et estimation, amène à faire certains choix déterminants. Ces choix se ramènent essentiellement à un problème d'allocation des ressources. Pour chaque débit de décodage, il faut déterminer les composantes à coder et les composantes à estimer. Deux principaux critères rentrent en ligne de compte :

- La robustesse de l'estimation. Toutes les composantes ne sont pas estimables d'une façon homogène. En effet, elles sont souvent estimées par rapport aux composantes déjà transmises. Dans notre cas, il s'agira d'estimer la bande manquante de 4000 à 8000 Hz à partir d'une synthèse bande étroite de 0 à 4000 Hz. Pour ces fréquences, l'enveloppe spectrale de la bande manquante partage peu d'information avec celle de la bande étroite [143]. Une simple estimation s'avère alors souvent non suffisante pour obtenir une qualité acceptable. Par contre, la structure fine montre des similarités beaucoup plus importantes entre les basses et les hautes fréquences.
- L'importance perceptuelle. Les composantes les plus importantes selon la perception auditive devront être décrites les premières. Encore une fois, l'enveloppe spectrale est une composante importante, et doit être transmise en priorité. Selon ce même critère perceptuel, une allocation binaire adéquate devra être faite entre les différentes sous-bandes de la structure fine. Cette allocation connue du codeur et du décodeur donnera un ordre de transmission des différentes sous-bandes de la structure fine ainsi que le débit alloué pour chacune d'elle.

Au vu de ces critères, il paraît primordial que l'enveloppe spectrale soit la première composante de la bande manquante à être transmise. D'autre part, la restitution d'une synthèse de bonne fidélité semble illusoire sans une bonne description de cette composante. C'est une position partagée par plusieurs travaux [81, 83, 84, 144, 85, 70, 134] qui, bien qu'utilisant une estimation pour l'extension de bande, codent tout de même l'enveloppe spectrale de la bande manquante. On contraint ainsi notre extension de bande à envoyer en premier lieu une description de l'enveloppe spectrale de la bande manquante. Si cette description n'est pas disponible au décodage, la largeur de bande de la synthèse ne sera pas étendue. Lorsque l'enveloppe spectrale est décodée, les composantes de la structure fine de la bande manquante pourront être, soit codées soit estimées.

Finalement, on peut résumer les hypothèses de recherche pour le codeur hiérarchique avec extension de bande par la Figure 7.1. Elle schématise le spectre de la synthèse à une trame et à un débit donnés. On peut constater qu'en bande de base l'enveloppe spectrale est obtenue par le

codage de base qui est dans notre cas un codeur de parole bande étroite. Il permet aussi d'avoir une première reproduction des sous-bandes de la bande de base. Le codage d'amélioration fournit quant à lui l'enveloppe spectrale de la bande manquante. Selon le débit disponible au décodage, il peut en plus fournir une description supplémentaire des sous-bandes de la bande de base ainsi que décrire certaines sous-bandes de la bande manquante. Celles qui ne seront pas décodées seront tout simplement estimées.

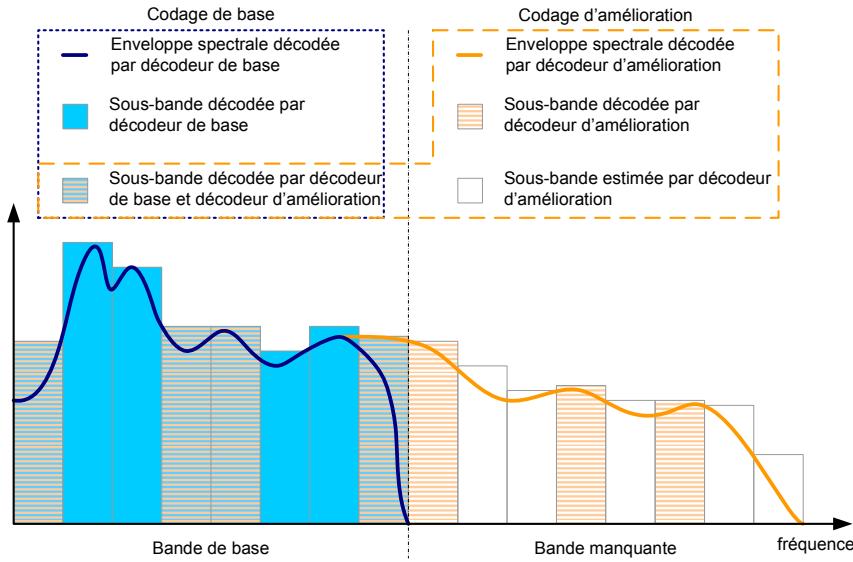


Figure 7.1 Décodage d'une trame à un débit de décodage donné par le codage hiérarchique avec extension de bande.

Dans la section suivante, nous allons présenter le codeur hiérarchique à extension de bande en détail. Nous allons ensuite discuter de différents compromis et optimisations qui peuvent être apportés au codeur selon l'application visée. Nous conclurons le chapitre par une évaluation subjective du codeur.

7.2 Structure hiérarchique du codeur

7.2.1 Principe

Il existe principalement deux approches pour étendre la bande transmise comme nous l'avons vu à la section 3.2 : l'extension en sous-bandes et l'extension en pleine bande. Pour l'extension en pleine bande, un effort doit être fait lors du codage pour distinguer la bande de base de la

bande manquante. C'est pour cette raison que nous adoptons une extension en sous-bandes qui permet de reporter cette complexité en amont. En effet, on décompose au préalablement le signal d'entrée en deux sous-bandes : une sous-bande correspondante à la bande de base de 0 à 4 kHz et une sous-bande correspondante à la bande manquante de 4 à 8 kHz.

La Figure 7.2 schématise la structure hiérarchique du codeur. Le filtre passe-bas $H_L(z)$ et le filtre passe-haut $H_H(z)$ permettent de décomposer le signal d'entrée échantillonné à 16 kHz en deux bandes de fréquence uniformes échantillonnées à 8 kHz : la bande basse (0-4 kHz) et la bande haute (4-8 kHz). Les filtres passe-bas $G_L(z)$ et passe-haut $G_H(z)$ sont les filtres de synthèse. Le codeur de parole bande étroite, G729, traite seulement la bande basse. Il génère le sous-flux de base à 8 kbit/s. La synthèse bande étroite obtenue est ensuite raffinée par un codage par transformée. La bande haute est transmise en parallèle par un codage paramétrique et par un codage par transformée. Le codage paramétrique transmet le sous-flux d'amélioration 1 à 2 kbit/s. Il est suffisant pour obtenir une première estimation de la bande manquante. Le décodeur doit alors estimer la structure fine de la bande manquante. Le codage par transformée permet de raffiner la qualité de la bande haute en codant certaines composantes fréquentielles préalablement estimées par le décodeur. Les deux descriptions, celle estimée et celle codée, sont multiplexées dans le domaine fréquentiel au décodeur. Les codages par transformée de la bande basse et de la bande haute forment un unique sous-flux d'amélioration 2. Il a un débit total de 22 kbit/s, mais peut être décodé à des débits variables. Le débit total du codeur hiérarchique est de 32 kbit/s.

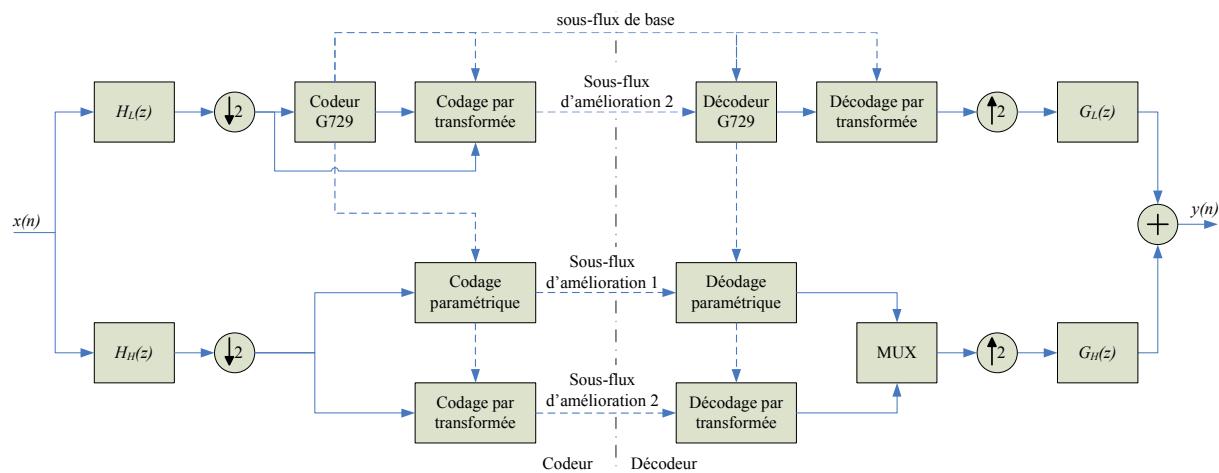


Figure 7.2 Schéma de principe du codage hiérarchique avec extension de bande.

Le schéma bloc de la Figure 7.4 donne une description détaillée du codeur. Le signal d'entrée x est directement décomposé par un banc de filtres miroir en quadrature (*Quadrature Mirror Filters*, QMF). On utilise des filtres FIR à phase linéaire d'ordre 63 comme ceux optimisés par Johnston [145]. Les réponses fréquentielles en amplitude des deux filtres d'analyse $H_L(z)$ et $H_H(z)$ sont données à la Figure 7.3 (a). Les Figure 7.3 (b) et (c) donnent la réponse en amplitude du filtre total résultant de l'analyse suivie de la synthèse par la structure QMF. La reconstruction est quasi-parfaite lorsqu'il n'y a aucune erreur de codage introduite. Le retard engendré est au total de 63 échantillons temporels.

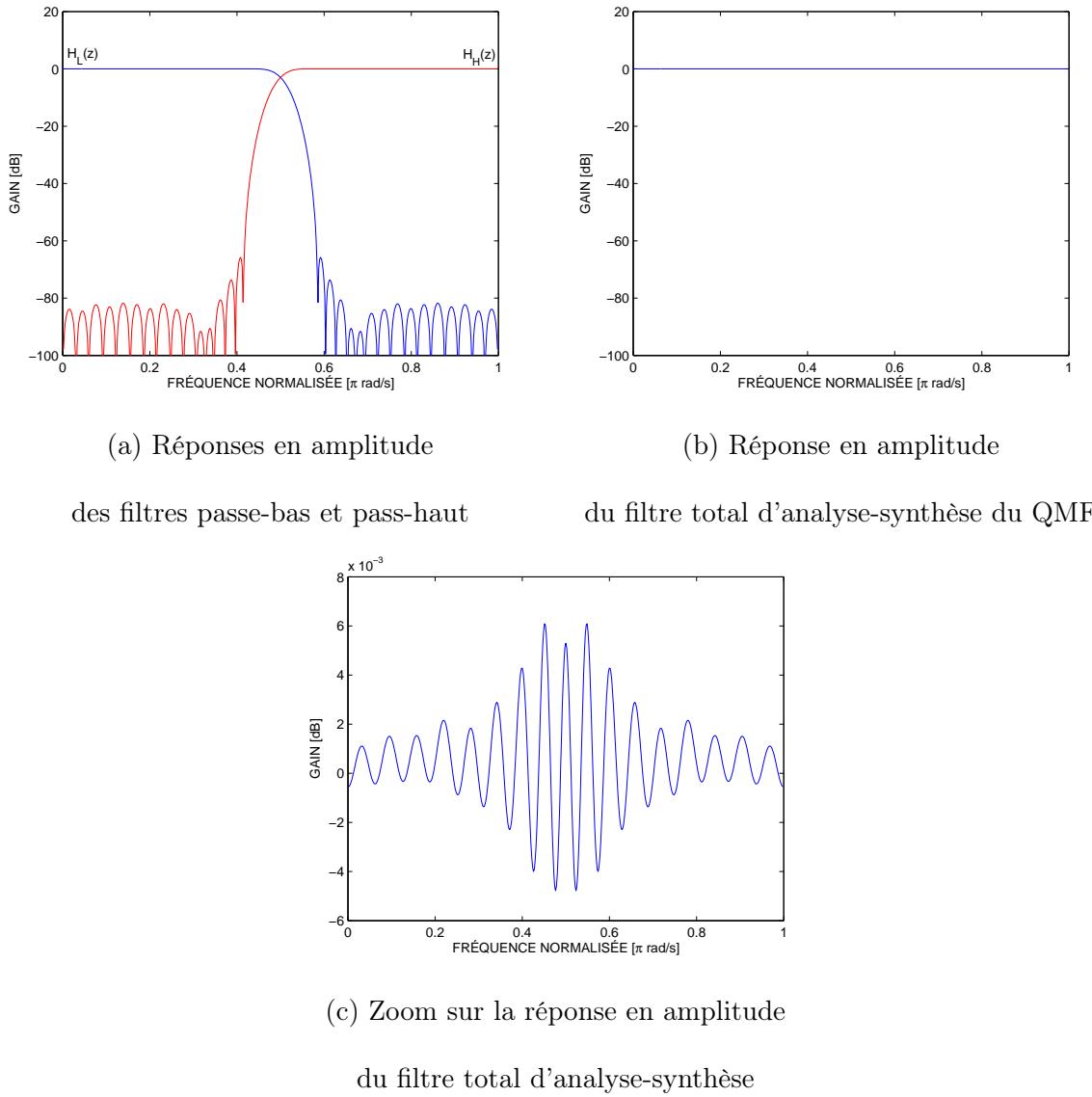


Figure 7.3 Caractéristique des filtres QMF d'ordre 63.

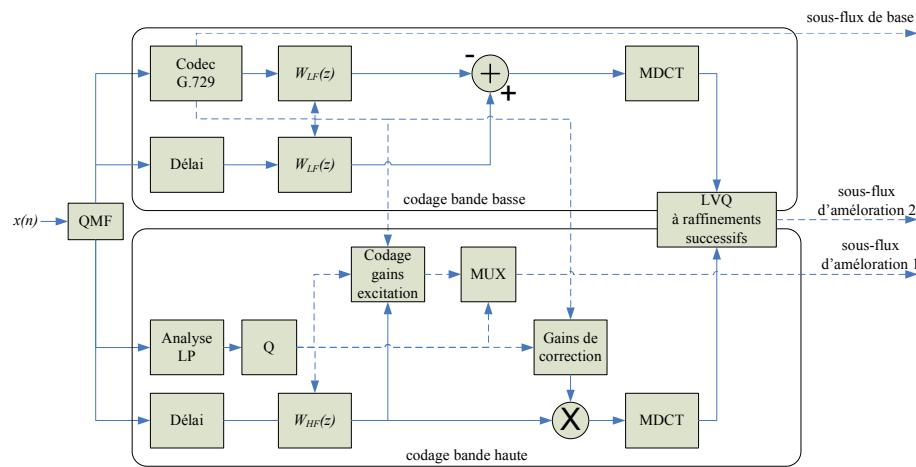


Figure 7.4 Schéma bloc du codeur hiérarchique avec extension de bande.

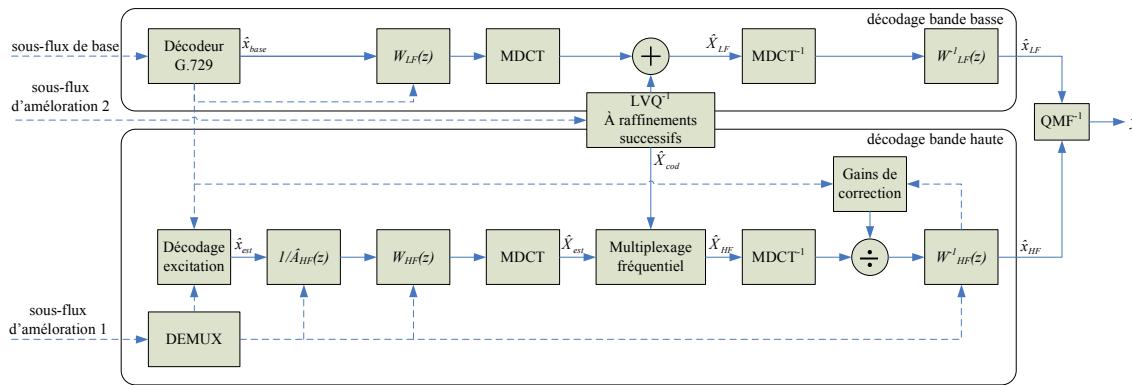


Figure 7.5 Schéma bloc du décodeur hiérarchique avec extension de bande.

Dans la bande basse, le signal est dans un premier temps codé par le codeur de base, le G.729 à 8 kbit/s, et forme le sous-flux de base. L'erreur de codage du G729 est dans un second temps pondérée par un filtre de pondération $W_{LF}(z)$ pour se mettre dans un domaine perceptuel. L'erreur pondérée est alors codée dans le domaine de la MDCT. Le filtre pondérateur $W_{LF}(z)$ s'exprime en fonction du filtre de synthèse $1/\hat{A}_{LF}$ transmis par le G729 :

$$W_{LF}(z) = \frac{\hat{A}_{LF}(z/\gamma_1)}{\hat{A}_{LF}(z/\gamma_2)}, \quad 0 \leq \gamma_2 < \gamma_1 \leq 1 \quad (7.1)$$

où γ_1 et γ_2 sont les facteurs contrôlant la pondération spectrale. Le filtre pondérateur $W_{LF}(z)$ utilise les mêmes facteurs de pondération que ceux du G.729 dans la boucle d'analyse par synthèse afin de blanchir l'erreur de codage et d'utiliser le même critère perceptuel. Les facteurs sont calculés par le G729 et sont fonction de la forme de l'enveloppe spectrale [5].

De l'autre côté, dans la bande haute, une prédiction linéaire (analyse LP) du signal original est opérée. Elle permet d'obtenir l'enveloppe spectrale de la bande manquante à travers le filtre de synthèse $1/A_{HF}(z)$. Le filtre pondérateur $W_{HF}(z)$ en est déduit de la même façon que l'équation 7.1. Les facteurs de pondération sont choisis de telle sorte que le filtre pondérateur inverse soit assez proche de l'enveloppe spectrale. On souhaite en effet blanchir un maximum le signal à coder dans le domaine pondéré tout en conservant quelques propriétés perceptuelles. On choisit donc $\gamma_1 = 0.9$ et $\gamma_2 = 0$. La structure fine du spectre de la bande manquante peut être retrouvée de deux façons au décodage. Soit par estimation de l'excitation du filtre de synthèse, soit par quantification des coefficients pondérés dans le domaine de la MDCT. L'estimation de l'excitation se fait à l'aide de l'excitation codée du G729. Pour que l'estimation soit plus juste, on transmet en plus des gains en énergie g_i . Les coefficients de la prédiction linéaire ainsi que les gains en énergie suffisent pour obtenir une première synthèse large bande. Ils forment le sous-flux d'amélioration 1 du codage paramétrique à 2 kbit/s. Les coefficients MDCT de la bande basse et haute sont tous codés ensemble par la même quantification multidébit. On utilise la LVQ à raffinements successifs introduite au chapitre 5. Pour pouvoir utiliser le principe du remplissage des eaux sur l'ensemble des deux bandes, on calcule des gains de correction dans la bande haute pour homogénéifier l'énergie sur tout le spectre de 0 à 8 kHz.

Le schéma bloc de la Figure 7.5 donne une description détaillée du décodeur. Le sous-flux de base permet d'obtenir une première synthèse bande étroite \hat{x}_{base} pour un débit de 8 kbit/s. La prise en compte du sous-flux d'amélioration 1 permet de décoder l'enveloppe spectrale de la bande haute par l'intermédiaire des coefficients du filtre de synthèse $1/A_{HF}(z)$, ainsi que les gains en énergie

g_i . La bande haute peut être alors estimée en répliquant l'excitation décodée par le G729 (\hat{x}_{est}). On obtient la première synthèse large bande pour un débit de 10 kbit/s. La prise en compte graduelle du sous-flux d'amélioration 2 remplace progressivement les composantes estimées \hat{X}_{est} par des composantes codées \hat{X}_{cod} . Dans la bande basse, les vecteurs décodés correspondants au signal de différence sont sommés aux coefficients de la synthèse bande étroite. Dans la bande haute, les vecteurs estimés et codés sont multiplexés dans le domaine fréquentiel.

7.2.2 Codage paramétrique de la bande haute

Principe

Nous utilisons un modèle source-filtre semblable à celui de la production de la parole pour générer la bande haute. Le filtre autorégressif de synthèse $1/A_{HF}(z)$ est obtenu par une prédiction linéaire de la bande haute. On utilise comme le G729 un ordre 10. Les coefficients de la prédiction sont quantifiés dans le domaine des LSFs. Le résidu du filtre de synthèse, c.-à-d. l'excitation, doit être modélisé. On utilise dans le codage paramétrique, une estimation de l'excitation ce qui permet pour un faible débit d'obtenir une première synthèse large bande. Elle est obtenue par un repliement spectral de l'excitation de la bande basse, codée par le G729, vers la bande haute. Le repliement spectral est un effet miroir autour de la fréquence normalisée $\pi/2$ qui est généralement réalisé dans le domaine temporel par un suréchantillonnage de facteur 2 [100]. Dans le cas d'une décomposition en sous-bandes par des filtres QMF, les fréquences de la bande haute sont inversées à cause du sous-échantillonnage. En conséquence, le repliement spectral consiste à une simple copie de l'excitation de la bande basse dans la bande haute. Un ajustement de l'énergie de l'excitation estimée est tout de même nécessaire pour obtenir une qualité acceptable. Le codage paramétrique transmet ainsi, en plus des LSFs de la prédiction linéaire, 4 gains d'ajustement en énergie, g_0, \dots, g_3 , par trame de 20 ms.

La Figure 7.6 résume le procédé du codage paramétrique de la bande haute. L'analyse LP comprend une prédiction linéaire ainsi que la quantification des coefficients de la prédiction par l'intermédiaire des LSFs. On obtient un filtre de synthèse $1/\hat{A}_{HF}(z)$ pour la bande haute. L'excitation codée par le G729 sert d'estimation de la bande haute. L'excitation obtenue est réajustée par des gains d'ajustement en énergie, g_0, \dots, g_3 . Ces gains sont calculés en comparant l'énergie du signal estimé de la bande haute avec celle du signal original dans le domaine pondéré et pour chaque sous-trame de 5 ms, c.-à-d. 40 échantillons. Ils sont ensuite codés à la suite d'une

prédition d'ordre 1 par des gains de correction $\bar{g}_0, \dots, \bar{g}_4$. Les gains de correction correspondent au rapport entre le gain du filtre de synthèse $1/\hat{A}_{LF}(z)$ et celui de $1/\hat{A}_{HF}(z)$ pour un signal aux fréquences s'étalant autour de 4000 Hz, comme le montre la Figure 7.7. Cette fréquence correspond à la même fréquence normalisée π pour les deux bandes de fréquence, basse et haute, toutes deux échantillonnées à 8 kHz. Le résidu de la prédition $\tilde{g}_0, \dots, \tilde{g}_4$ est codé par une quantification vectorielle de dimension 4. Ce procédé a été initialement introduit dans le standard AMR-WB+ [134] pour son extension paramétrique de la bande.

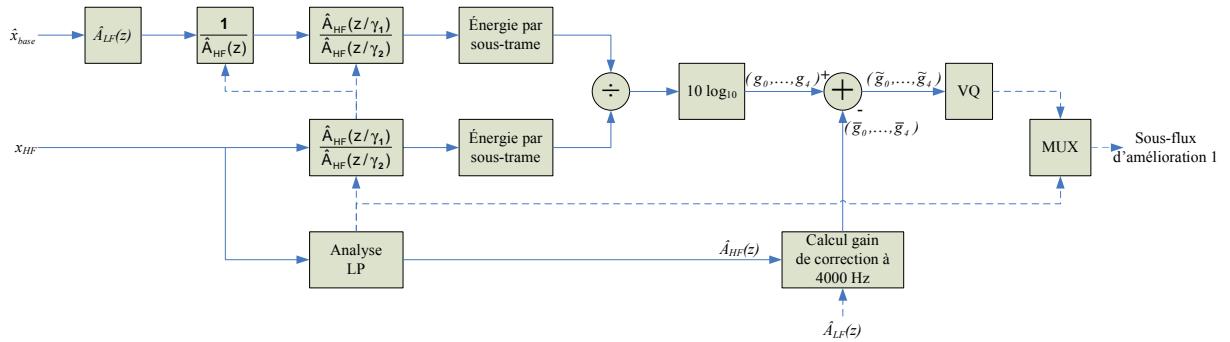


Figure 7.6 Codage paramétrique de l'excitation de la bande haute.

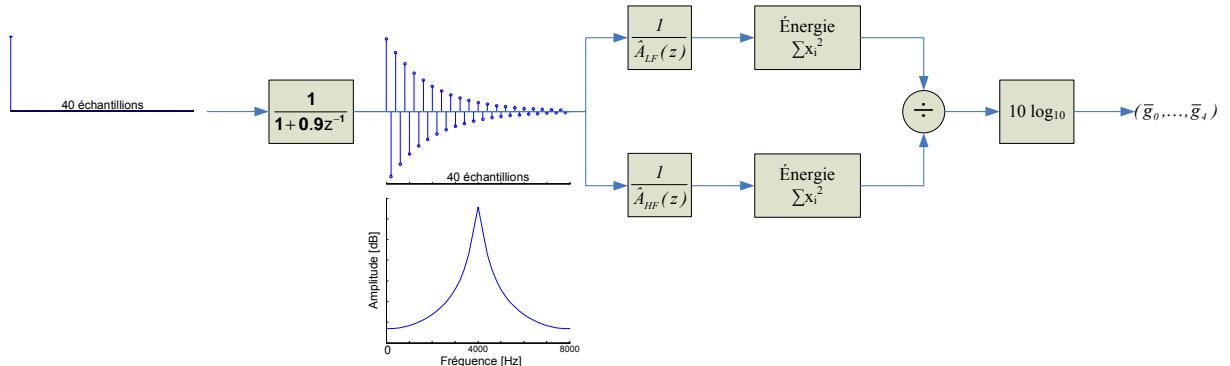


Figure 7.7 Calcul des gains de correction servant à la prédition des gains en énergie de l'excitation de la bande haute.

Quantification des LSFs

On utilise la même prédition linéaire d'ordre 10 que celle utilisée par le G729. Par contre, on ne peut pas réutiliser la même quantification, du fait que les dictionnaires ne sont pas optimisés pour la bande haute et que le débit alloué n'est plus le même. L'enveloppe spectrale est codée avec un

LSFs	1	2	3	4	5	6	7	8	9	10
1 ^{er} étage			6			6				
2 ^{ème} étage		6				6				
Total						24				

TABLEAU 7.1 Allocation binaire (bits/trame) pour la quantification vectorielle des LSFs.

débit de 1.2 kbit/s contrairement aux 1.8 kbit/s du G.729. L'enveloppe est calculée toutes les 20 ms au lieu de 10 ms. En outre, le codage d'amélioration ne peut tirer bénéfice de la prédition intertrame comme le G.729. C'est une des contraintes handicapantes dues à la hiérarchisation. La Figure 7.8 illustre l'analyse du signal et la quantification des coefficients de la prédition dans le domaine des LSFs. La même fenêtre d'analyse que le G.729 est utilisée, à savoir une fenêtre asymétrique de 30 ms avec 5 ms de chevauchement avec la trame future (*lookahead*). Le délai de 5 ms engendré est donc le même que celui du codeur G.729. On obtient après interpolation linéaire un filtre de synthèse $1/\hat{A}_{HF}(z)$ ainsi qu'un filtre pondérateur $W_{HF}(z)$ toutes les 5 ms.

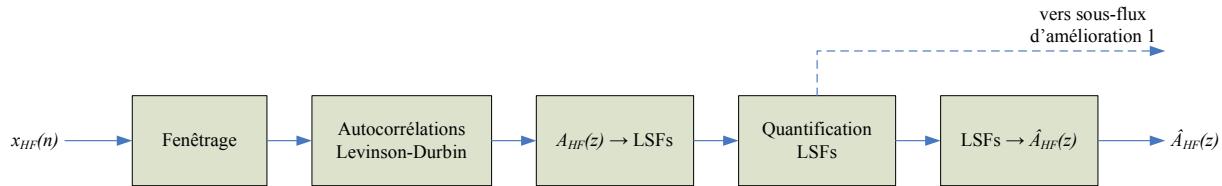


Figure 7.8 Codage de l'enveloppe spectrale par une analyse LP.

On code les LSFs avec une quantification vectorielle à deux étages. L'allocation binaire de la quantification est résumée par le Tableau 7.1. Pour évaluer la performance du codage, on calcule la distorsion spectrale moyenne (SD) sur 2 minutes de matériaux audio très divers. Elle est calculée sur N trames comme suit :

$$SD = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} \left[10 \log_{10} \frac{A_{HF}(e^{(j2\pi n/N)})}{\hat{A}_{HF}(e^{(j2\pi n/N)})} \right]^2} \quad (7.2)$$

Le Tableau 7.2 rapporte les performances obtenues. Il est généralement acquis que si la distorsion moyenne est au-dessous de 1 dB, l'erreur de codage est imperceptible. Dans notre cas la valeur moyenne est légèrement au-dessus de 1 dB, mais les erreurs de grande amplitude peu probables.

Quantification des gains de l'excitation

SD moyenne [dB]	Dépassements [%]	
	2-4dB	>4dB
1.35	13.32	0.32

TABLEAU 7.2 Performances du codage de l'enveloppe spectrale.

Les gains en énergie de l'excitation, g_0, \dots, g_3 , sont codés par l'intermédiaire du résidu, $\tilde{g}_0, \dots, \tilde{g}_4$, de la prédiction sur les gains par $\bar{g}_0, \dots, \bar{g}_4$. On utilise une quantification vectorielle stochastique de dimension 4 à deux étages pour une trame de 20 ms. Le Tableau 7.3 résume les caractéristiques de la quantification des gains. La résolution totale est de 4 bits par dimension, ce qui donne un débit de 0.8 kbit/s. On obtient finalement un codage paramétrique de la bande haute pour un débit total de 2 kbit/s. Une première synthèse large bande est ainsi disponible pour un débit de 10 kbit/s.

Sous-trame	1	2	3	4
1 ^{er} étage			8	
2 ^{ème} étage			8	
Total			16	

TABLEAU 7.3 Allocation binaire pour le codage des gains en énergie de l'excitation de la bande haute.

7.2.3 Codage par transformée

Le codage du signal de différence dans la bande basse entre la synthèse du G729 et le signal original, ainsi que le signal original dans la bande haute permet de raffiner la synthèse large bande obtenue à 10 kbit/s. Les deux signaux sont pondérés spectralement par leur filtre pondérateur respectif. Les deux signaux résultants sont ensuite codés ensemble dans le domaine de la MDCT par une même quantification multidébit utilisant une allocation globale. On se ramène alors à un codage de type TCX [62].

Modèle perceptuel

Les deux signaux pondérés à coder, venant de la bande basse et de la bande haute, sont regroupés dans le domaine de la MDCT avant d'être traités globalement par la quantification vectorielle à raffinements successifs. L'allocation est réalisée par un gain global g suivant ainsi le principe du

remplissage inverse des eaux. Les coefficients de la MDCT des deux bandes sont alors normalisés uniformément par le même gain g . Pour que l'allocation soit cohérente avec le modèle perceptuel, il faut que les deux signaux pondérés aient des énergies homogènes et cohérentes. Or, les filtres de pondération de la bande basse et de la bande haute n'ont pas les mêmes gains en énergie. Il est donc nécessaire d'utiliser des gains de correction pour homogénéifier l'énergie des deux signaux. Les gains de correction sont calculés de la même façon que ceux utilisés pour le codage paramétrique de la bande haute à la Figure 7.7. Par contre cette fois-ci il s'agit de mettre en correspondance les gains des filtres de pondération $W_{LF}(z)$ et $W_{HF}(z)$ autour de la fréquence normalisée π . Les gains de correction sont appliqués au signal de la bande haute. On obtient après regroupement des coefficients de la MDCT des deux bandes, une cible unique à quantifier.

Les Figures 7.9 (a) et (b) illustrent pour une trame donnée, la génération de la cible à quantifier. On constate que l'erreur de codage du G.729 est bien mise en forme par le filtre pondérateur inverse $W_{HF}^{-1}(z)$. Le filtre pondérateur $W_{LF}(z)$ blanchit alors le signal d'entrée du codage par transformée. Dans la bande haute, le signal à coder est égal au signal original. Le filtre pondérateur inverse $W_{HF}^{-1}(z)$ étant très conservateur suit d'assez près l'enveloppe spectrale de la bande haute. On obtient alors un signal presque blanchi après pondération. Au final, avec l'aide des gains correcteurs, l'ensemble des deux signaux forme une cible dont le spectre est très plat et uniforme.

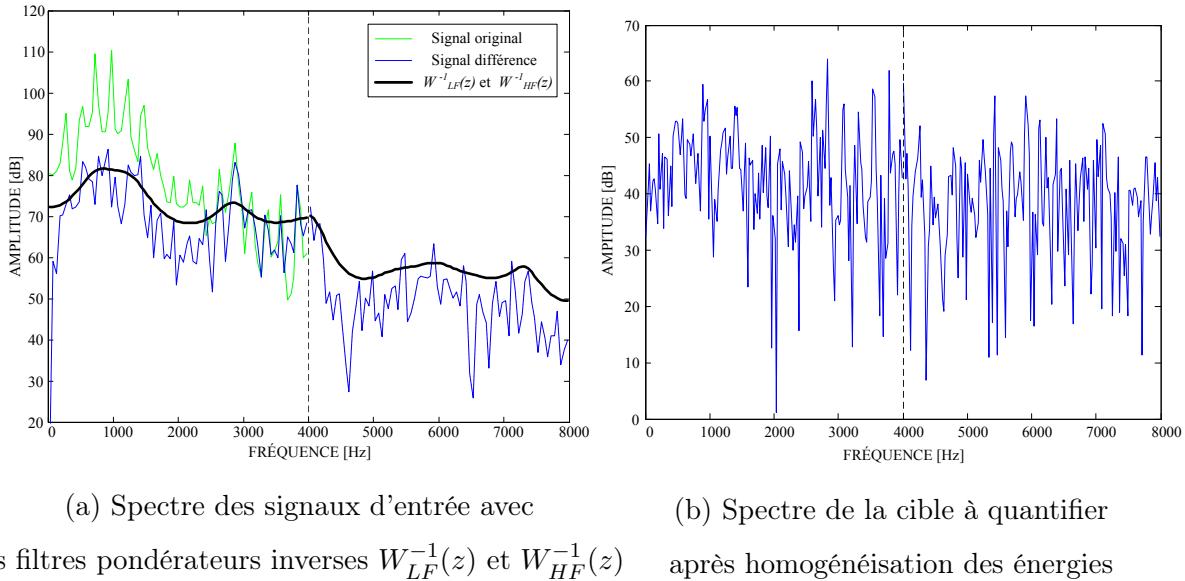


Figure 7.9 Illustration de la génération de la cible à quantifier.

Quantification vectorielle de la cible

On utilise la LVQ à raffinements successifs introduite au chapitre 5 pour coder la cible dans le réseau régulier de points RE_8 de dimension 8. La quantification est multidébit et utilise différents dictionnaires Q_n de résolution $n/2$ bits par dimension. Pour $n < 5$, on utilise des dictionnaires avec une région de support pyramidale, optimisée pour une source sans mémoire laplacienne. Le dictionnaire Q_0 représente le vecteur nul alors que le dictionnaire Q_1 n'est pas utilisé. Ce sont les dictionnaires de base utilisés par la recommandation AMR-WB+ [12]. Pour $n > 5$ les dictionnaires sont générés à partir des dictionnaires de base Q_3 et Q_4 alternativement par l'extension de Voronoï graduelle présentée au chapitre 5. On utilise, comme dans la recommandation AMR-WB+, un gain globale g normalisant la cible pour rentrer dans le budget de bits [67]. L'allocation est globale et adaptative entre les deux bandes. On code seulement les coefficients de 0 à 7 kHz, ce qui représente 35 vecteurs de dimension 8. Les fréquences au-delà de 7 kHz sont atténuées à la synthèse par un filtre passe-bas.

La Figure 7.10 donne un histogramme des coefficients de la cible normalisée par le gain g . On observe que la densité de probabilité associée à la cible est fortement concentrée autour de zéro. Elle a une forme plus pointue qu'une distribution laplacienne ($\gamma = 1$) et peut se modéliser par une gaussienne généralisée de paramètre $\gamma = 0.85$. Les dictionnaires de base, Q_0, \dots, Q_4 , sont donc relativement bien adaptés à la source.

Les Figures 7.11 et 7.12 proposent des statistiques sur les numéros des dictionnaires utilisés par la quantification pour deux différents débits de décodage, 21 kbit/s et 32 kbit/s, du même train binaire originellement codé à 32 kbit/s. Le débit alloué au codage à la quantification est alors de 22 kbit/s pour chacun des deux débits. Par contre, lors du décodage au débit de 21 kbit/s, le train binaire est tronqué et seulement les premiers 11 kbit/s de la quantification sont décodés. On a utilisé une longue séquence audio de musique et de parole pour compiler les statistiques. À ces débits, les dictionnaires utilisés sont essentiellement des dictionnaires de base comme le montre la Figure 7.11. L'extension a lieu environ une fois sur dix. Lorsque que le train binaire est tronqué à 21 kbit/s, les vecteurs codés par les dictionnaires les plus petits ne sont plus décodés, et le dictionnaire $n = 0$ est alors utilisé à la place. Ainsi, les occurrences des dictionnaires $n = 2, 3, 4$ sont reportées en totalité ou en partie sur les occurrences du dictionnaire nul $n = 0$. Par contre pour les dictionnaires de taille suffisante $n > 4$, un ou plusieurs code-vecteurs de la description

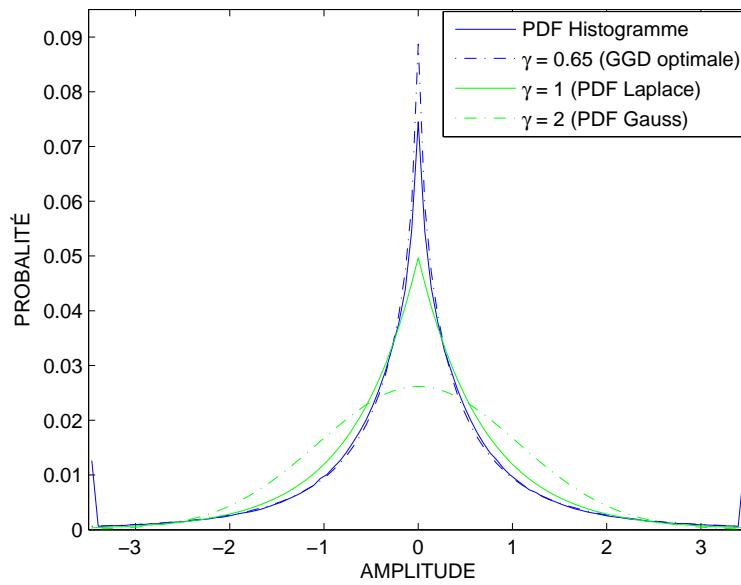


Figure 7.10 Densité de probabilité des coefficients de cible de la MDCT après normalisation, et sa modélisation par plusieurs gaussiennes généralisées de différents γ (les valeurs à -3.5 et 3.5 représentent les probabilités cumulées pour les valeurs <-3.5 et >3.5 respectivement).

du vecteur codé sont encore pris en compte. Les occurrences de ces dictionnaires ne changent presque pas.

À la Figure 7.12, on observe que le numéro de dictionnaire moyen utilisé par vecteur est plus important dans les basses fréquences. À 32 kbit/s, il décroît graduellement vers les hautes fréquences. L’allocation est donc plus favorable aux basses qu’aux hautes fréquences. On peut en conclure aussi que le G.729, même dans le domaine pondéré où il minimise son erreur de codage, génère une erreur importante dans les basses fréquences. Cela démontre que le modèle du G729 est mal adapté pour certains sons. Si le modèle du G729 était adapté quelque soit la source, le numéro de dictionnaire moyen devrait être constant dans la bande basse. Lorsque le train binaire est tronqué à 21 kbit/s, la différence entre les basses et les hautes fréquences s’accentue. Encore une fois, on visualise que les vecteurs codés par de petits dictionnaires sont les premiers à être ignorés. L’ordre de transmission est bien fonction de l’amplitude des vecteurs codés, suivant ainsi le principe du remplissage inverse des eaux.

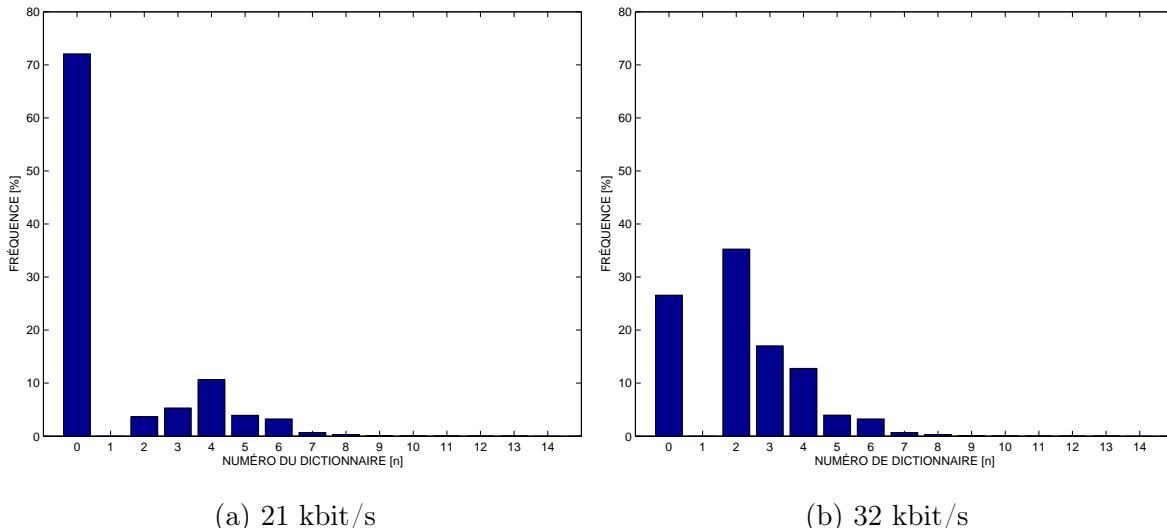


Figure 7.11 Fréquence d’utilisation des dictionnaires de la quantification.

La Figure 7.13 donne pour une trame donné l’erreur de codage après et avant quantification. On s’aperçoit que le codage d’amélioration couvre un large spectre des fréquences à 32 kbit/s. L’erreur dans le domaine perceptuel, est alors quasi uniforme sur tout le spectre, sauf entre 7 et 8 kHz, bande dans laquelle le signal n’est pas codé. Par contre à 21 kbit/s, c’est essentiellement la bande basse qui est décrite.

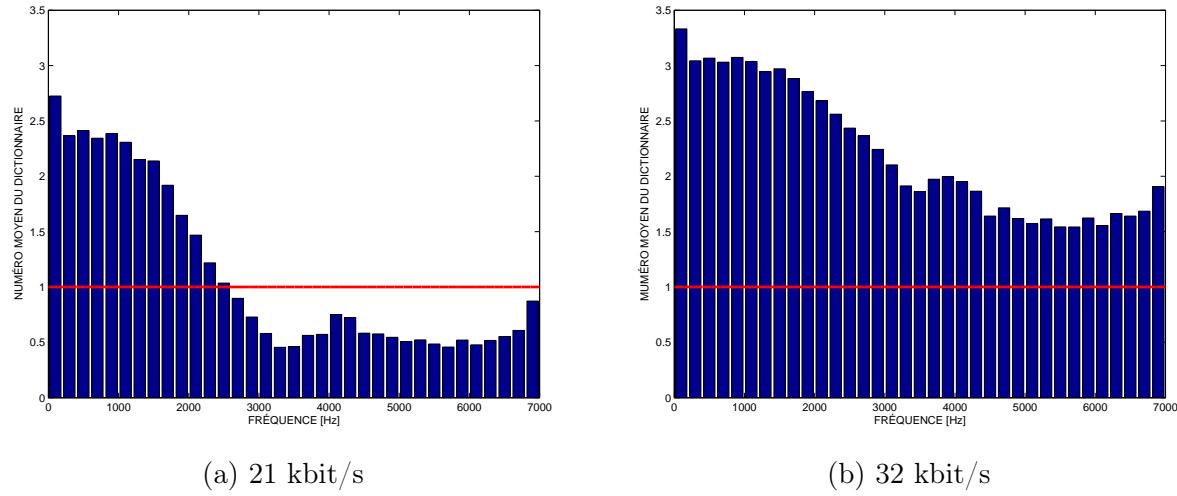


Figure 7.12 Numéro du dictionnaire moyen utilisé par chaque vecteur de dimension 8.

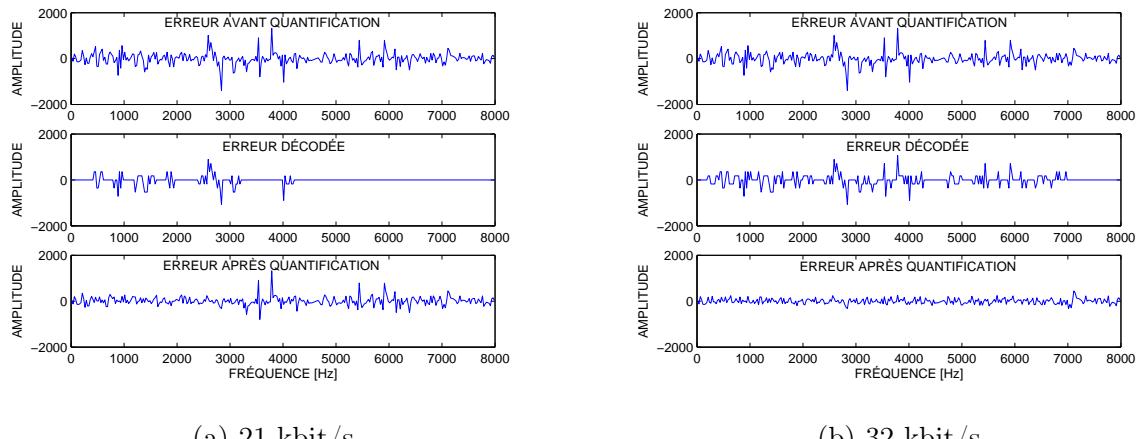


Figure 7.13 Erreur de codage avant et après quantification de la cible.

7.2.4 Multiplexage codage/estimation

Connaissant au décodage les numéros de dictionnaire utilisés pour chaque vecteur décodé, il est facile de recouvrir la position des vecteurs qui n'ont pas été décodés dans la bande manquante. Cette information sert alors à commuter pour chaque vecteur entre l'estimation \hat{X}_{est} ou bien la valeur codée \hat{X}_{cod} . La Figure 7.14 explicite le multiplexage fréquentiel des deux signaux \hat{X}_{est} et \hat{X}_{cod} pour former le signal composite de la bande haute \hat{X}_{HF} .

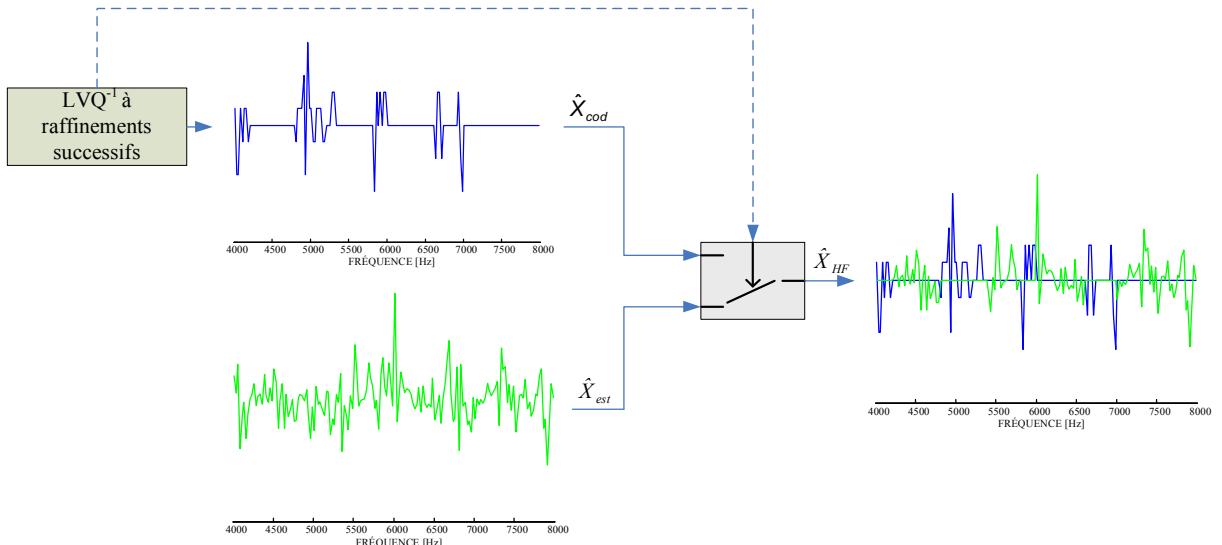


Figure 7.14 Multiplexage fréquentiel entre le signal estimé et codé.

7.3 Améliorations et compromis

La structure de codage hiérarchique présentée dans ce chapitre permet de nombreux ajustements et améliorations selon l'application visée. On va par la suite discuter de différentes améliorations qui peuvent améliorer la qualité pour certains types de signaux, mais aussi la détériorer pour d'autres. Nous discuterons alors des compromis pouvant être faits.

7.3.1 Ordonnancement des vecteurs codés

L'ordre de transmission des descriptions des vecteurs est comme nous l'avons aperçu dans le chapitre 5 primordial pour optimiser pour chaque débit de décodage la qualité de la synthèse. Si on se base sur le critère de l'erreur quadratique, on devrait coder et transmettre les différents

vecteurs du spectre selon le principe du remplissage inverse des eaux. L'utilisation d'un modèle perceptuel, nous a permis de nous ramener dans un tel domaine où l'erreur quadratique de la cible à coder est à minimiser. Cette simplification du problème ne prend pas en compte d'autres considérations pouvant être très importantes pour la qualité perceptuelle. Deux considérations semblent ainsi essentielles à la qualité de la synthèse :

- Le codage de coefficients transformés juxtaposés permet de définir une partie de l'enveloppe temporelle du signal, et ainsi d'obtenir une cohérence temporelle.
- Une cassure de la structure harmonique lors du multiplexage dans la bande manquante peut nuire à la qualité de la synthèse. Cela arrive chaque fois qu'on passe d'un vecteur codé à un vecteur estimé et vice versa.

On peut déduire de ces deux considérations que la transmission progressive définie par le principe du remplissage inverse des eaux, n'est pas forcément la transmission la plus optimale selon la qualité perceptuelle. Par exemple, une transmission graduelle vecteur par vecteur selon l'ordre des fréquences, prend plus en compte les considérations énumérées ci-dessus. Selon des tests informels, la transmission vecteur par vecteur est plus favorable pour la quantification de la bande manquante. Par contre, elle moins avantageuse pour la quantification de la bande de base.

7.3.2 Allocation binaire fixe entre la bande basse et la bande haute

Dans le codeur hiérarchique présenté dans ce chapitre, l'allocation binaire entre la bande basse et la bande haute est adaptable. Elle dépend du modèle perceptuel utilisé, à savoir les filtres pondérateurs $W_{LF}(z)$ et $W_{HF}(z)$. Même si les basses fréquences sont plus favorisées, l'allocation entre les deux sous-bandes est assez équitable. Il est difficile de savoir si le choix est adéquat, car on ne tient pas compte de la qualité apportée par l'estimation de l'excitation de la bande manquante. Pour la parole, le repliement spectral de l'excitation est une estimation d'assez bonne qualité. L'allocation devrait alors être encore plus favorable à la bande basse. Par contre, lorsque l'excitation de la bande manquante est faiblement corrélée à la bande basse, comme c'est le cas pour certains sons musicaux, le repliement spectral est trop simpliste. Il faudrait donc privilégier les hautes fréquences.

On ne tient pas compte aussi de l'effet du multiplexage fréquentiel sur la qualité globale lors de l'allocation. La qualité de la synthèse peut baisser si on insère une ou plusieurs ruptures de la structure harmonique dans la bande manquante. Il est très difficile de connaître les incidences de

l'allocation sur la qualité de synthèse de la bande haute. Une approche conservatrice consiste alors à allouer les ressources du codage par transformée uniquement dans la bande basse. On simplifie le problème en utilisant seulement le codage paramétrique de la bande haute et en se passant ainsi du multiplexage. Il est possible dans ce cas, d'estimer l'excitation de la bande manquante après le codage par transformée et plus seulement après le codage de parole G729. La qualité de l'estimation s'en trouve améliorée.

Afin de comparer l'allocation adaptative entre la bande basse et haute et l'allocation uniquement dans la bande basse, nous avons utilisé l'extension large bande du PESQ, le WPESQ (*Wideband Perceptual Evaluation of Speech Quality*). Le WPESQ a été adopté par l'ITU-T par la recommandation P.862.2 [146]. Cette mesure objective permet d'évaluer des codeurs large bande pour de la parole. Il est reconnu qu'il est assez efficace et conforme aux tests subjectifs pour évaluer un même codeur dans différentes conditions. Par contre, la comparaison entre différents codeurs est très difficile et peu cohérente [147]. On l'utilise donc seulement pour avoir une idée approximative des performances des deux allocations. Nous avons utilisé pour le test une séquence de 2 minutes de phrases parlées en anglais et en français mélangeant voix d'homme et de femme. On a aussi inséré les performances de l'AMR-WB pour simple référence. Les résultats sont reportés à la Figure 7.15. On note dans le deux cas le gain significatif qu'apporte l'extension de bande entre les 8 kbit/s et le 10 kbit/s. Pour les débits supérieurs, l'allocation des ressources uniquement dans la bande basse sévère être plus efficace. L'allocation unilatérale semble tout de même moins adaptée pour les sons plus variés qu'on retrouve en musique.

7.3.3 Réduction de l'étalement spectral

L'étalement spectral est un problème sous-jacent du codage par transformée, surtout à bas débit. Nous avons vu dans le chapitre 6 que le fenêtrage adaptatif permet de résoudre une grande partie du problème, mais implique un délai supplémentaire. C'est pour cette raison que nous avons introduit le commutateur du domaine de codage dans la couche d'amélioration du codeur hiérarchique du chapitre 6. Malheureusement, la commutation du domaine fréquentiel au domaine temporel ne peut s'appliquer dans le cas présent du fait que le multiplexage entre les coefficients estimés et codés de la bande haute se fait exclusivement dans le domaine fréquentiel. La mise en forme temporelle du bruit de codage (TNS) est alors la solution la plus appropriée [58]. Dans notre cas elle s'adapte particulièrement bien à la décomposition en sous-bandes, car elle permet

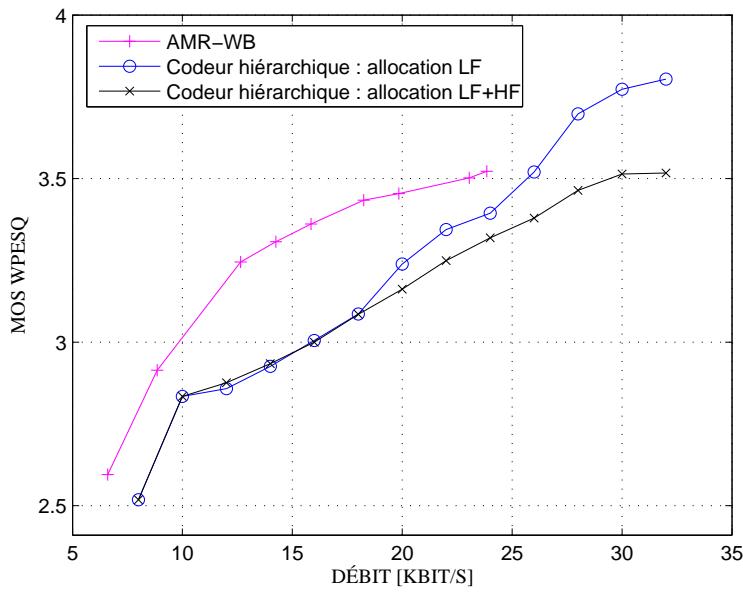


Figure 7.15 Performances WPESQ du codage hiérarchique pour différents types d’allocation.

de faire hériter facilement l’enveloppe de Hilbert de la bande basse à la bande haute. On fait l’hypothèse que l’enveloppe temporelle de la bande basse et celle de la bande haute sont fortement corrélées surtout lors des transitoires. Si c’est le cas, il est possible d’utiliser les coefficients de la prédiction fréquentielle de la synthèse bande étroite du codeur de parole pour mettre en forme l’erreur du codage par transformée que ce soit dans la bande basse ou dans la bande haute. Afin d’éviter tout effet néfaste du repliement temporel de la MDCT, il est souhaitable d’utiliser lors de l’activation du TNS, une fenêtre ayant peu de recouvrement. Comme le gain de codage baisse lors de l’utilisation d’une telle fenêtre, il est préférable d’utiliser le TNS seulement dans le cas où le signal présente une forte transitoire.

La Figure 7.16 résume le procédé du TNS au sein de la structure hiérarchique. La prédiction linéaire (analyse LP) est calculée au codeur et au décodeur sur le spectre \hat{X}_{core} de la synthèse du codeur de parole. Aucune information ne nécessite d’être transmise. On suppose que l’enveloppe temporelle de la bande haute est fortement corrélée à celle de la bande basse. Ainsi, l’erreur de la bande basse et de la bande haute sont toutes les deux mises en forme par le filtre de synthèse $1/A_{TNS}(z)$ issu de la prédiction. On utilise un ordre de 8 pour s’assurer de bien suivre l’enveloppe temporelle tout en évitant de modéliser les impulsions glottales. Le multiplexage fréquentiel entre les composantes codées et estimées se fait dans le domaine du résidu de la prédiction. L’erreur

résultante $x_{HF}(n) - \hat{x}_{HF}(n)$ est ainsi mise en forme temporellement. Les Figures 7.17 et 7.18 permettent de comparer dans le cas d'une attaque de castagnettes la mise en forme temporelle de l'erreur sans et avec le TNS respectivement. L'attaque est beaucoup mieux définie avec le TNS. La Figure 7.19 montre dans cet exemple que l'enveloppe temporelle issue du filtre de synthèse calculé sur le spectre X_{HF} est fortement corrélée avec celle de $1/A_{TNS}(z)$, ce qui conforte notre hypothèse préliminaire.

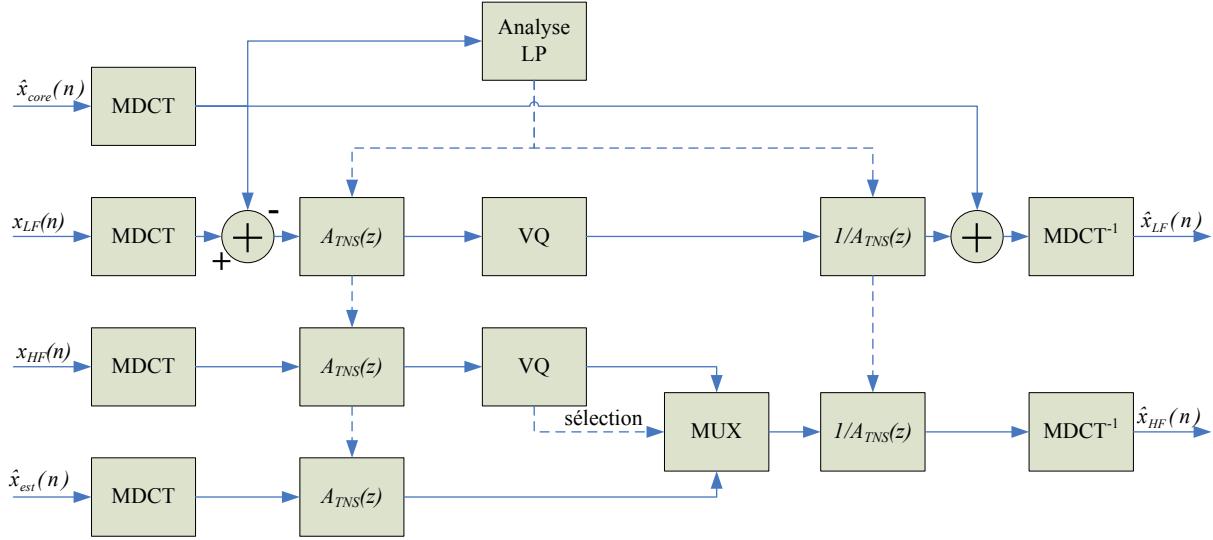


Figure 7.16 Prédiction linéaire fréquentielle pour mettre en forme temporellement l'erreur du codage par transformée.

7.3.4 Postfiltrages de la synthèse

Le postfiltrage est un procédé courant et efficace pour le codage de la parole à bas débit. Il permet d'augmenter la qualité perçue de la synthèse en creusant les vallées du spectre, là où l'erreur est susceptible d'être la plus importante [104]. Il peut être alors utilisé dans la bande basse du codage hiérarchique afin d'améliorer la qualité de la synthèse du codeur de parole. Il met en jeu deux filtres, un à court terme et l'autre à long terme.

Le filtre à court terme permet d'atténuer les vallées entre les formants du spectre. Il est dérivé du filtre de synthèse de la prédiction linéaire :

$$H_{LFst}(z) = [1 - \mu z^{-1}] \frac{\hat{A}_{LF}(z/\beta_{LF})}{\hat{A}_{LF}(z/\alpha_{LF})}, \quad 0 < \beta_{LF} < \alpha_{LF} < 1 \quad (7.3)$$

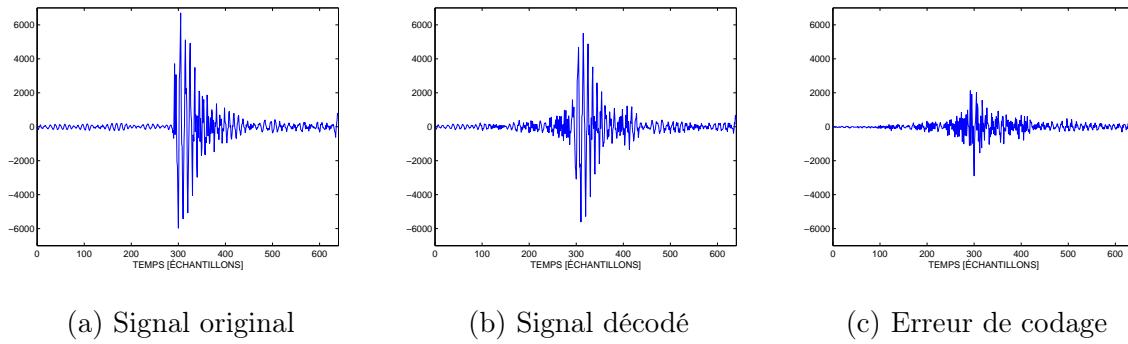


Figure 7.17 Codage sans mise en forme temporelle.

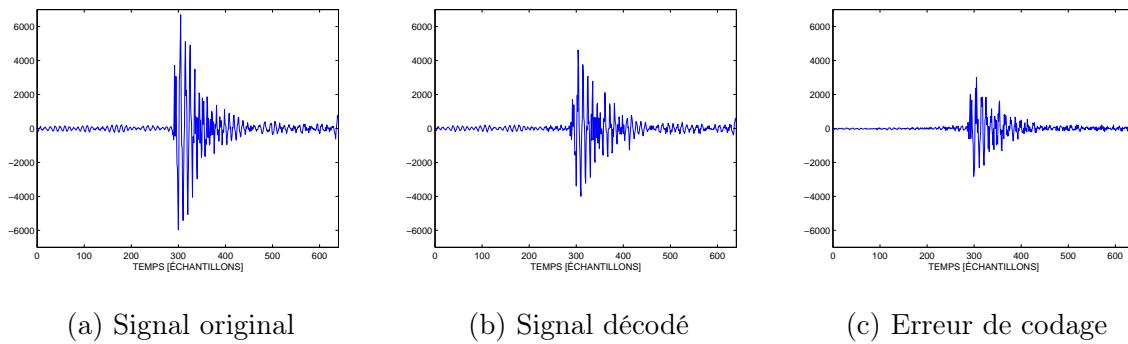


Figure 7.18 Codage avec mise en forme temporelle.

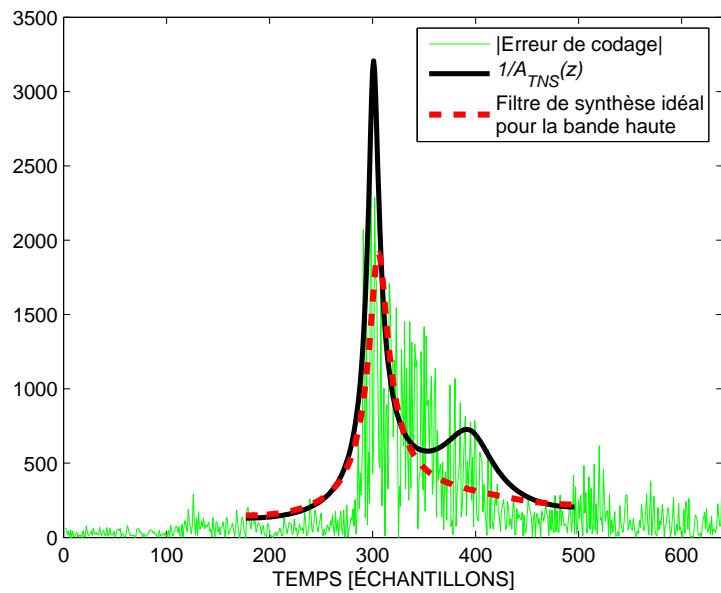


Figure 7.19 Enveloppe temporelle issue du filtre de synthèse de la prédiction fréquentielle.

Le filtre de premier ordre $[1 - \mu z^{-1}]$ permet d'annuler la pente (*tilt*) du filtre $H_{LFst}(z)$. Le coefficient μ peut être trouvé à l'aide du coefficient de réflexion de premier ordre de la réponse impulsionale de $\frac{\hat{A}_{LF}(z/\beta_{LF})}{\hat{A}_{LF}(z/\alpha_{LF})}$. Les coefficients α_{LF} et β_{LF} sont fixés respectivement à 0.7 et 0.55 pour notre codeur hiérarchique.

Le filtre à long terme permet d'atténuer le spectre entre les pics harmoniques de la fréquence fondamentale (fréquence de *pitch*). Il s'exprime alors de la façon suivante :

$$H_{LFlt}(z) = G_{LF} \frac{1 + \gamma_{LF} z^{-T}}{1 - \lambda_{LF} z^{-T}} \quad (7.4)$$

Le gain G_{LF} , le délai T , ainsi que les paramètres γ_{LF} et λ_{LF} sont adaptatifs et dépendent du voisement de la trame présente.

Comme le postfiltrage est très efficace pour les codeurs de parole, il serait intéressant d'utiliser le même principe pour le codage paramétrique de la bande haute afin d'améliorer l'estimation de l'excitation. C'est pour cette raison que nous avons introduit un nouveau postfiltrage dans [148] pour les extensions de bande hautement paramétriques, et plus particulièrement pour les extensions estimant la structure fine par un repliement spectral. Il permet d'améliorer les propriétés de l'excitation répliquée. Comme le postfiltage présenté ci-dessus, il met aussi en jeu des filtres à court et à long terme.

Le filtre à court terme permet de blanchir l'excitation venant du codage parole. En effet, les codeurs parole ont tendance à favoriser les formants du spectre ainsi que les basses fréquences. L'excitation générée est alors plus énergétique dans ces régions du spectre. Le filtre à court terme est ainsi déduit du filtre de synthèse de la bande basse :

$$H_{HFst}(z) = \frac{\hat{A}_{LF}(z/\beta_{HF})}{\hat{A}_{LF}(z/\alpha_{HF})}, \quad 0 < \alpha_{HF} < \beta_{HF} < 1 \quad (7.5)$$

β_{HF} et α_{HF} sont fixés expérimentalement à 1 et 0.3 respectivement.

Le filtre à long terme permet de réduire la surestimation du degré de voisement de l'excitation répliquée après le repliement spectral. En effet, les très basses fréquences se retrouvent par repliement spectral dans les très hautes fréquences du spectre. Généralement, le taux de voisement est très différent dans ces deux bandes de fréquence. La structure harmonique est alors surestimée dans les hautes fréquences de la bande haute. Le filtre à long terme est appliqué à l'excitation de la bande basse avant réplication dans la bande haute. Pour la bande haute dans une structure en sous-bandes, les fréquences sont inversées à cause du sous-échantillonnage du signal. De ce fait,

le filtre est alors la concaténation d'un filtre antiharmonique dans la première moitié de la bande et d'un filtre passe-tout dans la seconde :

$$H_{HFlt}(z) = \frac{1}{2}(1+z^{-1})(1-\lambda_{HF}z^{-T}) + \frac{1}{2}(1-z^{-1}) \quad (7.6)$$

$$= 1 - \frac{\lambda_{HF}}{2}z^{-T} - \frac{\lambda_{HF}}{2}z^{-(T+1)} \quad (7.7)$$

le délai T et le paramètre λ_{HF} sont adaptatifs et dépendent du voisement de la trame présente.

Les deux postfiltrages de la bande basse et de la bande haute sont plutôt adaptés pour le traitement de la parole. Ils ont une incidence significative sur la qualité surtout à bas débit, pour la première synthèse large bande obtenue à 10 kbit/s.

7.3.5 Autres améliorations potentielles

Il est possible d'apporter d'autres améliorations au codeur hiérarchique. On peut citer par l'exemple l'ajout d'un dictionnaire innovateur au codeur G729 avant le codage par transformée. On a alors un codage CELP imbriqué comme il en a été présenté à la section 3.1.1. On peut aussi rajouter dans la bande basse, le posttraitement fréquentiel du chapitre 4 comme il a été fait au chapitre 5 entre le codage de parole et le codage par transformée. D'autres améliorations peuvent être apportées et faire d'études futures. En particulier, il serait intéressant d'améliorer l'estimation de l'excitation pour l'adapter davantage au traitement de la musique.

7.4 Tests subjectifs

Nous avons conduit un test subjectif afin d'évaluer notre codeur hiérarchique avec extension de bande. Nous avons utilisé certaines des améliorations présentées dans la section précédente pour aboutir à un certain compromis. D'abord, l'allocation des ressources du codage par transformée se fait uniquement dans la bande basse. On simplifie alors le codage par transformée, et on évite ainsi les artefacts indésirables dus au multiplexage des coefficients codés et estimés. La bande haute est donc uniquement codée par le codage paramétrique, et l'excitation est entièrement estimée pour tous débits de décodage. Dans la bande basse, on utilise le posttraitement fréquentiel ainsi que le TNS pour le codage par transformée. Enfin, les deux types de postfiltrage présentés ont été utilisés, dans la bande basse et dans la bande haute.

L'évaluation des performances a été réalisée par un test *Mushra*. Le test comprend 4 séquences de parole de voix de femme et d'homme en anglais, 4 séquences de divers styles musicaux et 4 séquences de parole mixée à de la musique ou du bruit de fond. Il y a deux configurations différentes selon la nature du signal à tester. Pour la parole, on compare le codeur hiérarchique à 10, 24 et 32 kbit/s avec l'AMR-WB à 8.85 et 23.05 kbit/s. Pour la musique et le mélange parole-musique, on compare le codeur hiérarchique à 24 et 32 kbit/s avec le G722 à 48 et 54 kbit/s, le G722.1 à 24 kbit/s et enfin l'AMR-WB à 23.05 kbit/s. 10 auditeurs entraînés ont pris part au test. Les résultats du test sont reportés à la Figure 7.20.

Pour la parole (Figure 7.20 (a)), le codeur hiérarchique à 10 kbit/s, c.-à-d.avec seulement l'extension de bande paramétrique et sans codage par transformée, est meilleur que l'AMR-WB à 8.85 kbit/s, un codeur de parole large bande non hiérarchique. L'extension paramétrique est efficace pour ce genre de signaux. À 32 kbit/s et même à 24 kbit/s, le codeur hiérarchique surpassé l'AMR-WB à 23.05 kbit/s. Cela démontre que l'allocation unilatérale du codage par transformée et l'ordonnancement des vecteurs codés sont adaptés au signal de parole.

Pour la musique (Figure 7.20 (b)), les performances du codeur sont plus en retrait. On se rapproche à 32 kbit/s des performances du G722 à 48 kbit/s. La synthèse a une légère signature auditive due à l'extension paramétrique de la bande qui est surtout adaptée pour la parole. L'allocation des ressources uniquement dans la bande basse n'est plus aussi efficace. À 24 kbit/s, les artefacts sont plus prononcés et la qualité auditive baisse en conséquence. Pour le mélange parole-musique (Figure 7.20 (c)), le codeur hiérarchique à 32 kbit/s a une bonne qualité au dessus du G722.1 et de l'AMR-WB, mais à 24 kbit/s la qualité baisse encore. Les artefacts sur les sons musicaux handicapent le codeur à ce débit.

Le test subjectif met en évidence que les choix de configuration font que le codeur hiérarchique à extension de bande est dans le cas présent très efficace pour de la parole quelque soit le débit de décodage. Par contre, il s'adapte plus difficilement à la musique. La qualité est bonne à 32 kbit/s, mais ne se maintient pas à 24 kbit/s.

7.5 Conclusion

Nous venons de présenter une structure hiérarchique à base d'un codeur parole bande étroite permettant par une extension de la bande transmise d'obtenir une synthèse large bande. Par

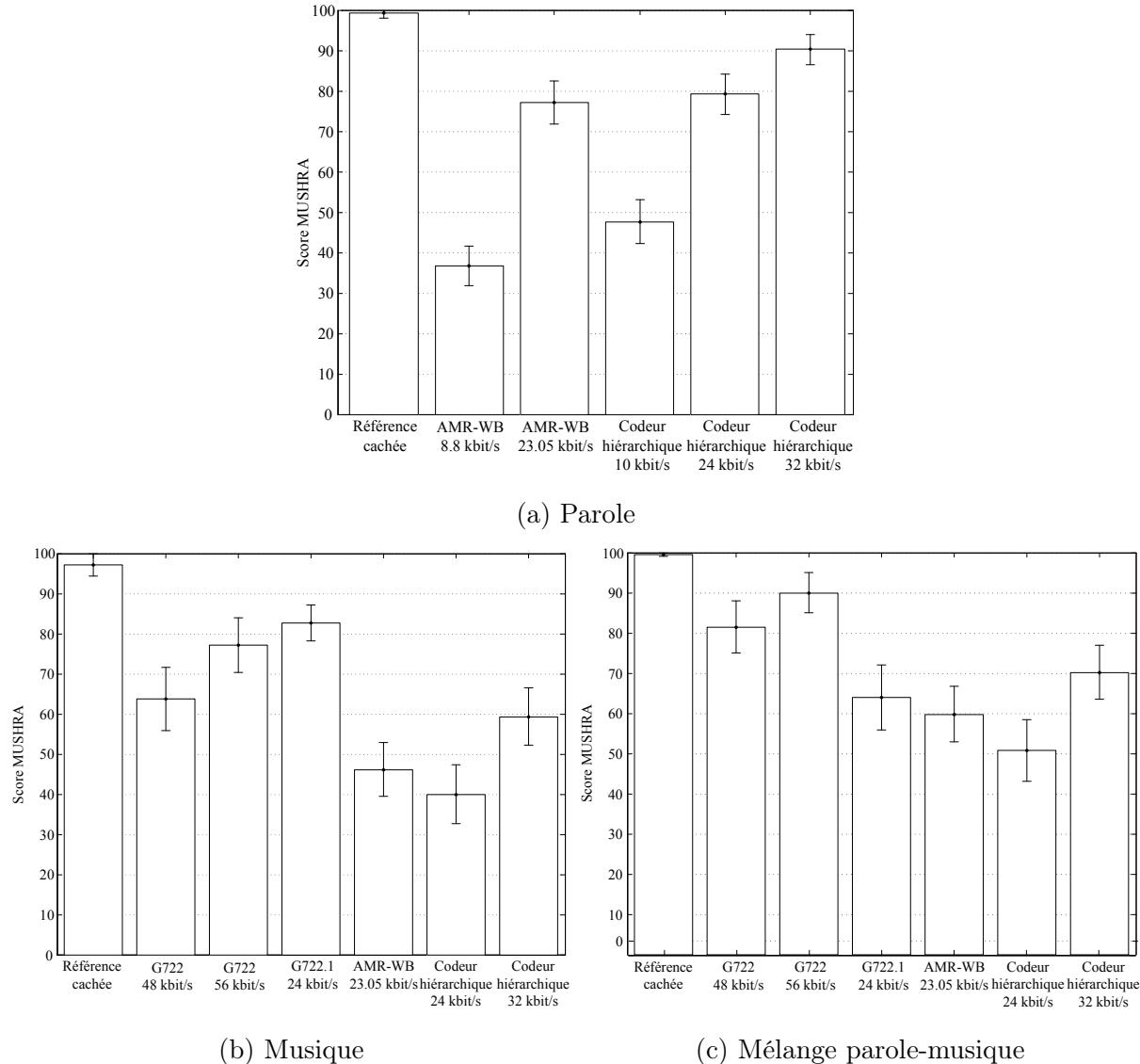


Figure 7.20 Résultats du test MUSHRA avec intervalles de confiance à 95%.

rapport à un codage d'amélioration graduel uniquement dans la bande de base comme présenté au chapitre 6, les choix scientifiques et technologiques sont plus difficiles.

Nous avons opté pour extension multiplexant dans le domaine fréquentiel des composantes estimées et des composantes explicitement codées. Cette stratégie permet de jouir d'une synthèse large bande dès un débit de 10 kbit/s. Par contre, elle complexifie l'optimisation de l'allocation des ressources. L'utilisation d'un modèle perceptuel consistant à se ramener dans un domaine où l'erreur quadratique est à minimiser, est trop simpliste et ne suffit pas pour tenir en compte tous les principaux critères perceptuels. En effet, il est essentiel de ne pas créer d'importantes ruptures de la structure harmonique dans la bande haute lors du multiplexage des coefficients estimés et codés. De plus, le codage parcimonieux dans le domaine spectral peut provoquer des artefacts temporels assez gênants. Il est ainsi préférable de coder des vecteurs se juxtaposant.

À partir de ces constatations, nous avons proposé une série d'améliorations, et présenté les compromis associés. Une amélioration peut être bénéfique pour un type de signal, mais en même temps pénalisante pour les autres. Le choix de son incorporation au sein du codeur dépendra de l'application visée. Nous avons présenté en particulier une configuration du codeur hiérarchique qui privilégie plus la qualité de la parole que celle de la musique. D'autres configurations, privilégiant la musique ou idéalement étant plus universelles, pourraient être intéressantes à étudier plus en profondeur.

CONCLUSION

L'objet de la thèse était d'étudier le codage hiérarchique à bas débit afin de proposer des solutions viables et fiables pour des communications sur des réseaux hétérogènes. Nous avons dans l'introduction fait état de la nécessité du codage audio hiérarchique dans le contexte actuel des télécommunications. Nous avons aussi constaté le manque de normalisation pour ce type de codage. Principalement, nous souhaitions nous atteler simultanément à deux problématiques.

La première est de pouvoir adapter le codage à des signaux audio de source très hétérogène. On souhaitait en particulier transmettre de la parole avec une très bonne qualité, mais aussi de la musique, ou du matériel sonore plus complexe, dans le cas où le débit disponible au décodage est plus élevé. Dans le chapitre 2, nous avons différencié deux paradigmes en codage audio : le codage de la parole et le codage audio générique. Les codeurs de parole se basent généralement sur un modèle de production de la voix. Ils utilisent un modèle source-filtre, en transmettant les coefficients d'un filtre de synthèse et en modélisant une excitation le plus souvent dans le domaine temporel. Les codeurs de parole sont surtout dédiés pour transmettre de la parole à faibles débits. De l'autre côté, les codeurs audio génériques font abstraction de l'origine du signal. Ils transforment le signal dans le domaine fréquentiel où ils utilisent certaines caractéristiques de l'audition humaine, comme le masquage, pour augmenter le taux de compression. Ils font seulement des suppositions d'ordre général sur les statistiques du signal d'entrée. Ils ont des débits nominaux plus élevés que les codeurs de parole, mais peuvent traiter convenablement tous types de signaux audio.

La deuxième problématique est de pouvoir adapter le codage à des débits de transmission variables dans le temps et dans l'espace. Le train binaire encastré généré par le codage hiérarchique peut d'autant plus s'adapter aux diverses conditions des réseaux qu'il peut être décodé graduellement avec une très fine granularité. Le chapitre 3 fait l'état de l'art des codeurs hiérarchiques à faibles débits construits le plus souvent autour d'un codeur de parole. On distingue d'abord les codeurs CELP imbriqués. Ils affinent la description de l'excitation en ajoutant de nouveaux dictionnaires innovateurs. Néanmoins, le codage hiérarchique résultant reste toujours un modèle source-filtre propre à la production de la parole. Pour obtenir un codage plus polyvalent, il est alors plus intéressant d'associer un codage CELP avec un codage par transformée. Les deux paradigmes du

codage ainsi associés peuvent alors se compléter et viser une gamme de débits plus importants. Il s'avère tout de même que pour le codage de la musique, il est nécessaire d'allouer un débit assez important pour combler les défaillances du codage de parole. Dans le cas d'une extension de bande, le fort débit exigé dans la bande de fréquence codée par le codeur de parole, peut être compensé par une extension artificielle ou bien hautement paramétrique.

Nous avons proposé dans la thèse plusieurs outils et solutions de codage afin de répondre au mieux à ces deux problèmes.

Contributions de la thèse

Le chapitre 4 propose d'étudier et d'améliorer la qualité de la synthèse du codeur de parole lors du traitement de signaux autres que la parole. On a tout d'abord caractérisé plusieurs artefacts venant d'un tel codeur. Le plus important est que l'énergie entre les pics spectraux de la synthèse est souvent trop élevée surtout lorsque le signal ne répond pas au modèle de source utilisé par le codeur. On a alors introduit un posttraitement dans le domaine fréquentiel. Ce posttraitement permet de creuser les vallées du spectre par une simple mise à zéro forcée de certaines composantes spectrales. La mise à zéro est guidée par une comparaison au moment du codage du spectre original avec le spectre de la synthèse du codeur de parole. Une information supplémentaire est alors ajoutée à l'information initiale pour former un train binaire encastré. L'amélioration est très significative pour la musique tout en ne pénalisant pas les performances initiales pour la parole. Le posttraitement est une première couche d'amélioration d'un codage hiérarchique, qui, pour certaines applications, peut suffire à elle-même pour rendre plus polyvalent un codeur de parole. Le posttraitement a été décrit au préalable dans [149].

Le chapitre 5 propose une solution beaucoup plus générique pour le codage hiérarchique, une quantification à raffinements successifs. Une telle quantification permet d'obtenir un décodage graduel du signal codé. Le raffinement graduel est obtenu grâce à l'utilisation de réseaux réguliers de points comme dictionnaires. L'exploitation des propriétés remarquables des réseaux permet de décomposer un vecteur codé en plusieurs code-vecteurs. La décomposition provient de l'extension de Voronoï graduelle dérivée de l'extension de Voronoï introduite dans [133]. Elle permet de former un dictionnaire ayant des codes imbriqués venant d'un même réseau de points dilaté à des facteurs différents. Elle peut être vue comme une généralisation en plusieurs dimensions du codage

par plan de bits issu d'une quantification scalaire. La quantification par extension de Voronoï graduelle tire aussi des avantages de ses dictionnaires définis algébriquement. La recherche du plus proche voisin est très peu complexe et le stockage des dictionnaires est négligeable. La définition de dictionnaires de base adaptés à la source permet d'adapter la quantification par extension de Voronoï graduelle vectorielle à de nombreuses sources et donc à de nombreuses applications. De plus, les performances de la quantification lors de la troncature de train binaire généré, sont à débit équivalent très proches de celles d'un codage à débit fixe.

Le chapitre 6 propose une solution complète d'un codage hiérarchique allant de 12.65 kbit/s à 24 kbit/s à base du codeur de parole large bande AMR-WB. La structure de codage, basée sur un modèle classique, imbrique le codage parole avec un codage par transformée. Les coefficients sont codés via une quantification par extension de Voronoï graduelle. Un procédé de masquage ainsi qu'une mise forme du bruit de codage, jouent l'interface entre les deux codages afin d'optimiser leur coopération. On met en évidence à travers un test subjectif deux problèmes majeurs liés à la structure hiérarchique : la difficulté à pallier aux défauts inhérents du codeur de parole lors du traitement de sons musicaux, et l'étalement spectral introduit par le codage par transformée. On adjoint alors deux optimisations perceptuelles au codage hiérarchique. Le posttraitement fréquentiel pour améliorer la qualité de synthèse de la musique, ainsi qu'un commutateur de domaine de codage pour atténuer le préécho. À débit équivalent, les performances du codage hiérarchique sont alors très proches de celles des codeurs classiques non hiérarchiques, optimisés pour un débit fixe et un seul type de signal. Un partie du codeur a été décrit dans [150].

Le dernier chapitre se penche sur l'extension de la bande transmise. À partir du codeur de parole bande étroite G.729 à 8 kbit/s, le codage hiérarchique améliore la bande de fréquence préalablement codée tout en augmentant le contenu fréquentiel afin d'obtenir une synthèse large bande. Une structure en sous-bande décompose le signal d'entrée en deux bandes : une bande haute et une bande basse. La bande basse correspond à la bande de base déjà transmise par le codeur de parole et la bande haute, la bande manquante à la synthèse. L'utilisation d'un codage paramétrique de la bande haute permet dès les 10 kbit/s d'obtenir une synthèse large bande. Le codage nécessite au décodeur d'estimer la structure fine du spectre. Pour des débits supérieurs à 10 kbit/s, les composantes estimées sont remplacées par une description explicite venant d'un codage par transformée. Ce même codage par transformée permet aussi de raffiner la synthèse du codeur de parole dans la bande de base. Le codage par transformée a un débit de codage de 22 kbit/s,

ce qui fait un total de 32 kbit/s pour le codeur hiérarchique. L'allocation des ressources entre les deux bandes pour le codage par transformée influence grandement les performances du codage. Un simple modèle perceptuel ne suffit généralement pas pour considérer toutes les caractéristiques psychoacoustiques de l'oreille humaine. Il faut alors faire un choix préalable qui peut favoriser comme défavoriser un type de signal. Nous avons apporté des améliorations au codeur avec un compromis plutôt favorable à la parole. Pour ce type de signal, le codeur réagit, à débit équivalent, tout aussi bien voire mieux, que les codeurs de parole standards non hiérarchiques. Par contre, pour la musique les performances sont en retrait, surtout lorsque le train binaire est tronqué. Il serait nécessaire dans de futurs travaux d'étudier de nouvelles améliorations plus avantageuses pour la musique. Ce codeur, optimisé différemment, a été utilisé comme candidat par la société Voiceage pour la norme ITU-T G.729.1. La solution a été parmi l'une de celle sélectionnée pour réaliser le codeur commun G.729.1. De plus, l'extension de bande du codeur a fait l'objet de la publication [148].

Perspectives pour des travaux ultérieurs

Au cours de la thèse, nous avons proposé des solutions de codage hiérarchique avec des performances consistantes. Néanmoins, le maintien d'une bonne qualité pour la synthèse de signaux de musique s'est avéré plus délicat. À partir des structures hiérarchiques présentées, il serait intéressant d'explorer des améliorations supplémentaires pour augmenter la qualité de la musique. Comme les signaux musicaux sont très divers, il est difficile de répondre à l'ensemble des artefacts sans en créer des nouveaux. Un codage adaptatif pourrait minimiser les compromis à faire en optimisant certains paramètres de codage selon le signal d'entrée. D'un autre côté, des voies de recherche plus ambitieuses et innovatrices seraient plus enclines à apporter des gains significatifs. Plusieurs axes de recherche semblent prometteurs au niveau du codage audio hiérarchique.

- *Le codage spatial* : Il a été évoqué du chapitre d'introduction que les couches d'amélioration puissent ajouter des canaux supplémentaires à la synthèse. Le codage d'une image stéréo voire de plusieurs canaux serait un ajout significatif au codage hiérarchique. Le codage multicanal paramétrique en prenant en compte des caractéristiques binaurales fait l'objet actuellement de nombreuses recherches [151]. Une extension des codeurs MPEG pour obtenir un codage spatial est en cours de normalisation [152].

- *Le codage par objets* : Le codage par objets est une approche prometteuse qui va au-delà du simple codage paramétrique. Il consiste à décomposer en objets audio un signal sonore. Une représentation appropriée est alors utilisée pour chaque objet. Certains codages existants, comme le codage sinusoïdal, décomposent déjà le signal en plusieurs composantes élémentaires comme des sinusoïdes, des transitoires ou du bruit [153]. Des objets plus complexes pourraient être aussi rajoutés. L'approche devrait intéressante pour le codage hiérarchique du fait que les objets selon leur importance et le débit disponible, peuvent être soit codés, estimés ou bien simplement négligés. De plus, une bonne modélisation par objets serait répondre au problème du codage universel à bas débit.
- *Une décomposition du signal adaptée* : Cette voie rejoint le codage par objets. Si on considère qu'un son est formé de plusieurs objets sonores mélangés dans le temps et dans les fréquences, il faut choisir la bonne décomposition pour sélectionner les bons objets à modéliser et en extraire les caractéristiques essentielles. Par exemple, un banc de filtres adaptatif permet d'obtenir un meilleur gain de codage ou d'obtenir une erreur de codage moins perceptible. Le bon choix de la décomposition permet d'obtenir le signal transformé le plus adéquat pour le codage. Le fenêtrage adaptatif et le commutateur du domaine de codage, présenté au chapitre 5, sont ainsi des cas particuliers d'un codage par banc de filtres adaptatif, alors que [56] propose une solution plus générale à l'aide de paquets d'ondelettes et d'une MDCT.
- *Modèle perceptuel plus réaliste* : La psycho-acoustique est une matière complexe et encore trop imprécise pour comprendre en totalité la perception auditive humaine. Il est en plus difficile de l'intégrer facilement dans des solutions techniques. Ainsi, un meilleur modèle perceptuel permettrait d'obtenir des codeurs plus performants. L'allocation binaire ainsi que la hiérarchisation de l'information seraient plus conformes à la perception auditive.

ANNEXE A

Modélisation par un processus de Markov

Un processus de Markov Z du premier ordre se résume à la Figure A.1 [102]. Le modèle est défini par une chaîne de deux états $TRUE$ et $FALSE$, ainsi que par deux probabilités de transition d'état t_{TRUE} et t_{FALSE} .

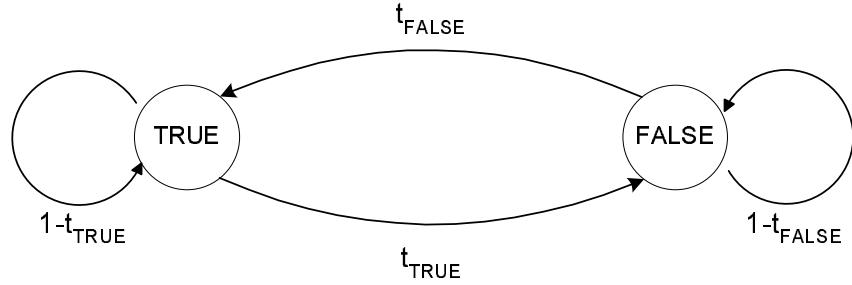


Figure A.1 Processus de Markov du premier ordre.

On modélise notre masque M par un tel processus, en assignant l'état $TRUE$ aux 1s et l'état $FALSE$ aux 0s. Il est alors possible d'en déduire la probabilité des longueurs de plage β_k pour une telle modélisation :

$$P(\beta_k|S) = t_S(1 - t_S)^{\beta_k - 1} \quad \text{avec} \quad \beta_k = 1, 2, \dots, \infty \quad \text{et} \quad S = \{\text{TRUE}, \text{FALSE}\} \quad (\text{A.1})$$

Maintenant calculons l'espérance des longueurs β_k :

$$E[\beta|S] = \sum_{\beta_k=1}^{\infty} \beta_k t_S(1 - t_S)^{\beta_k - 1} = -t_S \frac{d}{dt_S} \left\{ \sum_{\beta_k=1}^{\infty} (1 - t_S)^{\beta_k} \right\} \quad (\text{A.2})$$

$$= -t_S \frac{d}{dt_S} \frac{1}{1 - (1 - t_S)} = \frac{1}{t_S} \quad (\text{A.3})$$

On peut alors réécrire la probabilité $P(\beta_k|S)$ de la façon suivante :

$$P(\beta_k|S) = \frac{1}{E[\beta|S] - 1} \left[1 - (E[\beta|S])^{-1} \right]^{\beta_k} \quad (\text{A.4})$$

La distribution résultante du processus de Markov du premier ordre est donc géométrique. Il est possible de la comparer avec celle mesurée pour le masque M . La Figure A.2 prend comme exemple le masque issu du post-traitement du signal d'orgue. On s'aperçoit que la distribution mesurée suit approximativement la distribution d'un processus de Markov. Il est donc envisageable de modéliser les plages du masque M par un processus de Markov de premier ordre.

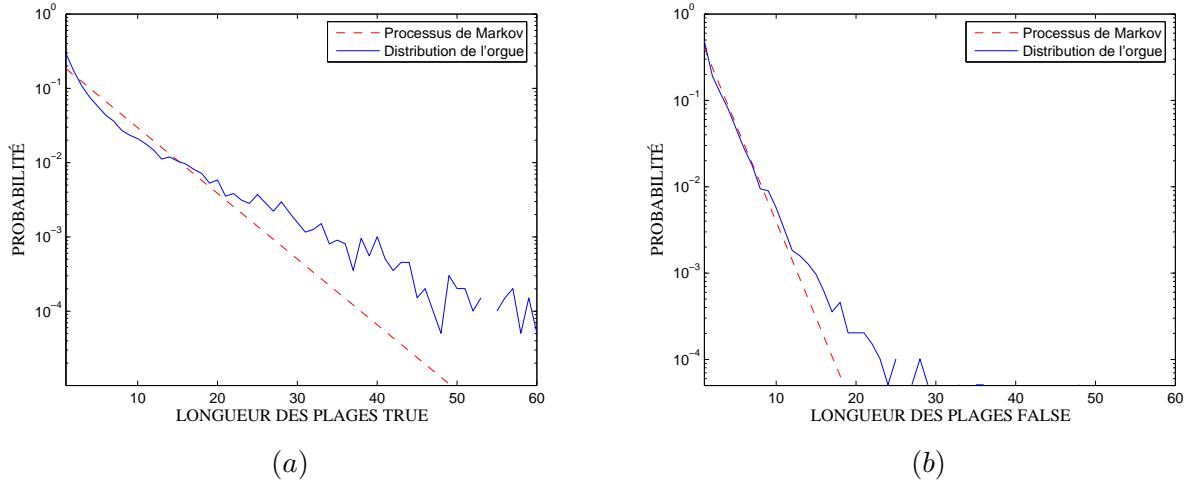


Figure A.2 Distribution mesurée versus distribution issue du processus de Markov.

L'entropie du processus de Markov Z est facilement calculable à l'aide de la formule suivante :

$$H(Z) = P(\text{TRUE})H(Z|S = \text{TRUE}) + P(\text{FALSE})H(Z|S = \text{FALSE}) \quad (\text{A.5})$$

avec

$$\begin{aligned} H(Z|S = \text{TRUE}) &= -(t_{\text{TRUE}}) \log_2(t_{\text{TRUE}}) \\ &\quad -(1 - t_{\text{TRUE}}) \log_2(1 - t_{\text{TRUE}}) \end{aligned} \quad (\text{A.6})$$

$$\begin{aligned} H(Z|S = \text{FALSE}) &= -(t_{\text{FALSE}}) \log_2(t_{\text{FALSE}}) \\ &\quad -(1 - t_{\text{FALSE}}) \log_2(1 - t_{\text{FALSE}}) \end{aligned} \quad (\text{A.7})$$

On obtient alors facilement les contours d'équi-entropie $H(Z)$ (iso-contours) en fonction de l'espérance des plages de 1s et de 0s. On observe que la source la moins redondante, c'est à dire ayant une entropie $H(Z) = 1$, a une espérance pour les deux plages égale à 2. La probabilité de transition d'un état à l'autre est alors de 0.5. La source est dans ce cas incompressible. Par contre si l'espérance des plages augmente, alors la compression devient de plus en plus performante. Il est possible de démontrer que pour une source quelconque ayant les mêmes espérances de plages, le modèle de Markov définit une limite supérieure de l'entropie [154]. Il est donc toujours possible de faire aussi bien voire mieux que l'entropie donnée par le modèle de Markov. Cela s'explique par le fait que le modèle ne tient pas compte des dépendances inter-plages.

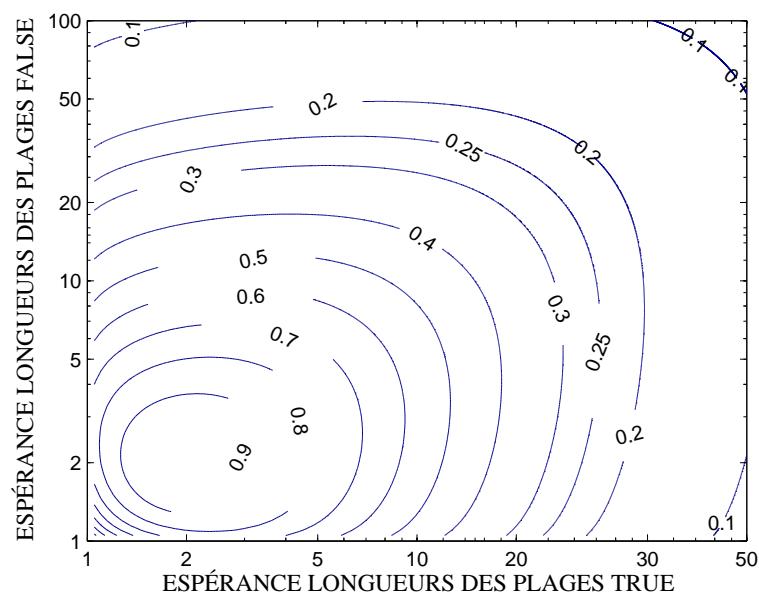


Figure A.3 Iso-contours de l'entropie $H(Z)$ en fonction des espérances des plages de 1s et de 0s.

BIBLIOGRAPHIE

- [1] I. Maniatis, G. Nikolouzou, and S. Venieris, "QoS Issues in the Converged 3G Wireless and Wired Networks," *IEEE Communication Magazine*, Aug. 2002.
- [2] Recommandation G.711, "Pulse code modulation (PCM) of voice frequencies," Tech. Rep., International Telecommunication Union (ITU-T), 1988.
- [3] B.S. Atal and M.R. Schroeder, "Code-Excited Linear Prediction (CELP) : high-quality speech at very low bit rates," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, 1985.
- [4] K. Järvinen, "Standardisation of the adaptive multi-rate codec," *European Signal Processing Conference (EUSIPCO)*, Sept. 2000.
- [5] R. Salami and al., "Design and description of CS-CELP : a toll quality 8kb/s speech coder," *IEEE Trans. on Speech and Audio Processing*, vol. 4, no. 2, Mar. 1998.
- [6] Xavier Maitre, "7 kHz Audio Coding Within 64 kbit/s," *IEEE Journal on Selected Areas on Communications*, vol. 6, no. 2, pp. 283–298, Feb. 1988.
- [7] B. Bessette et al., "The adaptive multirate wideband speech codec (AMR-WB)," *IEEE Trans. on Speech Audio Processing*, vol. 10, no. 8, pp. 620–636, Nov. 2002.
- [8] M. Jelinek and al., "On the architecture of the CDMA2000 variable-rate multimode wideband (VMR-WB) speech coding standard," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, vol. 1, pp. 281–284, May 2004.
- [9] Question 10 ITU-T SG16 Temporary Documents, "Terms of reference (ToR) for the G.729 based Embedded Variable Bit-Rate (G.729EV) extension to the ITU-T G.729 speech codec," Tech. Rep., International Telecommunication Union (ITU-T), July 2004.
- [10] ISO/IEC JTC1/SC29/WG11 (MPEG), "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s," Tech. Rep., International Standard ISO/IEC 11172-3, 1992.
- [11] M. Bosi and al., "ISO/IEC MPEG-2 advanced audio coding," *AES 101th Convention*, 1996.
- [12] J. Makinen and al., "AMR-WB+ : a new audio coding standard for 3rd generation mobile audio services," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 1109–1112, Mar. 2005.
- [13] 3 GPP TS 26.401, "Enhanced aacPlus general audio codec ; General description," Tech. Rep., The 3rd Generation Partnership Project (3GPP), Dec. 2004.
- [14] A. Gersho and R.M. Gray, *Vector quantization and signal compression*, Kluwer Academic Publishers, 1992.
- [15] Nicolas Moreau, *Techniques de compression des signaux*, chapter 3, pp. 21–72, Collection Technique et Scientifique des Télécommunications. Masson, 1995.
- [16] C.E. Shannon, "A Mathematical theory of communication," *University of Illinois Press*, 1949.
- [17] A. Le Guyader, P. Philippe, and J.B. Rault, "Synthèse des normes decodage de la parole et du son (UIT-T, ETSI, ISO/MPEG)," *Annales des Télécommunications*, vol. 55, no. 9-10, pp. 425–441, Sept. 2000.

- [18] W.B. Kleijn and K.K. Paliwal, *Speech coding and synthesis*, chapter An introduction to speech coding, pp. 1–47, Elsevier Science, 1995.
- [19] J.D. Markel and A.H. Gray, *Linear prediction of speech*, Springer-Verlag, New-York, 1976.
- [20] L. Rabiner and R. Schafer, *Digital processing of speech signals*, Englewood Cliffs, 1978.
- [21] John Makhoul, “Linear prediction : a tutorial review,” *Proc. of the IEEE*, vol. 63, no. 4, pp. 561–580, 1975.
- [22] B.S. Atal and M.R. Shroeder, “Predictive coding of speech signals,” *Proc. IEEE Conf. on communication and Processing*, pp. 360–361, 1967.
- [23] A. Oppenheim and R. Schafer, “Homomorphic analysis of speech,” *IEEE Trans. on Audio and Electroacoustics*, pp. 118–123, 1968.
- [24] J.H. Chung and R.W. Schafer, “A 4.8 Kbps homomorphic vocoder using analysis by synthesis excitation analysis,” *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 144–147, 1989.
- [25] H.W. Strube, “Linear prediction on a warped frequency scale,” *J. Acoust. Soc. Amer.*, vol. 68, pp. 1071–1076, Oct. 1980.
- [26] S. Imai, “Cepstral analysis synthesis on the mel frequency scale,” *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pp. 93–96, Apr. 1983.
- [27] J-P. Adoul and R. Lefebvre, *Speech coding and synthesis*, chapter Wideband speech coding, elsevier, 1995.
- [28] CCIT Recommendation G.721, “32 kbit/s adaptive differential pulse code modulation (ADPCM),” Tech. Rep., CCIT, Oct. 1988.
- [29] ITU-T Recommendation G.726, “40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM),” Tech. Rep., International Telecommunication Union (ITU-T), Dec. 1990.
- [30] ITU-T Recommendation G.722.1, “Low-complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss,” Tech. Rep., International Telecommunication Union (ITU-T), May 2005.
- [31] Federal Standard 1015, “Telecommunications : Analog to digital conversion of radio voice by 2400 bit/second linear predictive coding, national communication system,” Tech. Rep., National communication System-Office of Technology and Standards, Nov. 1984.
- [32] Federal Standard 1016, “Telecommunications : Analog to digital conversion of radio voice by 4800 bit/second code excited linear prediction (CELP),” Tech. Rep., National communication System-Office of Technology and Standards, Feb. 1991.
- [33] I. Gerson and M. Jasiuk, “Vector sum excited linear prediction (VSELP) speech coding at 8 kbit/s,” *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pp. 461–464, Apr. 1990.
- [34] C. Laflamme, J-P. Adoul, H.Y. Su, and S. Morissette, “On reducing computational complexity of codebook search in CELP Coder through the use of algebraic codes,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 177–180, 1990.
- [35] E. Zwicker and H. Fastl, *Psychoacoustics : facts and models*, Springer-Verlag, 1999.
- [36] Raymond Veldhuis and Armin Kohlrausch, *Speech coding and synthesis*, chapter Waveform coding and auditory masking, pp. 397–431, elsevier, 1995.

- [37] E. Zwicker and E. Terhardt, "Analytical expression for critical-Band Rate and Critical Band-Width as a Function of Frequency," *J. Acoust. Soc. Am.*, , no. 68, pp. 1523–1525, 1980.
- [38] James D. Johnston, "Transform Coding of Audio Signals Using Perceptual Noise Criteria," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 2, Feb. 1988.
- [39] ITU-R Recommandation BS.1387, "Method for objective measurements of perceived audio quality," Tech. Rep., International Telecommunication Union (ITU-R), July 1999.
- [40] Christopher R. Cave, "Perceptual modelling for low-rate audio coding," M.S. thesis, Mc Gill University, June 2002.
- [41] T. Berger, *A mathematical basis for data compression*, Prentice-Hall, 1971.
- [42] John P. Princen and Alan Bernard Bradley, "Analysis/synthesis filter bank design based on domain aliasing cancellation," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, 1986.
- [43] A. Croisier, D. Esyeban, and C. Galand, "Perfect channel splitting by use of interpolation, decimation and tree decomposition techniques," *Proc. Int. Conf. Inform. Sci. Syst.*, pp. 443–446, Aug. 1976.
- [44] H.J. Nussbaumer, "Pseudo-QMF filter bank," *IBM Tech. Disclosure Bull.*, vol. 24, pp. 3081–3087, Nov. 1981.
- [45] H.S. Malvar, *Signal processing with lapped transforms*, Artech House, Inc., Norwood, MA, USA, 1992.
- [46] J. Mau, "Perfect reconstruction modulated filter," *Proc. of International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, pp. 273–276, Mar. 1992.
- [47] H. Najafzadeh-Azghandi, *Perceptual coding of narrowband Audio signals*, Ph.D. thesis, Mc Gill University, Apr. 2000.
- [48] R. Gluth, "Regular FFT-related transform kernels for DCT/DST-based polyphase filter banks," *Proc. of International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, pp. 2205–2208, May 1991.
- [49] Y. Wang, L. Yaroslavsky, M. Vilermo, and M. Väänänen, "5th International Conference on Signal Processing (ICSP)", Aug. 2000.
- [50] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE ASSP Magazine*, pp. 14–37, Oct. 1991.
- [51] L.C. Vargas and H.S. Malvar., "ELT-Based wavelet coding of high-fidelity audio signals," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 124–127, 1993.
- [52] S. Levine, *Audio representation for data compression and compressed domain processing*, Ph.D. thesis, Stanford University, 1998.
- [53] A. Ferreira, "Perceptual audio coding and the choice of an analysis/synthesis filter bank and psychoacoustic model," *AES 104th Convention*, May 1998.
- [54] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proc. of the IEEE*, vol. 4, no. 88, pp. 451–513, Apr. 2000.
- [55] J. Johnston et al., "AT&T perceptual audio coding (PAC)," *Collected Papers on Digital Audio Bit-Rate reduction*, pp. 73–81, 1996.

- [56] D. Sinha and J. Johnston, "Audio compression at low bit rates using a signal adaptive switched filterbank," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1053–1056, 1996.
- [57] Y. Mathieu and J.-P. Petit, "High-quality audio transform coding at 64 kbps," *IEEE Trans. Commun.*, vol. 42, no. 11, Nov. 1994.
- [58] J. Herre and J.D. Johnston, "Enhancing the performance of perceptual coding coders using temporal noise shaping (TNS)," *AES 101th Convention*, 1996.
- [59] T. Moriya et al., "A design transform coder for both speech and audio signals at 1 bit/sample," *Proc. of International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, pp. 1371–1374, 1997.
- [60] M. Dietz et al., "Spectral band replication - A novel approach in audio coding," *AES 112th Convention*, 2002.
- [61] M. Wolters, K. Kjorling, D. Homm, and H. Purnhagen, "A closer look into MPEG-4 high efficiency AAC," *AES 115th Convention*, 2005.
- [62] R. Lefebvre, R. Salami, C. Laflamme, and J.P. Adoul, "8 Kbit/s coding of speech with 6 ms frame-length," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 612–615, 1993.
- [63] J.-H. Chen and D. Wang, "Transform predictive coding of wideband speech signals," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 275–278, 1996.
- [64] R. Lefebvre, R. Salami, C. Laflamme, and J.P. Adoul, "High quality of wideband audio signals using transform coded excitation (TCX)," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 193–196, 1994.
- [65] L. Tancerel, S. Ragot, V.T. Ruoppila, and R. Lefebvre, "Combined speech and audio coding by discrimination," *IEEE Speech Coding Workshop*, pp. 158–160, Sept. 2000.
- [66] S.A. Ramprashad, "The multimode transform predictive coding paradigm," *IEEE Trans. on Speech and Audio Processing*, vol. 11, no. 2, pp. 117–129, Mar. 2003.
- [67] B. Bessette, R. Lefebvre, and R. Salami, "Universal speech/audio coding using hybrid ACELP/TCX techniques," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 301–304, Mar. 2005.
- [68] S.A. Ramprashad, "Embedded coding using a mixed speech and audio coding paradigm," *International Journal of Speech Technology*, vol. 2, pp. 359–372, 1999.
- [69] Hervé Taddei, *Codage hiérarchique faible retard 8-14.1-24 kbit/s pour les nouveaux réseaux et services*, Ph.D. thesis, Université de Rennes I, 1999.
- [70] B. Kovesi, D. Massaloux, and A. Solla, "A scalable speech and audio coding scheme with continuous bitrate flexibility," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 273–276, May 2004.
- [71] P. Dymarski, N. Moreau, and A. Vigier, "Optimal and sub-optimal algorithms for selecting the excitation in linear predictive coders," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1990.
- [72] Mark Johnson and Tomohiko Taniguchi, "Low-complexity multi-mode VXC using multi-stage optimization and mode selection," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 221–224, 1991.

- [73] A. Le Guyader, C. Lamblin, and E. Boursicaut, “Embedded algebraic CELP/VSELP coders for wideband speech coding,” *Speech Communication*, pp. 319–328, 1995.
- [74] Rosario Drogo De Iacovo and Daniele Sereno, “Embedded CELP coding for variable bit-rate between 6.4 and 9.6 Kbit/s,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 681–684, 1991.
- [75] T. Nomura, M. Iwadare, M. Serizawa, and K. Ozawa, “A bitrate and bandwidth scalable CELP coder,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 341–344, 1998.
- [76] Jürgen Herre and Bernhard Grill, “Overview of MPEG-4 audio and its applications in mobile communications,” *International Conference on Communication Technologie (ICCT)*, 2000.
- [77] Christoph Erdmann, David Bauer, and Peter Vary, “Pyramid CELP : Embedded speech coding for packet communications,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 181–184, 2002.
- [78] Sean A. Ramprashad, “High quality embedded wideband speech coding using an inherently layered coding paradigm,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2000.
- [79] S. Voran, “Listener rating of speech passbands,” *IEEE Workshop on Speech Coding*, pp. 81–82, 1997.
- [80] S. Hayashi A. Kataoka, S. Kurihara, “A 16-Kbit/s wideband speech codec scalable with G.729,” *Eurospeech*, vol. 3, pp. 1491–1494, 1997.
- [81] A. McCree, “A 14 kb/s wideband speech coder with a parameteric highband model,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing, ICASSP’00*, pp. 1153–1156, 2000.
- [82] Kazuhito Koishida, Vladimir Cuperman, and Allen Gersho, “A 16-Kbit/s bandwidth scalable audio coder based on the G.729 standard,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1149–1152, 2000.
- [83] J.-M. Valin and R. Lefebvre, “Bandwidth extension of narrowband speech for low bit-rate wideband coding,” *IEEE Speech Coding Workshop*, pp. 130–132, Sept. 2000.
- [84] R. Taori, R. J. Sluijter, and A. J. Gerrits, “Hi-bin : an alternative approach to wideband speech coding,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1157–1160, 2000.
- [85] J. Epps and W.H. Holmes, “A new very low bit rate wideband speech coder with a sinusoidal highband model,” *Proc. IEEE Int. Symp. on Circuits and Systems (Sydney, Aust.)*, vol. II, pp. 349–352, 2001.
- [86] K.-T. Kim, S.-K. Jung, Y.-C. Clark, and D.H. Youn, “A new bandwidth scalable wideband speech/audio coder,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 657–660, 2002.
- [87] S.-K. Jung, K.-T. Kim, and H.-G Kang, “A bit-rate/bandwidth scalable speech coder based on itu-t g.723.1 standard,” *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 285–288, May 2004.
- [88] Junji Suzuki et Noahisa Ohta, “Variable rate coding scheme for audio signal with subband and embedded coding techniques,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 188–191, 1989.

- [89] G. Aguilar, J.-H. Chen, R.B. Dunn, R.J. McAulay, X. Sun, W. Wang, and R. Zopf, "An embedded sinusoidal transform codec with measured phases and sampling rate scalability," *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1141–1144, 2000.
- [90] Romain Trilling, "Codage large bande de la parole par encapsulation du codeur ITU G.729 (CS-ACELP)," 1998.
- [91] H. Carl and U. Heute, "Bandwidth enhancement of narrow-band speech signals," *Proc. EUSIPCO*, vol. 2, pp. 1178–1181, 1994.
- [92] Y. Yoshida and M. Abe, "An algorithm to reconstruct wideband speech from narrowband speech based on codebook mapping," *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, pp. 1591–1594, 1994.
- [93] Ming Cheng, Yan, Douglas O'Shaughnessy, and Paul Mermelstein, "Statistical recovery of wideband speech from narrowband speech," *IEEE Transactions On Speech And Audio Processing*, vol. 2, no. 4, Oct. 1994.
- [94] J. Epps and W.H. Holmes, "A new technique for wideband enhancement of coded narrowband speech," *IEEE Workshop on Speech Coding*, pp. 371–374, 1999.
- [95] Peter Jax and Peter Vary, "Wideband extension of telephone speech using a hidden markov model," *IEEE Workshop on Speech Coding*, pp. 133–135, september 2000.
- [96] M. Nilson, S.V. Andersen, and W.B. Kleijn, "On the mutual information between frequency bands in speech," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 1327–1330, June 2000.
- [97] Peter Jax and Peter Vary, "An upper bound on the quality of artificial bandwidth extension of narrowband speech signals," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 237–240, May 2002.
- [98] Yannis Agiomyrgiannakis and Yannis Stylianou, "Combined estimation/coding of highband spectral envelopes for speech spectrum expansion," *IEEE Int. Conf. on Acoustics Speech and Signal Processing*, 2004.
- [99] C.F Chan and W.K Hui, "Wideband re-synthesis of narrow-band CELP coded speech using multi-band excitation model," *Proc. ICSLP*, vol. 1, pp. 667–670, 1996.
- [100] John Makhoul and Michael Berouti, "High-Frequency Regeneration In Speech Coding Systems," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 428–431, April 1979.
- [101] E. Larsen, R.M. Aarts, and M. Danessis, "Efficient high-frequency bandwidth extension of music and speech," *AES 112th Convention*, May 2002.
- [102] N. S. Jayant and Peter Noll, *Digital coding of waveforms - Principles and applications to speech and video*, Prentice-Hall, 1984.
- [103] Recommendation ITU-R BS.1534-1, "Method for the subjective assessment of intermediate quality level of coding systems," Tech. Rep., International Telecommunication Union (ITU-R), Nov. 2003.
- [104] J.H. Chen and A. Gersho, "Adaptive postfiltering for quality enhancement of coded speech," *IEEE Trans. on Speech and Audio Processing*, vol. 3, no. 1, pp. 59–71, Jan. 1995.
- [105] M. Vilermo et al., "Perceptual optimization of the frequency selective switch in scalable audio coding," *114th AES convention*, Mar. 2003.

- [106] W.R. Bennett, "Spectra of quantized signals," *Bell systems Technical Journal*, pp. 446–472, July 1948.
- [107] P. Zador, "Asymptotic quantization error of continuous signals and the quantization dimension," *IEEE Trans. on Information Theory*, vol. IT-28, pp. 139–149, Mar. 1982.
- [108] T.D. Lookabaugh and R.M. Gray, "High-Resolution Quantization Theory and the Vector Quantizer Advantage," *IEEE Trans. on Information Theory*, vol. 35, no. 5, pp. 1020–1033, Sept. 1989.
- [109] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. on Information Theory*, vol. IT-25, pp. 373–380, July 1979.
- [110] J.H. Conway and N.J.A. Sloane, "A lower bound on the average error of vector quantizers," *IEEE Trans. Information Theory*, vol. IT-31, pp. 106–109, Jan. 1985.
- [111] J.H. Conway and N.J.A. Sloane, *Sphere packings, lattice, groups*, Springer-Verlag, New York, 1988.
- [112] J.-P. Adoul, C. Lamblin, and A. Leguyader, "Baseband speech coding at 2400 bps using spherical vector quantization," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, vol. 1, pp. 1.12.1–1.12.4, 1984.
- [113] T.R. Fisher, "A pyramid vector quantizer," *IEEE Trans. on Information Theory*, vol. 32, pp. 568–583, 1986.
- [114] J.H. Conway and N.J.A. Sloane, "A Fast Encoding Method for Lattice Codes and Quantizers," *IEEE Trans. on Information Theory*, vol. 29, no. 6, pp. 820–824, Nov. 1983.
- [115] Minjie Xie, *Quantification vectorielle algébrique et codage de parole en bande élargie*, Ph.D. thesis, Université de Sherbrooke (Québec), Feb. 1996.
- [116] Stéphane Ragot, *Nouvelles techniques de quantification vectorielle algébrique basées sur le codage de Voronoï - Application au codage AMR-WB+*, Ph.D. thesis, Université de Sherbrooke (Québec), May 2003.
- [117] Thomas R. Fisher, "Geometric Source Coding and Vector Quantization," *IEEE Trans. on Information Theory*, vol. 35, no. 1, pp. 137–145, Jan. 1989.
- [118] N.J.A. Sloane, "Tables and sphere packings and spherical codes," *IEEE Trans. on Information Theory*, vol. 27, no. 3, pp. 327–338, May 1981.
- [119] J.H. Conway and N.J.A. Sloane, "Fast Quantization and Decoding Algorithms for Lattice Quantizers and Codes," *IEEE Trans. on Information Theory*, vol. 28, no. 2, pp. 211–226, Mar. 1982.
- [120] K. Sayood, J.D. Gibson, and M.C. Rost, "An algorithm for uniform vector quantizer Design," *IEEE Trans. on Information Theory*, vol. 30, no. 6, pp. 805–814, Nov. 1984.
- [121] M. Antonini, M. Barlaud, and P. Mathieu, "Image coding using lattice vector quantization of wavelet coefficients," *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2273–2276, May 1991.
- [122] M. Barlaud, P. Sole, T. Gaidon, M. Antonini, and P. Mathieu, "Pyramidal lattice vector quantization for multiscale image coding," *IEEE Trans. Image Processing*, pp. 367–381, July 1994.
- [123] A. Woolf and G. Rogers, "Lattice vector quantization of image wavelet coefficient vectors using a simplified form of entropy coding," *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, vol. 5, pp. 269–272, Apr. 1994.

- [124] W.-H. Kim, Y.-H. Hu, and T.Q. Nguyen, “Wavelet-based image coder with entropy-constrained lattice vector quantizer (ECLVQ),” *IEEE Trans. on Circuits and Systems*, vol. 45, no. 8, pp. 1015–1030, Aug. 1998.
- [125] J.M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *IEEE Trans. on Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.
- [126] A. Said and W.A. Pearlman, “A new fast and efficient image codec based on set partitioning in hierarchical trees,” *IEEE Trans. on Circuits Systems Video Technol.*, vol. 6, pp. 243–250, June 1996.
- [127] D. Taubman, “High performance scalable image compression with EBCOT,” *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, vol. 3, pp. 344–348, Oct. 1999.
- [128] E.A.B. Da Silva, D.G. Sampson, and M. Ghanbari, “A successive approximation vector quantizer for wavelet transform image coding,” *IEEE Trans. Image Processing*, vol. 5, pp. 229–310, Feb. 1996.
- [129] J. Knipe and B. Han, “An improved lattice vector quantization scheme for wavelet compression,” *IEEE Trans. Signal Processing*, vol. 46, pp. 239–243, Jan. 1998.
- [130] D. Mukherjee and S.K. Mitra, “Successive refinement lattice vector quantization,” *IEEE Trans. on Signal Processing*, vol. 11, no. 12, pp. 1337–1348, Dec. 2002.
- [131] C. Lamblin, J.-P. Adoul, D. Massaloux, and S. Morissette, “Fast CELP coding based on the Barnes-Wall lattice in 16 dimensions,” *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, pp. 61–64, May 1989.
- [132] M. Xie and J.P. Adoul, “Embedded algebraic vector quantizers (EAVQ) with application to wideband speech coding,” *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 240–243, May 1996.
- [133] S. Ragot, B. Bessette, and R. Lefebvre, “Low-complexity multi-rate lattice vector quantization with application to wideband TCX speech coding at 32 kbit/s,” *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2004.
- [134] J. Makinen and al., “AMR-WB+ : a new audio coding standard for 3rd generation mobile audio services,” *IEEE Int. conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 1109–1112, Mar. 2005.
- [135] G.D. Forney, “Coset codes. I. Introduction and geometrical classification,” *IEEE Trans. on Information Theory*, vol. 34, no. 5, pp. 1123–1151, Sept. 1988.
- [136] G.D. Forney, “Multidimensional constellations - part II : Voronoï constellations,” *IEEE Trans. on Delect. Areas in Commun.*, vol. 7, no. 6, pp. 941–958, Aug. 1989.
- [137] C. Lamblin and J.-P. Adoul, “Algorithme de quantification vectorielle sphérique à partir du réseau de Gosset d’ordre 8,” *Annales de Télécommunications*, vol. 43, no. 3-4, pp. 172–186, 1988.
- [138] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, Wiley, 1991.
- [139] A.J.S. Ferreira, “Combined Spectral Envelope Normalization and Subtraction of Sinusoidal Components in the ODFT and MDCT Frequency Domains,” *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1986.
- [140] A.V. Oppenheim, R.W. Schafer, and J.R. Buck, *Discrete-time signal processing*, Prentice Hall, 1999.
- [141] M. Athineos and D.P.W. Ellis, “Frequency-domain linear prediction for temporal features,” *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 261–266, 2003.

- [142] D. Sinha and A.H. Tewk, "Low bit rate transparent audio compression using adapted wavelets," *IEEE trans. on Speech and Audio Processing*, vol. 41, no. 12, pp. 3463–3479, Dec. 1993.
- [143] M. Nilsson, S.V. Andersen, and W.B. Kleijn, "Gaussian mixture model based on mutual information estimation between frequency bands in speech," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, 2002.
- [144] Alan McCree, Takahiro Unno, Anand Anandakumar, and Alexis Bernard, "An embedded adaptive multi-rate wideband speech coder," *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)*, pp. 761–764, 2001.
- [145] J.D. Johnston, "A filter family designed for use in quadrature mirror filter banks," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 291–294, Apr. 1980.
- [146] Recommandation P.862.2, "Wideband extension to recommandation P.862 for the assessment wideband telephone networks and speech codecs," Tech. Rep., International Telecommunication Union (ITU-T), Nov. 2005.
- [147] A. Kurashima C. Morioka and A. Takahashi, "Proposal on objectif speech quality assessment for wideband IP telephony," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 49–53, May 2005.
- [148] G. Fuchs and R. Lefebvre, "A new post-filtering for articially replicated high-band in speech coders," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2006.
- [149] G. Fuchs and R. Lefebvre, "A speech coder post-processor controlled by side-information," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2005.
- [150] G. Fuchs and R. Lefebvre, "A scalable CELP/transform coder for low bit Rate speech and audio coding," *Proc. 120th Convention of AES*, May 2006.
- [151] C. Faller, "Parametric multichannel audio coding : synthesis of coherence cues," *IEEE Trans. on Audio, Speech, and Language Processing*, , no. 1, pp. 290–310, Jan. 2005.
- [152] J. Breebaart et al., "Mpeg spatial audio coding / mpeg surround : overview and current status," *119th Convention of AES*, 2005.
- [153] R.J. McAulay and Th.F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. on Acoust., Speech and Signal Processing*, vol. 34, pp. 744–754, 1986.
- [154] J. Capon, "A probabilistic model for run-length coding of picture," *IRE Trans. on Information Theory*, pp. 278–287, Mar. 1959.

