UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE HOUARI BOUMEDIENNE (USTHB) – ALGER

FACULTE D'ELECTRONIQUE

DEPARTEMENT D' ELECTRONIQUE

THESE

Présentée à l'USTHB pour obtenir le diplôme de

MAGISTER

Spécialité en Electronique

Electronique des systèmes

Par

Kamel AMIR

Conception et réalisation d'un codeur/décodeur de la parole à 8 kbits/s et à faible retard

Soutenue le : **16-12-2002** devant le jury :

Mr. HAOUACINE	**Professeur, USTHB**	**Président**
Mr. HALIMI	**Chargé de recherche, CDTA**	**Directeur de thèse**
Mme. GUERTI	**Maître de conférence, EMP**	**Examinatrice**
Mme. BOUDRAA	**Maître de conférence, USTHB**	**Examinatrice**
Mr. DJERADI	**Maître de conférence, USTHB**	**Examinateur**

Table des matières

Liste des tableaux

Liste des figures

Liste des Acronymes

		Première apparition dans les pages
MIC	Modulation d'Impulsions Codées……	2
PCM	Pulse Coded Modulation……	2
MICDA	Modulation d'Impulsions Codées Différentielle Adaptative……	2
ADPCM	Adaptative Delta Pulse Coded Modulation……	2
CELP	Code Excited Linear Prediction……	3
LD-CELP	Low Delay Code Excited Linear Prediction……	3
LPAS	Linear Prediction Analysis by Synthesis……	3
MOS	Mean Opinion Score……	18
LPC	Linear Prediction Coding……..	20
RSB	Rapport Signal à Bruit……	21
RSB seg.	Rapport Signal à Bruit segmental……..	21
DRT	Diagnostic Rhyme Test……...	22
DAM	Diagnostic Acceptability Measure……	22
QV	Quantification Vectorielle……	24
GLA	Generalized Lloyd Algorithm……..	30
ARMA	Auto Régressif à Moyenne Ajustée……...	35
MA	Moyenne Ajustée……	35
AR	Auto Régressif……	35
ACELP	Algebric Code Excited Linear Prediction……..	43
RMS	Root Mean Square……...	61
PDF	Probability Density Fonction……...	62
IIR	Infinite Impulse Response……..	63
ZIR	Zero Input Response……...	71
MSE	Mean Squared Error……	72
LBG	LINDE BUZO GRAY (Algorithme)……	74
LTP	Long Term Predictor……..	84

ملخص

ان تخصيص مرامز الكلام مرتكز على ثلاث خصائص أساسية :
النوعية، التدفق و صعوبة الاءنجاز. حاليا عامل التأخر أصبح من أهم الاءهتمامات في كثير من الحالات. الاختيار اللأئق لهذا العامل للاستعلامات الشبكية أدى الى تطور المرمز LD-CELP تحت تدفق 16 كيلوبيت في الثانية ذو النوعية العالية وذو تأخر ضعيف يساوي 2 ميليثانية. هذا الخوارزمي تمت الموافقة عليه من طرف هيئة UIT تحت تسمية G728 .

في اطار هذه الأطروحة تم انجاز مرمز الكلام ذو حزمة ضيقة من 300 الى 3400 هرتز تحت تدفق 8 كيلوبيت على الثانية و تأخر يساوي 10 ميليثانية. و قد تم تخفيض درجة التعقيد في عملية البحث لأحسن شعاع الترميز وذلك بتحويل عمليات الترشيح الى حاصل ضرب المصفوفات. بعد ذلك استعملنا تقنية الترشيح الخلفي لتخفيض درجة التعقيد في عملية البحث في القاموس. و استعملنا مرشح تنبؤ لترميم دورية اءشارة الكلام (PITCH).

القياسات الموضوعية أثبتت أن مرمز الكلام الذي أنجز أعطى نوعية أحسن لما ادخلنا عليه مرشح تنبؤ ترميم دورية الكلام.

Résumé

La caractérisation des codeurs de la parole est traditionnellement basée sur trois critères fondamentaux : qualité, débit et complexité d'implémentation. Récemment, le facteur retard (ou délai) est devenu une spécification importante pour beaucoup d'applications. Un choix rigoureux du retard pour les applications à réseaux a conduit au développement du codeur LD-CELP 16 kb/s de qualité supérieure (toll quality) et avec un retard dans un seul sens de seulement 2 ms. Cet algorithme a été adopté par le CCITT comme recommandation G728.

Dans notre thèse, nous décrivons l'extension apportée au codeur réalisé au CDTA (équivalent du G728) afin d'obtenir un codeur à faible retard (<=10 ms) et opérant à 8 kbits/s. Donc un LD-CELP 8 kb/s a été conçu en doublant la taille du vecteur d'excitation de 5 à 10 échantillons et en gardant le même algorithme. Le LD-CELP 8 kb/s produit un signal de parole plutôt bruyant. La conclusion que nous pouvons tirer à partir de cette expérience est claire : pour avoir une meilleure qualité de parole à 8 kb/s avec un faible délai, nous devons exploiter la redondance du pitch de façon explicite. Pour cela, nous avons augmenté la taille des trames à 20 échantillons (2.5 ms) et avons rajouté un *prédicteur pitch*. Dans une première étape, l'adaptation du prédicteur pitch s'est faite en backward (à posteriori) et pour la seconde étape elle s'est effectué en forward (à priori). L'ordre du filtre LPC a été réduit à 20.

Les mesures objectives montrent que le codeur donne une meilleure qualité après l'incorporation du prédicteur pitch. Afin de réduire la complexité de calcul, nous avons utilisé un dictionnaire algébrique et la technique du Backward Filtering.

Le codeur LD-CELP à 8 kb/s est utilisé dans les réseaux téléphoniques et peut être utilisé comme programme de compression de signaux de parole dans les supports de stockage.

Abstract

Speech coders have traditionally been characterized on the basis of three primary criteria : quality, rate and implementation complexity. Recently delay has also become an important specification for many applications. A very stringent delay objective for network applications, led to the development of the 16 kb/s LD-CELP algorithm, with toll quality and a one-way delay of only 2 ms. This algorithm has been adopted as CCITT Recommendation G.728.

In our thesis, we describe our attempts to extend the G728 in order to have a low delay codec operating at 8 kbits/s. So a G.728-compatible 8 kb/s LD-CELP is designed by doubling the excitation vector dimension from 5 to 10 samples and keeping the same algorithm. The 8 kb/s LD-CELP coder produced rather noisy speech. The conclusion from this experiment was clear: to achieve a better speech quality at 8 kb/s with low delay, we need to exploit the pitch redundancy explicitly. Hence, we have taken a frame delay of 2.5 ms (20 samples) and added in a first step a backward adaptive pitch predictor and in the second step a forward adaptive pitch predictor. The LPC order predictor is reduced to 20.

Objective measures show that the coder achieves better quality after the incorporation of the pitch predictor. In order to reduce the computational complexity we used an algebraic codebook and the Backward Filtering technique.

The 8 kb/s LD-CELP is used in telephonic and sub-telephonic networks and may be used as compression program for speech supports.

Introduction Générale

Pour transmettre (ou stocker) un signal, que cela soit de la parole, de la musique ou des images, on utilisait jusqu'à une époque récente presque exclusivement des chaînes de communications analogiques. A cause des perturbations et des bruits apparaissant inévitablement dans le canal de transmission, le signal reconstruit au récepteur ne pouvait être une réplique exacte du signal émis. L'avantage principal des chaînes de communication numériques, largement utilisées actuellement, est de pouvoir transmettre un message obtenu par numérisation d'un signal avec un taux d'erreur aussi faible que l'on veut. On entend par numérisation d'un signal à temps continu s(t) la réalisation d'un échantillonnage (discrétisation de l'axe temporel) et d'une quantification (discrétisation des amplitudes). L'inconvénient de cette technique est d'introduire une distorsion lors de la numérisation et de nécessiter une plus large occupation spectrale lors de la transmission. Pour économiser les ressources des canaux de communication, on cherche à compresser le signal numérisé. C'est cette double opération de numérisation et de compression que l'on appelle **codage de source** [1].

Cette double opération de numérisation et de compression doit se faire en respectant un certain nombre de contraintes que l'on retrouve habituellement quel que soit le signal et quelle que soit l'application mais à des degrés divers. Définir un codeur est, comme souvent en traitement de signal une affaire de compromis.

Il faut d'abord, pour un débit visé, évaluer la dégradation apportée au signal et se demander si cette dégradation est tolérable. C'est un problème difficile parce que l'évaluation de la dégradation de façon objective à l'aide d'une mesure de distorsion du type erreur quadratique moyenne ou rapport signal sur bruit est insuffisante. Bien sûr, pour mettre en place un développement théorique ou pour définir des algorithmes, on utilisera de telles mesures mais, en dernière analyse ce qui comptera sera l'opinion subjective de l'utilisateur et cet utilisateur peut avoir des désirs très variés. Par exemple, il existe souvent entre les exigences des militaires et le souhait du grand public des différences marquées : un codeur de parole pour des militaires ne devra pas avoir les mêmes caractéristiques qu'un codeur pour le grand public.

Il faut ensuite rendre le coût du traitement inférieur à un certain seuil. Une nouvelle fois, c'est une notion très relative. Pour un signal de parole, dans des applications de type téléphonie, le critère habituel est la possibilité d'implanter tous les algorithmes nécessaires dans un seul microprocesseur de traitement du signal.

Il faut également garder une bonne robustesse aux erreurs de transmission. L'affirmation précédente, à savoir que l'avantage principal des chaînes de communication numériques est de pouvoir transmettre un message avec un taux d'erreur aussi faible que l'on veut, correspond à une vision ''idéaliste''. Dans la pratique, spécialement lorsque l'on utilise le canal radio pour la transmission du message, il faut protéger fortement l'information et malgré cela , des erreurs désagréables restent inévitables. Malgré son importance, il est encore peu habituel de chercher à optimiser conjointement le codeur de source et le codage du canal. On continue à appliquer le deuxième théorème de Shannon qui explique que l'optimisation d'une chaîne de transmission peut se faire en optimisant séparément les deux codeurs précédents [2]. L'étude de cette optimisation conjointe reste encore du domaine de la recherche.

Enfin, il faut généralement rendre le délai de reconstruction le plus faible possible. C'est obligatoire pour des applications *full-duplex* , par exemple, pour des conversations téléphoniques.

Le codage de la parole est essentiel dans les efforts d'obtenir un usage plus efficace des réseaux de télécommunications numériques, en particulier les réseaux cellulaires, et pour réduire la mémoire nécessaire dans les systèmes de stockage de la parole.

Précisons un qualificatif habituellement associé au *signal de parole*. On parle de signal de parole dans la bande téléphonique si on filtre le signal à temps continu dans la bande [300-3400 Hz] puis si l'on échantillonne à la fréquence de 8 kHz. On parle de signal à bande élargie si on le filtre dans la bande [50-7000 Hz] puis on l'échantillonne à 16 kHz. Le signal de parole dans la bande téléphonique est celui qui est transmis dans le réseau téléphonique public. Les enjeux économiques sont très nettement les plus significatifs dans cette bande. L'intérêt de transmettre du signal de parole en bande élargie est d'obtenir un signal de parole reconstitué plus net et avec un meilleur confort d'écoute que dans le cas précédent. Les applications sont les conférences audiovisuelles, le visiophone, la téléphonie sur haut-parleurs, etc.

Plusieurs normes ont été recommandées par L'UIT-T pour le réseau téléphonique public ces vingt dernières années. Depuis 1972, la norme internationale G.711 précise un codage par modulation par impulsion codées (MIC ou PCM pour pulse code modulation) correspondant à un débit de 64 kbit/s : l'amplitude des échantillons est simplement quantifiée sur 8 bits après une compression de type non linéaire. Depuis 1984, la norme G.721 définit un codage MIC différentiel adaptatif (MICDA ou ADPCM) correspondant à un débit de 32 kbit/s : on ne quantifie plus directement l'amplitude de l'échantillon mais la différence entre l'amplitude et

une valeur prédite déterminée par un filtrage de type adaptatif. Un codeur à 16 kbit/s basé sur des techniques de modélisation et de quantification vectorielle, a été sélectionné par l'UIT-T en 1991. Cette norme G.728, appelée également LD-CELP (Low Delay Code Excited Linear Predictive coder) met en évidence le fait que c'est un codeur de type CELP et qu'il présente un faible délai de reconstruction, propriété particulièrement importante pour un échange téléphonique.

Dans ce mémoire, le but de notre travail est la réalisation d'un codeur-décodeur de type LD-CELP ayant les caractéristiques sont les suivantes :

- débit égal à 8 kbit/s;
- largeur de bande : 300 Hz – 3400 Hz (bande téléphonique).
- retard de codage inférieur à 10 ms.

Comme le G.728, notre codeur appartient à la classe des codeurs qui utilisent *l'analyse par synthèse dans la prédiction linéaire* (LPAS). Ils sont aussi appelés codeurs *hybrides*.

Dans le codage LPAS, le signal de parole d'entrée est analysé pour chaque trame et le signal d'excitation est déterminé pour chaque sous-trame parole. Le signal d'excitation est filtré à travers le filtre de synthèse pour produire le signal synthétique. Le signal original est soustrait du signal reconstruit et l'erreur de codage est minimisée en utilisant un critère de pondération quadratique moyenne. Le signal d'excitation qui minimise l'énergie de l'erreur est sélectionné et les paramètres correspondants sont transmis au récepteur.

Le récepteur utilise la même structure de synthèse pour reconstruire le signal synthétique. Puisque la procédure de recherche de la meilleure séquence d'excitation au codeur exige le calcul du signal synthétique, ce type de codage est appelé codage analyse par synthèse.

Ce mémoire est constitué de cinq chapitres :

Le **premier** chapitre comporte des généralités sur le modèle de production de la parole humaine, les propriétés du signal de parole ainsi que quelques notions sur la mesure de la qualité du signal de parole reconstruit. Le principe de la quantification vectorielle est également abordé dans ce chapitre.

Le **deuxième** chapitre est consacré à l'analyse par prédiction linéaire et la définition des codeurs CELP (Code Excited Linear Prediction) qui utilisent cette technique.

Le **troisième** chapitre définit un certain type de codeurs CELP ayant la particularité d'avoir un faible retard ou délai appelé Low Delay Code Excited Linear Prediction ou LD-CELP et opérant à 16 kb/s.

Le **quatrième** chapitre décrit la mise en œuvre de ce Low-Delay CELP à 16 kb/s (norme G.728) avec la description de chaque bloc le constituant.

Le **cinquième** chapitre aborde la mise en œuvre de notre codeur LD-CELP 8 kb/s.

Chapitre 1

Généralités

Ce chapitre comporte des généralités qui regroupent des notions de production, d'audition et les propriétés statistiques d'un signal de parole. Nous donnons un aperçu sur les systèmes de communication, les dimensions de performance des codeurs et les mesures objectives et subjectives utilisées pour l'évaluation de la qualité du signal de parole synthétisé. Ensuite, nous abordons un aspect très important dans toute opération de codage qui est la quantification vectorielle. Des définitions générales et des notions de base de la quantification vectorielle seront données. Nous décrivons ensuite l'algorithme de Lloyd généralisé pour la conception de dictionnaire stochastique et une technique de conception de dictionnaire initial.

1.1 Mécanisme de la phonation

La parole est le résultat de l'action volontaire et coordonnée des appareils respiratoires et masticatoires. Cette action se déroule sous le contrôle du système nerveux central qui reçoit en permanence des informations par rétroaction auditive et par les sensations cénesthésiques.

L'appareil respiratoire fournit l'énergie nécessaire lorsque l'air est expiré par la *trachée-artère*. Au sommet de celle-ci se trouve le *larynx* où la pression de l'air est modulée avant d'être appliquée au *conduit vocal*, qui s'étend du *pharyn*x jusqu'au *lèvres* (fig. 1.1).

Le larynx est un ensemble de muscles et de cartilages mobiles qui entourent une cavité située à la partie supérieure de la trachée (fig. 1.2(a)). Les cordes vocales sont en fait deux lèvres symétriques placées en travers du larynx ; ces lèvres peuvent fermer complètement le larynx et, en s'écartant, déterminer une ouverture triangulaire appelée *glotte* (fig. 1.2(b)). L'air y passe librement pendant la respiration et la *voix chuchotée*, et aussi pendant la phonation de *sons sourds* ou *non voisés*. Les sons voisés résultent au contraire d'une vibration périodique des cordes vocales ; des impulsions périodiques de pression sont ainsi appliquées au conduit vocal.

Le conduit vocal peut être considéré comme une succession de tubes ou cavités acoustiques de sections diverses.

Les sons non voisés, fricatifs ou plosifs, sont générés en laissant volontairement les cordes vocales ouvertes. Pour les fricatives (/f/, /s/, /ch/), le débit d'air, forcé à travers un

resserrement du conduit vocal, donnera naissance à un bruit. Un resserrement lié aux vibrations des cordes vocales générera des fricatives voisées (e. g. /v/). Un son plosif (ou occlusif) est produit par une occlusion momentanée du conduit vocal en un point donné suivi par une ouverture brusque; Il peut être voisé (/b/, /d/, /g/) ou non voisé (/p/, /t/, /k/) [3] et [4].

Les sons voisés résultent donc de l'excitation du conduit vocal par des impulsions périodiques de pression liées aux oscillations des cordes vocales : l'ouverture brusque de la glotte libère la pression accumulée en amont ; elle se referme ensuite plus graduellement.

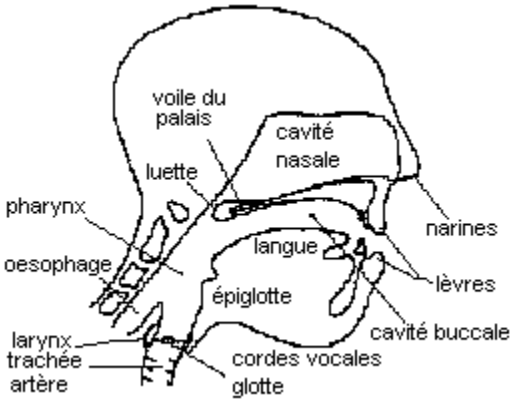


Fig 1.1 Appareil phonatoire

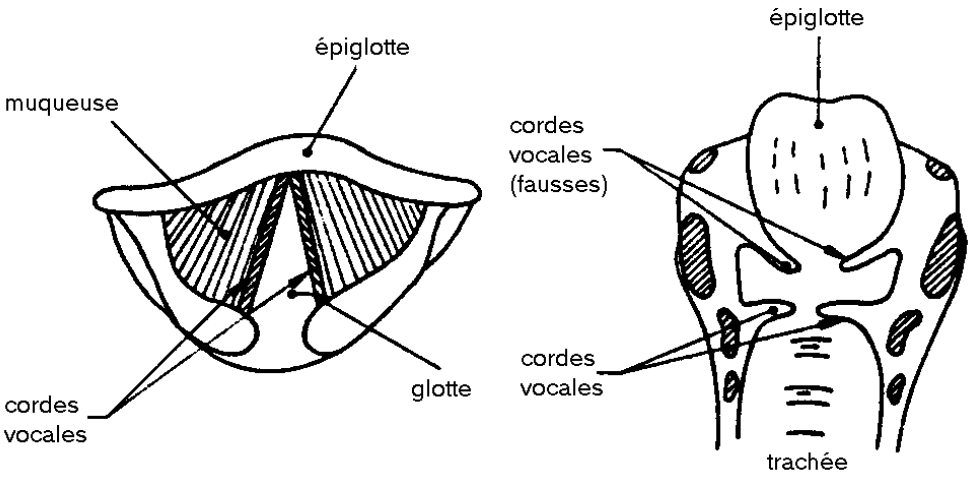


Fig 1.2 Vues du larynx : (a) vue de haut ; (b) coupe verticale

Le signal parole peut donc être divisé en deux catégories : **voisé** et **non voisé.** Pour la parole voisée, l'excitation est générée par l'ouverture et la fermeture périodique de la glotte générant ainsi des impulsions périodiques. Pour la parole non voisée, l'excitation glottale est un bruit à spectre plat qui a souvent comme modèle un générateur de bruit aléatoire. Un modèle général et simplifié de production de la parole est donnée à la figure 1.3 [5]. La proportion relative des composantes voisées et non voisées est contrôlée dans le modèle en ajustant les gains correspondants. Pour un signal purement voisé le gain de la source de bruit est nul et pour un signal purement non voisé le gain de la source voisée est égal à zéro.

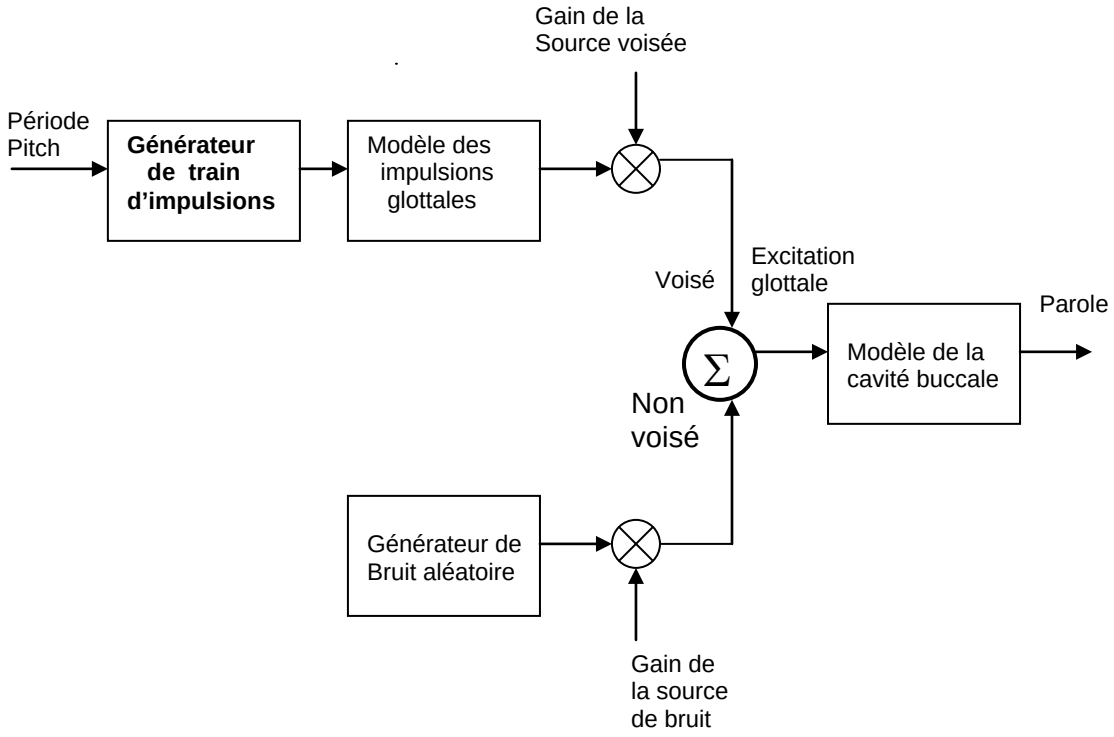


Fig 1.3 Modèle général de production de la parole.

1.2 Spectre du signal de parole

1.2.1 La source

Le signal de source est l'onde que l'on obtient à la sortie de la glotte. Il est appelé aussi *onde glottique.* Il peut être périodique ou non.

Lorsque l'onde glottique est périodique, elle peut être développée en série de Fourier. La première raie sera la fréquence fondamentale F_0 ; les autres raies, espacées de F_0 sont les harmoniques. Ce train d'impulsions est très étendu en fréquence, son amplitude présente une décroissance de 10 à 15 dB/octave selon le locuteur [6].

1.2.2 Cavités supra glottiques

L'ensemble des cavités supra glottiques constitue un filtre assimilable à un ensemble de résonateurs qui renforcent ou atténuent sélectivement les composantes du spectre du signal de source, i.e l'onde glottique. Les résonances sont appelées *formants*. Ces formants sont modifiés en fonction de la position des articulateurs (langue, voile du palais et lèvres). Un son est donc le résultat du filtrage de l'onde glottique par les cavités supra glottique.

On observe généralement dans le spectre du son entre 0 et 4 kHz trois formants principaux notés F_1, F_2, F_3. L'enveloppe spectrale du signal de parole présente une décroissance moyenne de 6 dB par octave.

La puissance spectrale court-terme d'un segment de parole calculée sur une fenêtre de 30 ms montre les caractéristiques de base du signal de parole (fig. 1.4). Nous pouvons identifier la structure spectrale fine due à l'excitation glottale et l'enveloppe spectrale imposée par le conduit vocal. Pour la parole voisée, nous pouvons observer dans la structure fine des harmoniques régulièrement espacées qui sont le résultat des oscillations périodiques des cordes vocales. Par contre pour la parole non voisée, la structure fine ne présente pas d'harmonique. L'excitation non voisée est telle un bruit (pas de périodicité apparente). Les pics prononcés de l'enveloppe spectrale correspondent aux résonances du tube acoustique. Les résonances sont appelées *formants* et le conduit vocal impose une structure formantique à l'excitation glottale.

La structure fine du spectre établit la corrélation long-terme des échantillons du signal dans le domaine temporel. Dans la parole voisée, la structure spectrale harmonique indique une similarité des cycles séquentiels de la période *pitch*. La fréquence pitch ou fréquence du *fondamental* peut varier :

- de 80 à 200 Hz pour une voix masculine,
- de 150 à 450 Hz pour une voix féminine,
- de 200 à 600 Hz pour une voix d'enfant.

Pour la parole non voisée, cette corrélation long-terme est minime ou inexistante. L'enveloppe spectrale correspond aux corrélations court-terme entre les échantillons proches.

Ces deux types de corrélation (long-terme et court-terme) sont importants et sont exploités dans les codeurs de parole.

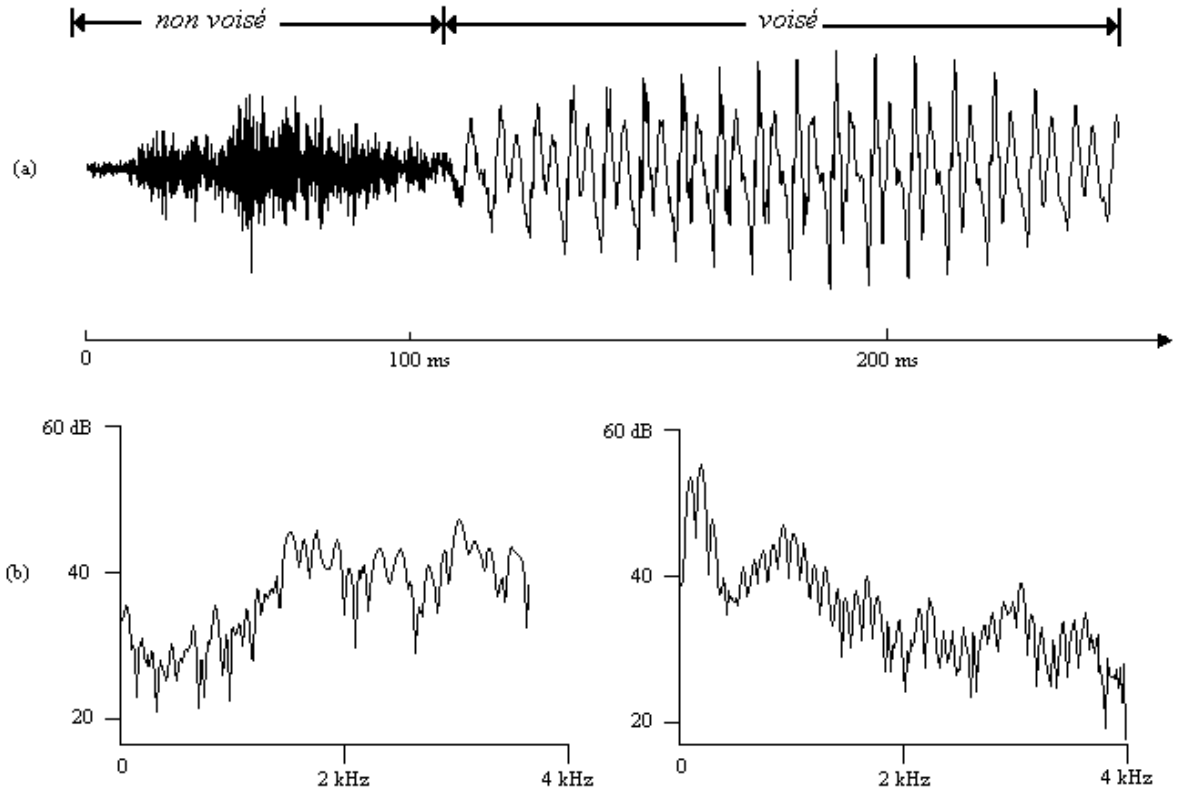


Fig 1.4 (a) Segments de parole non voisée et voisée. (b) Puissances spectrales calculées dans les régions non voisées et voisées respectivement. La puissance spectrale est calculée sur des segments de 30 ms et pondérés par une fenêtre de Hamming.

La redondance dans le signal de parole conduit à la conclusion que les échantillons de parole sont corrélés. L'enveloppe spectrale correspond aux corrélations court terme et la structure harmonique correspond aux corrélations à long-terme.

1.3 Les redondances dans le signal parole

Le signal vocal est caractérisé par une très grande redondance, condition nécessaire pour résister aux perturbations du milieux ambiant.

On peut tenter de déterminer la cadence maximale à laquelle un auditeur peut assimiler un message. On va dans ce but définir l'information associée à un message constitué par des éléments discrets X_i appartenant à un ensemble donné X. Si $p(X_i)$ est la probabilité a priori d'occurrences du symbole X_i, sa sélection apporte une information [3] :

$$I = -\log_2 p(x_i) \tag{1.1}$$

L'information moyenne associée à l'occurrence du message $X=[X_1, X_2, \ldots, X_l]$ vaut :

$$H(X) = -\sum_i p(x_i) \log_2 p(x_i) \tag{1.2}$$

C'est l'**entropie** de la source, exprimée en bits.

Ainsi, si X est l'ensemble des 42 phonèmes de la langue anglaise dont les probabilités d'occurrence sont connues, on trouve la valeur moyenne H=4.9 bits ; si tous les phonèmes étaient équiprobables, on aurait trouvé 5.39 bits (car $2^{5.39}$ =42). En d'autres termes, chaque phonème doit être codé avec 5 bits si l'on tient compte des probabilités a priori, sinon avec 6 bits.

Pour la langue française, qui comporte 36 phonèmes, on a trouvé H=4.37, valeur à comparer avec 5.17 ($2^{5.17} = 36$).

Dans la conversation courante, environ dix phonèmes sont prononcés par seconde ; l'information moyenne est donc inférieure à 50 bits/s.

Or on sait que pour un canal continu sans erreurs, le débit maximum d'information est donné par :

$$C = B \log_2[1 + S/N] \text{ bits/s}. \tag{1.3}$$

si B est la largeur de la bande passante en Hz et S/N est le rapport signal à bruit en dB.
Pour un canal téléphonique avec B=3000 Hz et S/N =30 dB, on trouve C=30000 bits/s. Il y a donc apparemment une redondance énorme dans ce canal. En réalité, il y a beaucoup plus d'information dans un message téléphonique que dans un message écrit où elle se réduit à l'intelligibilité [3].

La suppression partielle des redondances permet une représentation plus efficace des données. La compression des données peut se faire sans pertes d'informations (e.g. le codage de Huffmann) ou avec perte d'information en exploitant dans ce cas la tolérance de l'organe récepteur (e.g. l'oreille). La compression du signal de parole consistera à réduire les redondances qui sont essentiellement dues à [7] :

- le manque de platitude du spectre court terme ;
- la quasi périodicité des signaux voisés ;
- la limitation des formes et des vitesses de mouvement possibles du conduit vocal ;
- les distributions non uniformes des valeurs des paramètres de transmission.

Les trois premières sont dues à des propriétés physiques du mécanisme de production de la parole. La dernière est fonction du codage utilisé.

Le manque de platitude du spectre court-terme est lié au fait que les échantillons de parole adjacents sont corrélés entre eux. On peut décorréler ces échantillons de parole par un filtrage spectral adapté. La quasi-périodicité des signaux parole voisée peut être supprimée en utilisant un prédicteur long-terme. La lenteur du conduit vocal permet d'envoyer les paramètres des filtres toutes les 10-30 ms. La dernière des redondances citées peut être exploitée par un codage approprié.

1.4 Propriétés statistiques du signal parole

Un signal vocal est une réalisation particulière d'un processus aléatoire non stationnaire ; ses propriétés statistiques moyennes doivent donc être estimées sur des intervalles de temps importants, par exemple sur plusieurs dizaines de secondes, et moyennées pour plusieurs locuteurs. On parle dans ce cas de statistiques à long terme. On définit également une statistique à court terme, estimée sur des tranches temporelles de durée de 10 à 30 ms pendant lesquelles le signal est quasi-stationnaire. On pourra donc appliquer pour le traitement des signaux de parole les méthodes classiques du traitement du signal en prenant certaines précautions et toujours au prix de certains compromis.

Les principales caractéristiques des signaux de parole sont [3] :

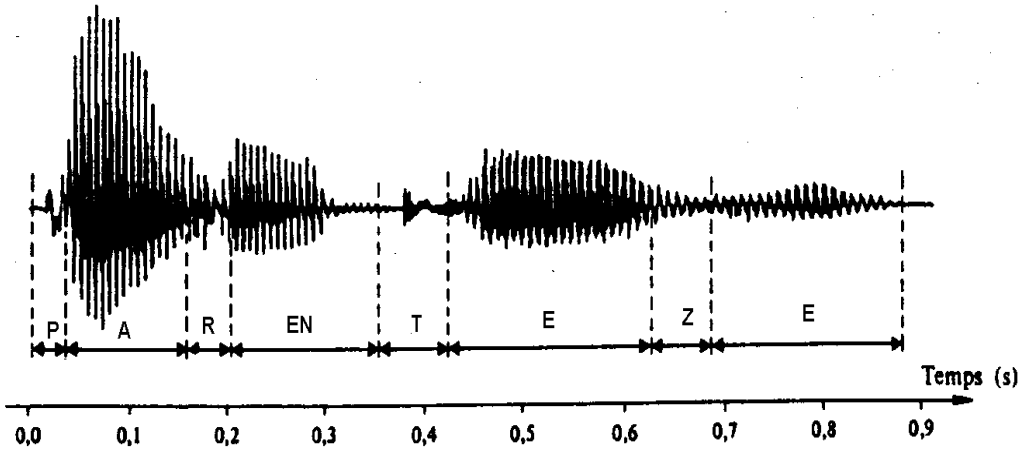
- La *densité de probabilité* pour un signal de parole est proche d'une répartition gamma ;
- Le *taux de passage par zéro*, c'est-à-dire le nombre d'échantillons successifs de signes opposés présente une répartition sensiblement gaussienne et une moyenne d'environ 49 pour les sons non-voisés et 14 pour les sons voisés (pour des fenêtres de 10 ms) ;
- En estimant la fonction d'autocorrélation sur un fenêtre de N échantillons par

$$\Phi_{xx}(k) = \frac{1}{N-k}\sum_{n=0}^{N-k} x(n)x(n+k) \qquad (1.4)$$

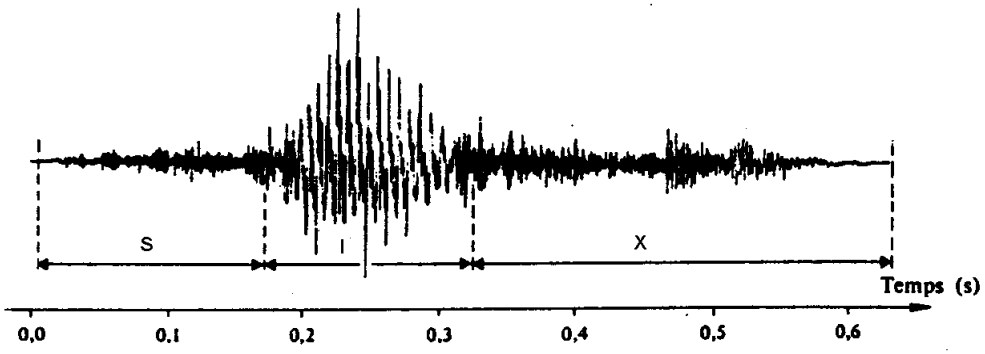
Le coefficient d'autocorrélation $\rho_{xx}(k) = \Phi_{xx}(k)/\Phi_{xx}(0)$ est toujours compris entre –1 et +1 ;

- La densité spectrale de puissance (à court terme) qui est la transformée de Fourrier de la fonction d'autocorrélation présente une structure périodique fine lorsque la tranche est voisée ; elle correspond aux harmoniques de l'excitation glottique. Les maximums de l'enveloppe de ce spectre sont les formants ; ils correspondent aux résonances du conduit vocal. Par contre, le spectre d'un signal non voisé ne présente aucune structure particuliére, sauf une accentuation vers les hautes fréquences.

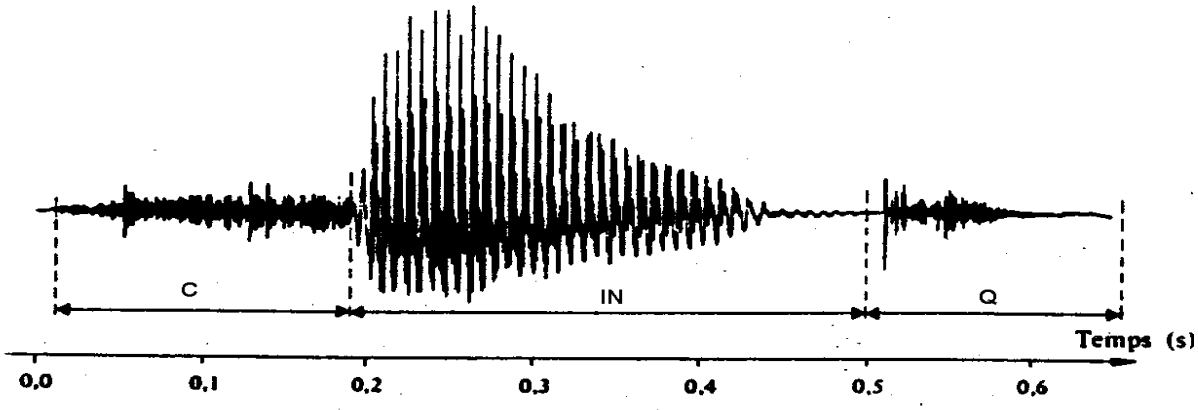
La figure 1.5 représente l'évolution temporelle du signal vocal pour les mots *parenthèse*, *six* et *cinq* [3]. Une représentation plus fine d'une tranche de signal voisé et non voisé est donnée à la figure 1.6. On voit que la forme d'onde voisée à basse fréquence varie plus lentement que la haute fréquence de la forme d'onde non voisée. On notera que le segment voisé est quasi-périodique, avec environ "6" périodes sur 50 ms, ce qui correspond dans ce cas, à une fréquence pitch de 120 Hz.



Mot ‘’Parenthèse’’

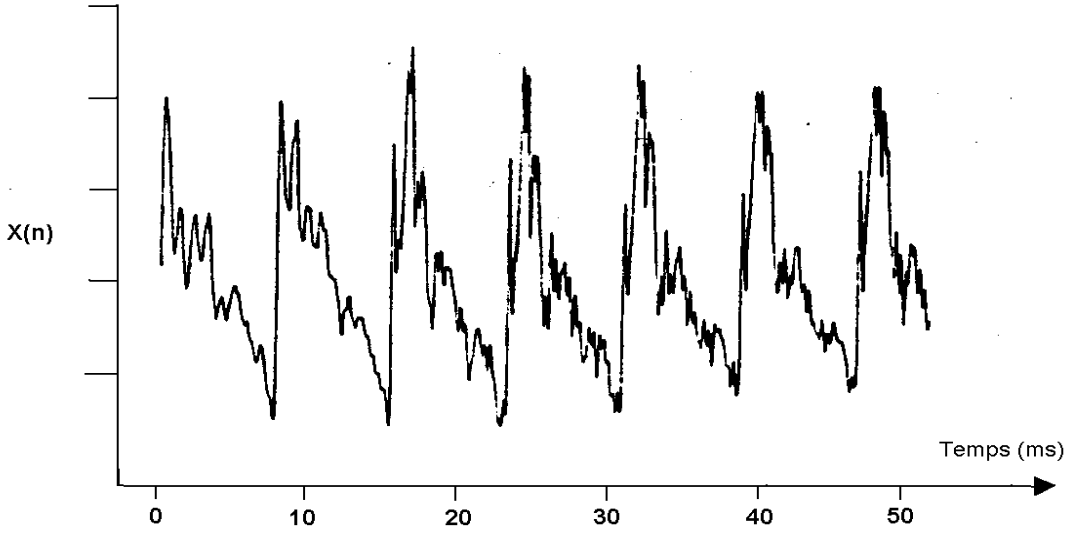


Mot ‘’Six’’

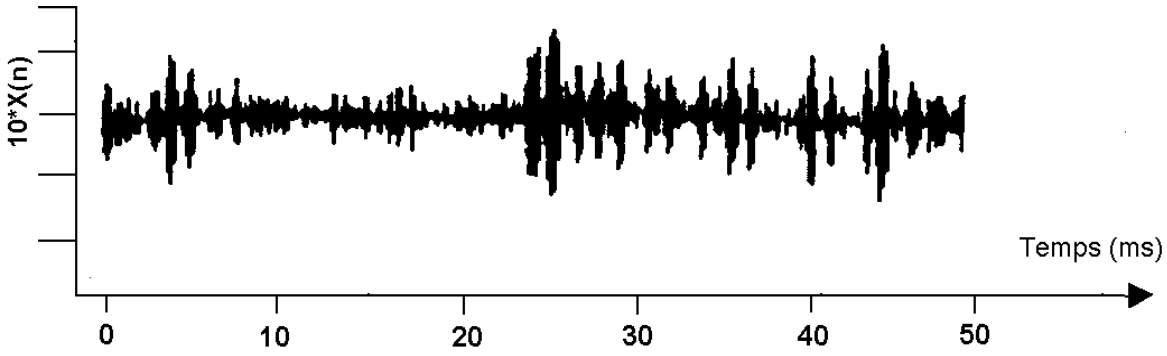


Mot "Cinq"

Fig 1.5 Représentation temporelle d’un mot d’après [3].



(a) Tranche voisée



(b) Tranche non voisée

Fig 1.6 Exemple de tranche voisée (a) et non voisée (b).

La forme d'onde non voisée, d'autre part, est ressemblante à un bruit. Le niveau d'énergie est plus bas que celui du segment voisé. La forme d'onde de la parole est extrêmement non stationnaire avec des différences très significatives entre segment en termes de niveaux d'amplitudes et du contenu spectral. La figure 1.7 montre la dépendance temporelle de la variance de segment $\sigma^2_x(k)$ mesuré à travers des segments de 32 ms de parole. On constate que la dynamique de la variance court-terme peut excéder 40 dB sur une durée de moins de 1 seconde.

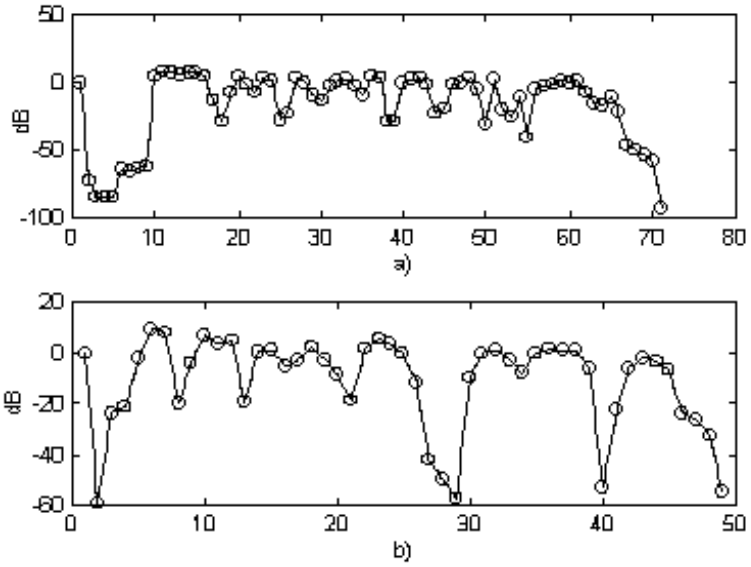


Fig 1.7 Evolution de la variance de la parole court-terme a) signal de 2.24 s et b) signal parole différent de 1.60 s.

1.5 Classification des codeurs de la parole

Différents algorithmes de codage de la parole exploitent les propriétés du signal parole à différents degrés. Ils peuvent être classés en trois catégories : *codeurs par formes d'ondes* qui travaillent généralement à des débits élevés et qui n'utilisent pas de modèle sur le signal à traiter. D'autre part, les *vocodeurs* qui travaillent à de faibles débits et qui modélisent le conduit vocal et le signal d'excitation. Enfin Les *codeurs hybrides*, qui comme leurs noms l'indiquent se positionnent entre les deux classes précédentes[8].

1.5.1 Codeurs par forme d'ondes

Ces codeurs sont utilisés pour une représentation fidèle de l'onde dans le domaine temporel. Ils minimisent la différence entre le signal original et le signal reconstruit. Les codeurs par forme d'ondes n'exploitent généralement pas les caractéristiques du signal d'entrée donc ils n'utilisent aucune connaissance à priori sur la façon dont le signal est généré. Cependant, ils sont robustes car pouvant être utilisés pour coder n'importe que type de son. Plus le débit augmente et plus le signal de sortie (signal reconstruit) ressemble au signal original. D'autre part, pour un débit inférieur à 2 bits/échantillon, la qualité perceptuelle se dégrade. Les codeurs par formes d'ondes sont utilisés aussi bien dans le domaine temporel que dans le domaine fréquentiel.

1.5.2 Vocodeurs

Les vocodeurs (voice coders) appartiennent à la classe des codeurs paramétriques. Ces derniers utilisent une méthode dite *analyse par synthèse* où l'on essaie d'extraire du signal un ensemble de paramètres liés à un modèle simplifié. La connaissance de la fonction de transfert du système de synthèse et de son excitation suffit pour reproduire le signal de sortie. Le système de synthèse et le signal d'excitation sont donc définis par un ensemble de paramètres qui seront codés pour la transmission. Pour un signal localement stationnaire comme la parole, le modèle et les paramètres de l'excitation peuvent être représentés de manière compacte ; ceci étant l'intérêt principale du codage de la source. Ces paramètres sont l'enveloppe du spectre court terme et les informations sur le signal d'excitation (amplitude, pitch). Ces codeurs sont sensibles aux bruits de transmission et la qualité de la parole est limitée. Ils sont souvent inefficaces pour le codage de signaux autre que la parole et leur débit est généralement faible.

1.5.3 Codeurs hybrides

Les codeurs hybrides ressemblent aux précédents codeurs en ce sens qu'il y a une estimation de paramètres d'un modèle de synthèse pour un signal donné (analyse par synthèse) et une utilisation de technique de codage par formes d'ondes pour le signal d'excitation. Ces codeurs sont généralement plus complexes à mettre en œuvre que les vocodeurs ou les codeurs par formes d'ondes mais ils permettent d'obtenir une bonne qualité de signal à des débits intermédiaires. Considérant la demande de réduction du débit et l'amélioration de la qualité perceptuelle de la parole synthétisée, les générations présentes de codeurs appartiennent à cette catégorie.

1.6 Systèmes de transmission

Le codage d'un signal est la procédure de représenter un signal ''information'' de façon que l'objectif de communication soit réalisé telle une conversion analogique numérique, transmission à bas débit ou cryptage de message. Dans la littérature les termes codage de source, codage numérique, compression de données et compression des signaux sont tous utilisés pour désigner les techniques utilisées pour achever une représentation numérique compacte du signal. La figure (1.8) représente un schéma bloc de codage numérique. On notera que le but final est le récepteur humain [9].

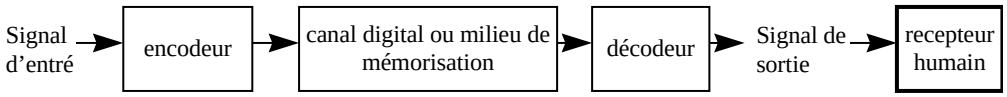


Fig 1.8 Codage numérique pour compression de signaux .

1.6.1 Chaîne de communication numérique

La figure (1.9) décrit les différents blocs d'une chaîne de communication numérique :

- le codeur source essaie de minimiser le débit binaire nécessaire pour représenter fidèlement le signal d'entrée ;
- le MOdulateur DEModulateur (MODEM) cherche à maximiser le débit binaire que peut supporter un canal donné sans causer un niveau inacceptable de probabilité d'erreur binaire ;

- Le bloc de codage canal ajoute des redondances au flux binaire pour la protection contre les erreurs.

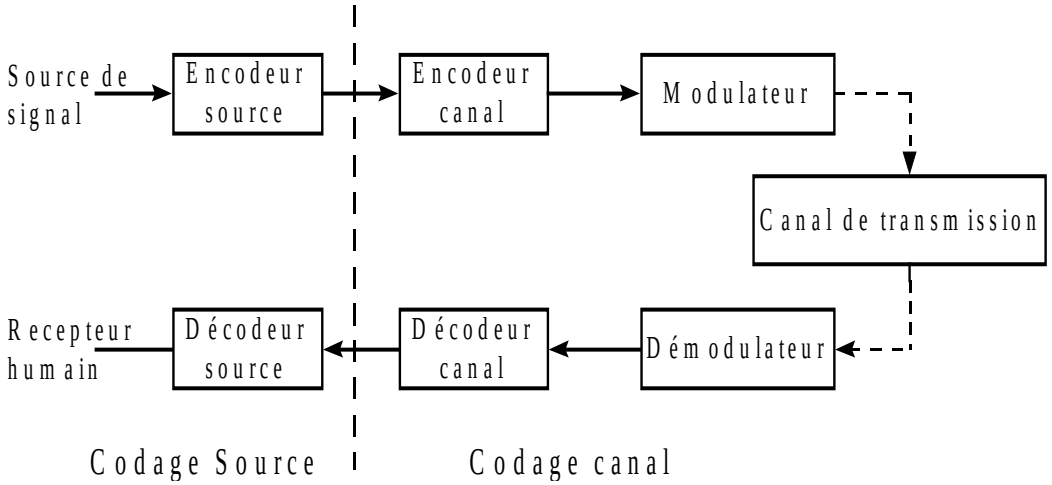


Fig 1.9 Schéma bloc d'un système de communication numérique [9].

La capacité de la compression des signaux a été une majeure préoccupation de la technologie de la communication long-distance, stockage de haute qualité et le cryptage de message.

1.6.2 Attributs du codeur de parole

La qualité de la parole produite par un codeur de parole est fonction de quatre (04) paramètres : le débit, la complexité, le retard (ou délai), et la bande de travail. Ainsi, en étudiant les codeurs de parole il est important d'examiner tous ces attributs.

1.6.2.1 *Qualité du signal*

La qualité du signal perçu est souvent évaluée sur une échelle de 5 points qui est connue comme étant l'échelle **MOS** (**M**ean **O**pinion **S**core) dans les tests de la qualité de la parole : une moyenne à travers un grand nombre d'entrée parole, locuteurs et testeurs d'écoute évaluant la qualité du signal. Les cinq points de la qualité sont associés à un ensemble d'adjectifs de description : mauvais, médiocre, inacceptable, bon, excellent. On attribue ainsi un seul niveau à chaque signal parole à évaluer durant la procédure d'évaluation subjective.

1.6.2.2 *Débit binaire*

On mesure le débit binaire d'une représentation digitale en bits par échantillon, ou bit par seconde (b/s) selon le contexte. Le débit en bits par seconde n'est que le produit de la fréquence d'échantillonnage et du nombre de bits par échantillon. La fréquence d'échantillonnage doit être au moins deux fois plus grande que la largeur de bande du signal correspondant (Théorème de Shannon). Dans le cas de la téléphonie, pour une bande de 3.2 kHz (200-3400 Hz), la fréquence d'échantillonnage de 8 kHz est utilisée.

1.6.2.3 *Complexité*

La complexité d'un algorithme de codage est l'effort de calcul exigé pour implanter les processus de l'encodage et du décodage dans les cartes de traitement du signal (hardware), mesuré en terme de la capacité arithmétique et de l'espace mémoire utilisé. D'autres mesures de complexité peuvent être signalées telles que la taille physique de l'encodeur ou du décodeur ou codec, le prix et la consommation de puissance (en Watt ou en milliWatt, mW) ce dernier étant un important critère dans un système portable.

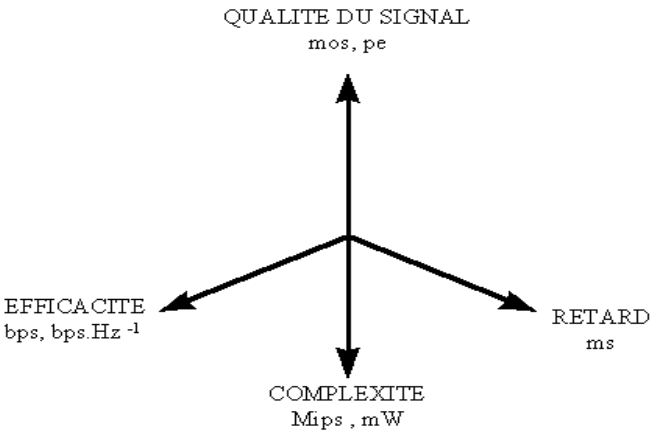


Fig 1.10 Critères de performance d'un codeur [9].

1.6.2.4 *Retard de communication*

La complexité dans un algorithme de codage est souvent accompagnée d'une augmentation de la durée de traitement dans l'encodeur et le décodeur. Bien que l'évolution des capacités des processeurs de traitement du signal est un facteur en faveur d'utilisation d'algorithme plus sophistiqué, le besoin de limiter le retard de communication ne doit pas être d'une importance moindre. Ce besoin impose des restrictions pratiques importantes dans l'utilisation des algorithmes. Selon l'environnement de communication, le retard total permis à un sens peut être aussi bas qu'une milliseconde (comme en réseaux téléphoniques sans annulateur d'écho).

Le retard de codage à un seul sens est défini comme étant le temps écoulé entre l'instant où l'échantillon du signal de parole arrive à l'entrée de l'encodeur et l'instant où le même échantillon apparaît à la sortie du décodeur, moins tout retard introduit par les autres équipements de communication (comme les MODEM) entre la paire encodeur-décodeur et le retard de propagation du signal qui dépend de la distance. En d'autres termes, c'est comme si l'encodeur et le décodeur sont directement connectés par fils sans aucun équipement entre eux. Cette définition fait que le retard de codage dépend seulement de l'algorithme de codage.

Avec cette définition, le retard de codage des codeurs CELP peut être grossièrement déterminé en fonction de la taille de la trame du signal de parole utilisée.

Le retard de codage consiste en trois catégories de retard [10] :

- retard algorithmique de bufferisation ;
- retard de traitement ;
- retard de transmission binaire.

- Le premier type de retard est dû à l'analyse LPC (Linear Prediction Coding) adaptative progressive. Le codeur CELP doit bufferiser une trame d'échantillon avant de commencer le codage du premier échantillon de cette trame. Cette opération introduit au moins une trame de retard de bufferisation.

- En supposant que le hardware utilisé est assez rapide pour exécuter le codage en temps réel (ce qui est le cas en général -cartes TMS320C30- par exemple.), alors, cela peut prendre presque une valeur d'une trame de retard de traitement pour achever l'encodage et décodage de la trame de parole bufferisée.

- Il est supposé que l'encodeur ne commence l'envoi de bits correspondant à une trame donnée que si l'encodage de la trame entière soit terminé. Le décodeur ne commence le décodage que si tous les bits de cette trame sont reçus, d'où, une trame additionnelle de retard de transmission binaire doit être introduite. Ceci est dû au fait que le temps mis entre l'envoi du 1^{er} bit de la trame et la réception du dernier bit de la trame est égal à une trame, étant donné que le débit binaire du canal de communication est le même que le débit binaire du codeur de la parole. Alors, le retard de codage total (à un sens) du codeur CELP vaut environ trois trames.

En pratique, on peut réduire le retard de traitement en utilisant des processeurs plus rapides.

Si on prend un retard de 2.5 à 3 trames comme moyenne, alors le codeur CELP, avec des trames de 20 ms chacune, aura un retard de codage global de 50 à 60 ms.

1.7 Mesure de la qualité

On distingue les mesures objectives de qualité qui sont des mesures purement mathématiques des mesures subjectives de qualité qui évaluent la qualité du codage à l'écoute.

1.7.1 Mesure objective de la qualité

Les Rapports Signal à Bruit (RSB) et Signal à Bruit segmental (RSB_{seg}) sont les mesures les plus couramment utilisées dans le domaine temporel de la distorsion entre le signal de parole original et son synthétique correspondant.

Le Rapport Signal à Bruit (RSB) peut être définit pour un enregistrement de N échantillons comme étant le rapport entre la puissance du signal d'entrée et la puissance du bruit et donné en décibels (dB) par :

$$RSB = 10\log\left(\frac{E_s}{E_e}\right) = 10\log\frac{\sum_{n=0}^{N-1} s^2[n]}{\sum_{n=0}^{N-1}(s[n]-\hat{s}[n])^2} \text{ dB,} \qquad (1.5)$$

où s est le signal de parole original et $\hat{s}$ le signal de parole synthétisé. L'avantage principal de l'utilisation du SNR comme mesure de qualité réside dans sa simplicité de calcul. La mesure représente une erreur moyenne dans le domaine temporel et fréquentiel pour un signal traité [11].

Le signal de parole étant par nature non-stationnaire, certains segments du signal peuvent avoir une énergie plus ou moins grande. En supposant que l'énergie de l'erreur soit à peu près constante, le RSB pourra être très important comme très faible. On utilise plutôt le RSB segmental. Le signal est découpé en M segments de 15 à 30 ms puis on calcule une moyenne des RSB.

$$RSB_{seg} = \frac{1}{M}\sum_{i=0}^{M} 10\log\left(\frac{\sum_{n=0}^{N-1} s^2[n]}{\sum_{n=0}^{N-1}(s[n]-\hat{s}[n])^2}\right) \text{ dB,} \qquad (1.6)$$

Cette mesure présente l'avantage de tenir compte de l'évolution du RSB au cours du temps et, en particulier, de bien prendre en compte les segments de faible énergie et de limiter en outre les trop grands écarts. Si un segment possède un RSB supérieur à 35 dB, on le remplace par 35 dB. De même dans les zones de silence, le RSB peut atteindre des valeurs très négatives : dans ce cas, on peut ou bien retirer du calcul ces zones ou bien fixer un seuil inférieur S tel que $0 \le S \le -10$ dB.

L'exemple suivant nous donne une idée sur le fait que le RSB segmental reflète mieux la réalité de distorsion du signal [12]. Soit les deux tableaux de valeurs suivant :

Numéro du segment	1	2
RMSdu signal d'entrée	10	1
RMS du bruit d'entrée	1	1
RSB(m)	100	1
RSB(m) dB	20	0

Rapport signal à bruit

RSB conventionnel	(100 + 1) / 2 = 50.5 $\rightarrow$ 17 dB
RSB segmental	(20 dB + 0 dB) / 2 $\rightarrow$ 10 dB

Tableau 1.1 RSB segmental et RSB conventionnel d'après [7].

La valeur RSB conventionnel = 17 dB est très influencée par RSB(1) = 20 dB, correspondant au segment de haute énergie. Alors que, RSB seg = 10 dB, montre une pondération égale des valeurs des composantes RSB(1) = 20 dB et RSB(2) = 0 dB.

1.7.2 Mesures subjectives de la qualité

Les essais d'écoute sont nécessaires car la qualité d'un système de codage de la parole ne vaut que par le jugement humain. De plus, le RSB n'est pas nécessairement corrélé avec la qualité d'écoute [6].

Les méthodes les plus utilisées sont :

- *Diagnostic Rhyme Test* (DRT) qui mesure l'intelligibilité sur un grand nombre de mots.
- *Diagnostic Acceptability Mesure* (DAM) qui mesure le naturel perçu de la parole.
- *Mean Opinion score* (MOS) où l'auditeur évalue un codeur sur une échelle absolue allant de 1 à 5 avec

5. Excellent
4. Bon
3. Passable
2. Médiocre
1. Mauvais

1.7.3 Scores

Nous donnons quelques scores obtenus pour les codeurs les plus courants (pour une bande passante téléphonique de 3.4 kHz) [6] :

Codeur	Débit en kbps	Norme	DRT	DAM	MOS
PCM log	64	G711	95	73	4.3
ADPCM	32	G721	94	68	4.1
LD-CELP	16	G728	94	70	4.0
CELP	8		93	68	3.7
CELP	4.8	FS-1016	93	67	3.0
LPC	2.4	OTAN	90	54	2.5

Tableau 1.2 Scores pour les codeurs les plus courants.

La figure ci-dessous représente la **qualité** en fonction du **débit** :

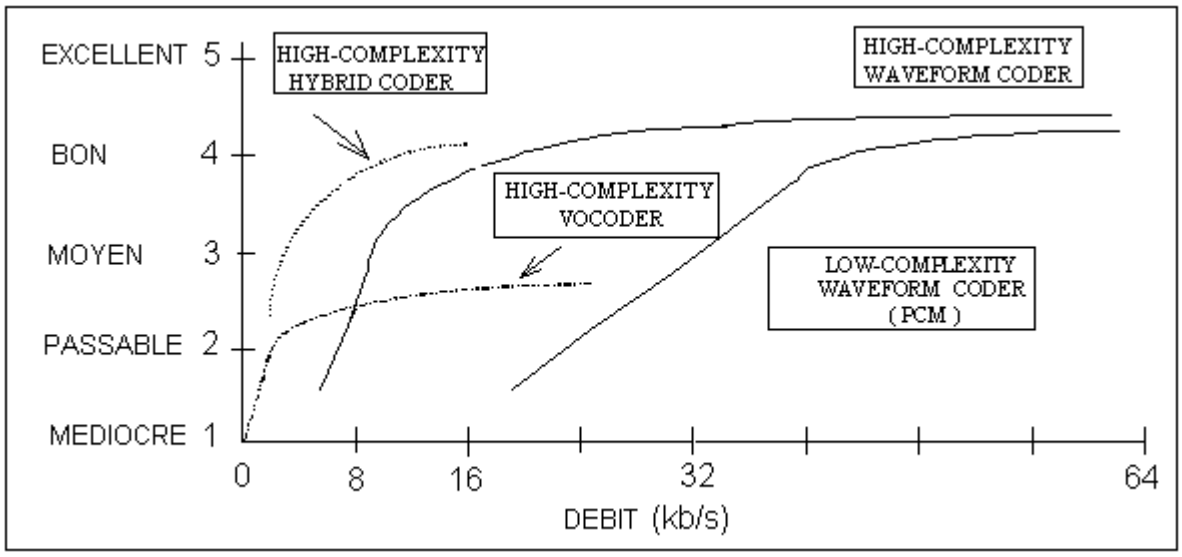


Fig 1.11 Relation entre le débit de codage et la qualité de parole [9].

1.8 La Quantification Vectorielle

1.8.1 Introduction

La numérisation demeure la méthode la plus simple de codage d'un signal. Cette technique consiste à coder le signal échantillon par échantillon, à l'aide d'un quantificateur scalaire. L'idée de la *quantification vectorielle* est de segmenter le signal en blocs d'échantillons. Ces blocs que l'on nomme vecteurs, seront codés par d'autres vecteurs pré-définis, choisis dans un catalogue couramment appelé *dictionnaire ou répertoire*.

Cette technique qui a fait son apparition dans les années 50, ne sera pas réellement mise en œuvre efficacement avant les travaux de Kang et Coulter [13] en 1976 et ceux de Buzo & al. En 1980.

Ce sont les vocodeurs qui utiliseront les premiers la quantification vectorielle pour réduire le débit binaire aux environ de 2400 bps, mais c'est le codeur CELP d'Atal associant les techniques de prédiction linéaire, d'analyse par synthèse et de la quantification vectorielle, qui ouvre la porte aux principales voies de recherche actuelles. Cette méthode de codage de la parole a montré et montre toujours son efficacité [14].

1.8.2 Principe de la Quantification Vectorielle

La quantification est une partie intégrante dans tout codeur de la parole. Elle consiste en l'approximation d'un signal continu par un signal d'amplitude discrète. La plupart des paramètres utilisés pour représenter le signal de parole doivent être quantifiés avant l'encodage. Si, à chaque instant, une seule valeur est quantifiée, on parlera de quantification scalaire. La valeur est représentée par l'une des valeurs discrètes fixes dites niveaux de quantification. Dans la quantification scalaire uniforme, ces niveaux sont régulièrement espacés. Dans la quantification logarithmique, l'espacement est uniforme dans le domaine logarithmique. Dans le cas où la grandeur à quantifier est composée de plusieurs variables : un vecteur, on parle alors de **Quantification Vectorielle** (notée **QV**) [15,16]. Elle consiste alors à représenter tout vecteur **x** de dimension k par un autre vecteur **yi** de même dimension mais ce dernier appartenant à un ensemble fini *D* de *L* vecteurs. Les **yi** sont appelés les **vecteurs représentants**, les **vecteurs de reproduction** ou les **code vecteurs**. *D* est appelé le **dictionnaire** ou le **catalogue des formes** .

Il n'y a rien de mystérieux à considérer des espaces de grandes dimensions. Pour

mieux appréhender les raisonnements il suffit de savoir que tout s'organise autour des coordonnées des vecteurs et qu'il n'y a pas lieu de s'imposer une représentation mentale géométrique [17]. Pour l'illustrer nous précisons qu'un vecteur **x** de l'espace R^k est simplement une matrice colonne constituée de k nombres réels xi : $\mathbf{x}=(x_1,x_2,\ldots,x_k)^T$, et que par exemple, une sphère entièrement caractérisée par son centre $\mathbf{u} = (u_1,u_2,\ldots,u_k)^T$ et son rayon ρ est constituée de points dont les cordonnées satisfont : $\sum_{i=1}^{k}(x_i - u_i)^2 = \rho^2$

Un quantificateur vectoriel de dimension k et de taille *L* peut être défini mathématiquement comme une application *Q* de R^k vers *D* :

$$Q : R^k \mapsto D$$

$$\mathbf{x} \quad Q(\mathbf{x}) = y_i$$

avec $D = \{ y_i \in R^k / \ i=1,2,\ldots, L\}$

Cette application *Q* détermine implicitement une partition de l'espace source R^k en *L* régions C_i. Ces régions encore appelées **classes** ou **régions de Voronoï** sont déterminées par :

$$C_i = \{ x \quad R^k / Q(x) = y_i \}$$

Nous appellerons distance entre x et y_i = Q (x), généralement notée par $d(x_i\ y_i)$ le degré de distorsion dû à l'approximation du vecteur x par le vecteur **arrondi** y_i. Cette distance doit avoir les propriétés suivantes :

- $d(x,y_i) \geq 0$
- $d(x,y_i) = 0$ si x = y
- $d(x,y_i) = d(y_i,x)$ symétrie

  $d(x,z) \leq d(x,y) + d(y,z)$ inégalité triangulaire.

Dans le cas de la parole, la distance doit avoir deux propriétés supplémentaires :

- d(x,y) a une interprétation physique ;
- d(x,y) est simple à calculer

Exemples de distances avec

$$x = (\alpha_1, \alpha_2, \ldots\ldots \alpha_n)$$
$$y = (\alpha'_1, \alpha'_2, \ldots\ldots \alpha'_n)$$

$$
\begin{aligned}
d(x,y) &= \sum_{i=1}^{n} \left|\alpha_i - \alpha_i^{'}\right| && \text{distance de Minkowsky} \\
d(x,y) &= \left(\sum_{i=1}^{n} \left|\alpha_i - \alpha_i^{'}\right|^2 \right)^{1/2} && \text{distance Euclidienne} \\
d(x,y) &= \underset{i}{Max} \left|\alpha_i - \alpha_i^{'}\right| && \text{distance de Chebychev}
\end{aligned}
\qquad (1.7)
$$

En supposant que la grandeur d'entrée est un vecteur aléatoire distribué selon une loi p(x), les performances du quantificateur peuvent être mesurées par la distorsion moyenne D_Q introduite, c'est à dire par l'espérance mathématique de la distance d :

$$D_Q = E\ \left[d(x,Q(x))\right] = \int d(x,Q(x)).p(x).dx \qquad (1.8)$$

Dans la pratique, la distribution des points d'entrée étant généralement inconnue, on approximera D_Q par une distorsion moyenne calculée sur un large nombre d'échantillons $\{x_1, x_2, .., x_N\}$de vecteurs d'entrée:

$$D_Q \cong \frac{1}{N} \sum_{j=1}^{N} d(x_j, Q(x_j)) \qquad (1.9)$$

La distance introduit implicitement une partition de l'ensemble des vecteurs d'entrée en k classes $\{C_i, i = 0, 1, ..., k - 1\}$, la classe C_i étant l'ensemble des vecteurs associés à y_i par le quantificateur :

$$C_i = Q^{-1}(y_i) = \{x; Q(x) = y_i\} \qquad (1.10)$$

Nous appellerons centroïde de la classe C_i le vecteur c^i tel que sa distance moyenne à tous les éléments de la classe soit minimale (en géométrie euclidienne, le centroïde est le centre de gravité) :

$$E\left[d(x,c_i); x \in C_i\right] = \inf_{x_i} \left\{E\left[d(x,x_i); x \in C_i\right]\right\} \qquad (1.11)$$

Etant donné une distance et une taille de dictionnaire, il existe un quantificateur qui minimise la distorsion moyenne : c'est le quantificateur optimal.

1.8.3 Quantification Vectorielle Optimale

Pour une distribution statistique donnée de la source et un débit fixé :

- Le quantificateur globalement optimal est celui qui minimise la distorsion moyenne;
- Un quantificateur localement optimal a un dictionnaire qui peut être légèrement perturbé sans que la distorsion moyenne n'augmente.

La théorie qui établie la supériorité de la quantification vectorielle sur la quantification scalaire n'est pas constructive, elle ne décrit pas la façon de concevoir le dictionnaire globalement optimal. Seuls des propriétés suffisantes sont connues qui permettent de construire des dictionnaires localement optimaux.
Un quantificateur se décompose en deux applications : un codeur et un décodeur (figure1.12). Le quantificateur (localement) optimal est alors celui réunissant [18] et [19] :

- Le **codeur optimal** (pour un dictionnaire fixé), celui-ci respecte la '' règle du plus proche voisin '' que nous allons décrire en **1.8.4.1** ;
- Le **décodeur optimal** (pour une partition C^i donnée), le vecteur y^i doit minimiser la distorsion associée au voronoï C^i, y^i est donc le centroïde de cette cellule : $y^i=cent(C^i)$; c'est la '' règle du centroïde ''.
- Une troisième condition est nécessaire : il faut que la probabilité d'avoir un vecteur à coder à la même distance de deux représentants soit nulle, sinon ce vecteur source est affecté à l'un des deux représentants, la partition de l'espace n'est plus optimale et donc la condition du décodeur optimal est impossible de satisfaire. Si les vecteurs source sont à amplitude continue, cette troisième condition est toujours vérifiée.

1.8.4 Quantification Vectorielle appliquée au Codage

La quantification Vectorielle offre la combinaison des opérations de codage et de décodage. En effet, considérons l'ensemble I des L index des vecteurs de reproduction y_i du dictionnaire : $I = \{1,2, \ldots, L\}$ [17].
Alors la loi de codage est déterminée par :

$$\begin{array}{ccccc} C & : & R^k & \mapsto & I \\ & & x & & C(x) = I \end{array} \qquad (1.12)$$

Le processus de décodage est lui défini par:

$$D : I \mapsto D$$
$$i \qquad D(i) = y_i \tag{1.13}$$

Un QV se décompose donc en deux applications : un codeur et un décodeur.

1.8.4.1 Le codeur

Le rôle du codeur consiste, pour tout vecteur **x** du signal d'entrée, à rechercher dans le dictionnaire *D* le code-vecteur $\mathbf{y_i}$ le plus proche de **x.**

Nous introduisons ici la définition générale de la **métrique** $L_{p,}$ où la norme d'un vecteur **x** de dimension *k* est :

$$L_p\ (\mathbf{x}) = \|x\|^p = \sum_{j=1}^{k} \left|x_j\right|^p \tag{1.14}$$

La distance entre 2 vecteurs $\mathbf{x_1}$ et $\mathbf{x_2}$ est alors :

$$\left\|x_1 - x_2\right\|^p = d_p\ (\mathbf{x_1}\ ,\ \mathbf{x_2})$$
$$= \sum_{j=1}^{k} \left|x_{1j} - x_{2j}\right|^p \tag{1.15}$$

La notion de proximité que nous avons mise en place est la **distance euclidienne** :

$$D\ (x,y_i) = \sum_{j=1}^{k} \left(x_j - y_{ij}\right)^2 \tag{1.16}$$

C'est donc le cas particulier de L_p ou p=2. Cette distance mesure la distorsion entre les vecteurs **x** et $\mathbf{y_i}$. Les régions de voronoï sont alors données par :

$$C_i = \{\ x \in \boldsymbol{R^k} / Q(\mathbf{x}) = \mathbf{y_i}, \text{ si } d(\mathbf{x},\mathbf{y_i}) \leq d(\mathbf{x},\mathbf{y_j}), \text{ pour } \square j \neq i\} \tag{1.17}$$

C'est la condition du plus proche voisin. Chaque région contient un ensemble de vecteurs de $\boldsymbol{R^k}$ et tous les vecteurs **x** qui appartiennent à C_i sont représentés par le même vecteur $\mathbf{y_i}$ du dictionnaire.

La **compression d'information** est réalisée à ce niveau car c'est uniquement l'index du représentant $\mathbf{y_i}$ minimisant le critère de distorsion qui sera transmis ou stocké au lieu d'un vecteur. Cette opération, qui perd de l'information, fait que la quantification vectorielle est

une opération irréversible, le signal original ne pourra plus être restitué tel quel.

La quantité d'information requise pour représenter le vecteur source est donnée par *R* qui est le **débit binaire** en bits par composante du vecteur ou **bits par dimension**, ou encore **bits par échantillon** :

$$R = \frac{1}{k} \times \log_2 L \qquad (1.18)$$

1.8.4.2 Le décodeur

Le **décodeur** est considéré comme un récepteur voué à la reconstruction du vecteur source. Il dispose pour cela d'une réplique du dictionnaire qu'il consulte afin de restituer le code vecteur correspondant à l'index qu'il reçoit. Le décodeur réalise donc l'opération de décompression.

1.8.5 Schéma général d'un QV

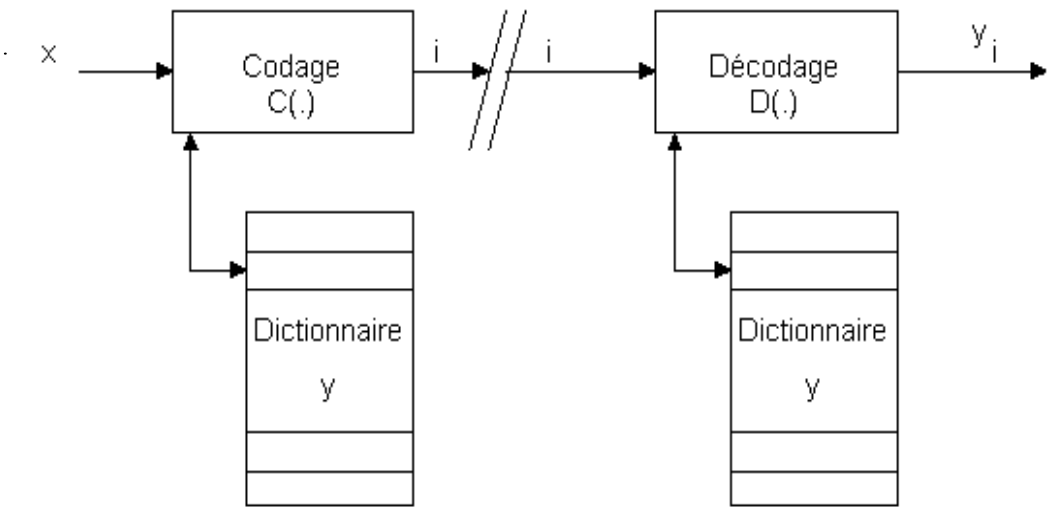


Fig 1.12 Schéma général d'un quantificateur vectoriel.

1.8.6 Algorithme de Lloyd généralisé (GLA : Generalized Lloyd Algorithm)

Supposons que nous disposions d'une certaine distance d. Construire un quantificateur revient donc à établir une stratégie de choix du dictionnaire associé. Cette stratégie est intimement liée à la nature de la distribution des vecteurs à quantifier.

Dans le cas ou les points d'entrée sont distribués d'une façon non uniforme, on adoptera une approche statistique visant à tirer parti de cette non-uniformité. Le dictionnaire sera construit par apprentissage à partir d'une large base de vecteurs d'entrée où sera sélectionné un nombre réduit de points susceptibles d'en refléter les propriétés statistiques.

En revanche, si la distribution des vecteurs d'entrée est plutôt uniforme, on aura intérêt à conférer à l'espace de représentation une structure mathématique forte, indépendamment de la "réalité" des données à traiter. Cette approche algébrique utilise généralement les propriétés des réseaux réguliers de points.

Les trois conditions d'optimalité citées précédemment conduisent à la conception d'un algorithme qui réalise, à partir d'une séquence d'apprentissage représentative de la statistique de la source à coder, la construction d'un dictionnaire (localement) optimal.

Cet algorithme de classification, encore appelé **algorithme des K-moyens** (K-means [12]) est l'extension au cas vectoriel de l'algorithme de Lloyd-Max du cas scalaire.
Il s'agit d'un algorithme d'optimisation itératif opérant à partir d'un dictionnaire initial (comme nous le verrons plus loin, le choix de ce dernier est essentiel). A chaque itération (dite ''itération de Lloyd''), deux opérations distinctes sont appliquées :

- Une classification suivant la règle de codage optimal (règle du plus proche voisin)
- Une optimisation suivant la règle de décodage optimal (condition du centroïde)

Chaque itération de Lloyd, en modifiant localement le dictionnaire, réduit ou laisse inchangée la distorsion moyenne. L'algorithme converge en un nombre fini d'itérations vers un minimum local.

La figure 1.13 illustre la procédure de fonctionnement de l'algorithme de Lloyd.

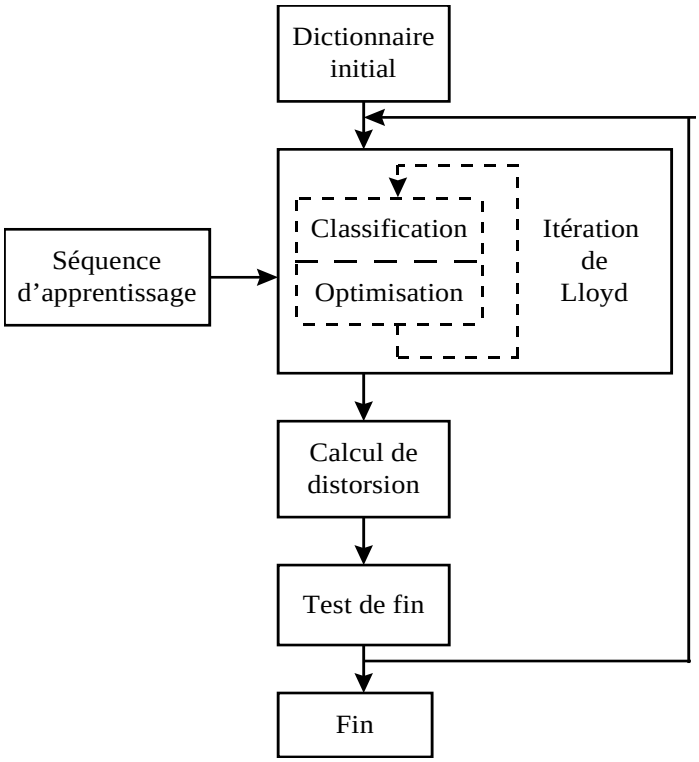


Fig 1.13 Schéma de fonctionnement de l'algorithme de Lloyd.

Etape 1 : on commence par un dictionnaire initial D_1, mettre k = 1;

Etape 2a : ayant un dictionnaire $D_k = \{y_i\}$, on partitionne la séquence d'entraînement en un ensemble de classes (clusters) C^i en utilisant la condition du plus proche voisin,

où $$C^i = \{x \in T \mid d(x, y_i) \leq d(x, y_j) \text{ ; pour tout } j \neq i\} \quad (1.19)$$

Etape 2b : en utilisant la condition du centroïde, on calcule les centroïdes pour l'ensemble des classes trouvées en étape 1 pour obtenir le nouveau dictionnaire :

$$C_{k+1} = \{Cent(C_i) \mid i = 1,... N\} \quad (1.20)$$

Etape 3 : On calcule la distorsion moyenne pour C_{k+1}, si elle a changé d'une petite quantité par rapport à l'itération précédente , mettre k = k+1 et on répète les étapes 2 et 3,sinon stop.

1.8.7 Conception du dictionnaire initial

Le choix de ce dictionnaire initial est essentiel car il conditionne les résultats finaux de l'algorithme sus-cité.

L'algorithme que nous allons décrire est une procédure d'optimisation itérative basée sur la méthode de projections successives et donc conduit vers un minimum local.

La vitesse de convergence des itérations de l'algorithme de Lloyd généralisé et les performances du dictionnaire obtenu après convergence dépendent du dictionnaire initial D_1. Par conséquent, il est important de trouver un bon dictionnaire initial.

L'une des techniques d'initialisation pour l'algorithme de Lloyd généralisé, est décrite ci-dessous [20].

Le principe de la technique, consiste à donner une attention particulière aux vecteurs d'apprentissage qui sont les plus éloignés l'un de l'autre, car ils sont susceptibles d'appartenir à des classes différentes.

Soit v_i, i = 1, ..., K, la séquence d'apprentissage des vecteurs. La procédure peut être exprimée comme suit :

a- Calculer les normes de tous les vecteurs de l'ensemble d'apprentissage. Choisir le vecteur ayant la norme maximum comme mot code ;

b- Calculer la distance de tous les vecteurs d'apprentissage par rapport au premier mot de code, et choisir le vecteur ayant la plus grande distance comme second mot de code. On a alors un dictionnaire de taille 2 ;

c- Généralement, avec un dictionnaire de taille i, i = 2, 3,...,L, nous calculons la distance entre les vecteurs d'apprentissage restants v_k et tous les mots codes existants. La plus petite valeur trouvée est appelée distance entre v_k et le dictionnaire. Alors, le vecteur d'apprentissage ayant la plus grande distance par rapport au dictionnaire est choisi pour être le $(i+1)^{ieme}$ mot code. La procédure s'arrête quand on obtient un dictionnaire de taille L.

L'idée de base de cette procédure est d'utiliser le vecteur le plus ''différent'' des codes vecteurs existants comme un nouveau mot code. La procédure est applicable pour n'importe quelle taille de dictionnaire. Dans ce cas nous n'avons pas besoin de définir un seuil quelconque.

A titre d'exemple, appliquons cette procédure pour concevoir un dictionnaire initial de 16 mots de codes en dimension 2, la séquence d'apprentissage est une source gaussienne de moyenne nulle et de variance 1. On obtient, comme on le remarque dans la figure 1.14, une partition de mots de codes très proche de la structure 1-6-9 qui est la structure optimale pour une source gaussienne à 2 dimensions.

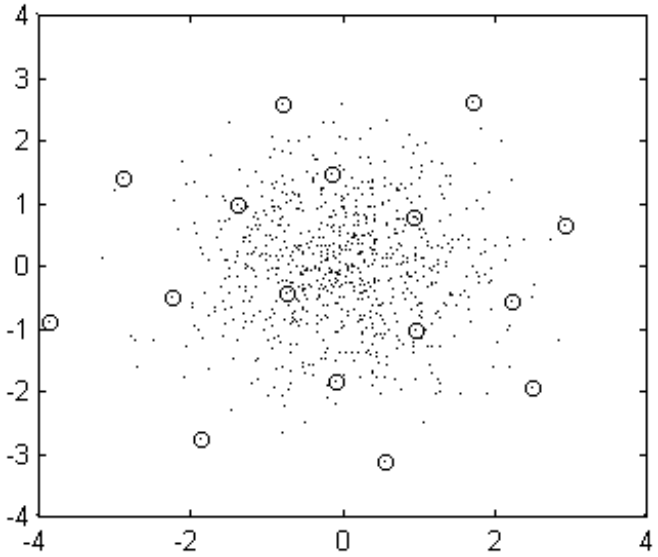


Fig 1.14 Structure obtenue pour une source gaussienne après une seule itération (proche de la structure 1-6-9 qui est la structure optimale pour cette source). (**.** : échantillon, o : mot de code obtenu)

1.9 Conclusion

A bas débit, la quantification vectorielle offre de meilleures performances que la quantification scalaire, particulièrement pour les signaux qui possèdent un caractère marqué de dépendance non linéaire. L'utilisation de la quantification vectorielle permet donc d'avoir une représentation des signaux plus compacte.

Chapitre 2

Codeur hybride utilisant la prédiction linéaire

2.1 Introduction

La prédiction linéaire est l'un des outils les plus importants en analyse de la parole. Sa relative simplicité de calcul et sa capacité à estimer convenablement les différents paramètres, font que cette méthode soit prédominante dans le codage de la parole à bas débit. L'idée de base de la prédiction linéaire est qu'un échantillon de parole peut être approximativement prédit par une combinaison linéaire des échantillons passés. Ainsi, en minimisant sur un intervalle fini (fenêtre) la somme des carrés des différences entre les actuels échantillons de parole et ceux prédits linéairement, on pourra déterminer les coefficients de prédiction. La prédiction peut être utilisée pour enlever les redondances du signal de parole ou pour établir un modèle pour le conduit vocal. La redondance est ainsi supprimée par un filtre prédictif linéaire (LP). Durant la prédiction court-terme, i.e suppression des redondances des échantillons proches, ce filtre est communément appelé filtre d'analyse. Le filtre d'analyse élimine la structure formantique du signal de parole (filtre d'analyse formant) et donne à sa sortie une erreur de prédiction de faible énergie appelée résiduel ou signal d'excitation. L'inverse du filtre **d'analyse**, i.e le filtre de **synthèse**, modélise le conduit vocal et sa fonction de transfert décrit l'enveloppe spectrale du signal de parole. Plus d'améliorations peuvent être obtenues en prenant en compte les corrélations long-terme du son voisé avec l'utilisation de la prédiction long-terme. Dans ce cas un autre filtre de prédiction linéaire est utilisé pour enlever les redondances des échantillons lointains. Ce filtre est usuellement appelé **prédicteur pitch**, il exploite la périodicité du signal. L'inverse du prédicteur pitch, souvent appelé **filtre pitch**, modélise l'effet de la glotte et sa fonction de transfert décrit la structure harmonique du signal de parole. La prédiction pitch n'est pas utile pour la parole non voisée du moment que l'excitation correspondante est aléatoire et que son spectre est pratiquement plat [11].

2.2 La Prédiction court-terme

Le modèle ''source-filtre'' nous permet l'utilisation de la prédiction linéaire pour enlever les redondances court-terme du signal de parole. Dans une trame de N échantillons,

le signal s[n] peut être considéré comme étant la sortie d'un système ayant une excitation d'entrée u[n] inconnue [21] :

$$s[n] = \sum_{k=1}^{p} a_k s[n-k] + G \sum_{l=0}^{q} b_l u[n-l], \qquad b_0 = 1 \tag{2.1}$$

où $\{a_k\}$, $\{b_l\}$ et le gain G représentent les paramètres du système. Dans l'équation (2.1), le signal de parole est *prédit* comme une combinaison linéaire des sorties antérieures et des entrées antérieures et présentes. La transformée en *z* du système H(z), est ainsi égale à :

$$H(z) = \frac{S(z)}{U(z)} = G \frac{1 + \sum_{l=1}^{q} b_l z^{-l}}{1 - \sum_{k=1}^{p} a_k z^{-k}} \tag{2.2}$$

où S(z) et U(z) sont les transformés en z de s[n] et u[n] respectivement.

H(z) représente dans l'équation (2.2) un modèle général pôle-zéro dit *modèle autorégressif à moyenne ajustée* (ARMA). Si a_k=0 pour, H(z) devient un modèle tout zéro ou modèle à moyenne ajustée (MA). Si b_l=0 pour, H(z) devient un modèle tout pôle ou modèle *autorégressif* (AR).

Pour des raisons de simplicité et d'efficacité de calcul, le modèle tout pôle est préféré dans la plupart des applications et s'identifie assez fidèlement au modèle du tube acoustique pour la production de la parole [22]. Bien que ce modèle donne une très bonne représentation du système vocal pour les voyelles qui contiennent des résonances, ce n'est qu'une approximation pour les classes de phonèmes comme les nasales et les fricatives qui contiennent des vallées spectrales correspondant aux zéros dans H(z). Néanmoins, l'oreille humaine est plus sensible aux pôles qu'aux zéros ce qui rend la simplification acceptable. De plus, il a été montré que l'effet d'un zéro dans la fonction de transfert peut être obtenu en incluant plus de pôles [23].

En se basant sur le modèle tout pôle, l'échantillon de parole courant est prédit par une combinaison linéaire des p échantillons passés :

$$\hat{s}[n] = \sum_{k=1}^{p} a_k s[n-k] \tag{2.3}$$

et l'erreur de prédiction ou résiduel du signal r[n] est donnée par :

$$r[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k] \tag{2.4}$$

En prenant la transformée en z des deux membres de l'équation (2.4) on obtient :

$$R(z) = A(z).S(z) \tag{2.5}$$

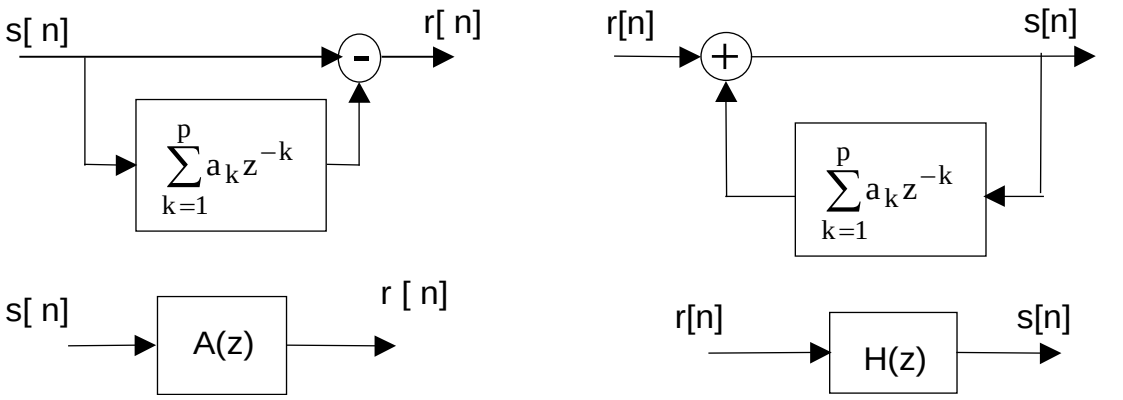
où R(z) représente la transformée en z du signal résiduel, et

avec $$A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k} \tag{2.6}$$

le filtre A(z) est appelé *filtre d'analyse* de la prédiction linéaire. Le filtre de synthèse de la prédiction linéaire,

$$H(z) = \frac{1}{A(z)} \tag{2.7}$$

modélise l'enveloppe spectrale de puissance du signal de parole. Un diagramme bloc d'analyse et de synthèse de la structure formantique est donné à la figure 2.1 :



(a) Etage d'analyse du formant (b) Etage de synthèse du formant

Fig.2.1 Diagramme bloc de la structure formantique.

L'ordre p du modèle est choisi de façon à ce que l'estimation de l'enveloppe spectrale soit adéquate. Une façon de procéder est d'allouer une paire de pôles pour chaque formant présent dans le spectre. On ajoute 2 à 4 pôles pour approximer les zéros dus aux sons non voisés.

Si la prédiction linéaire est basée sur les échantillons de parole passés s(n), celle-ci est dite prédiction linéaire adaptative progressive (Forward Adaptative) et dans ce cas les coefficients de prédiction doivent être transmis au décodeur comme information externe. Si maintenant la prédiction est basée sur les échantillons de parole passés reconstruits $\bar{s}(n)$, celle-ci est dite prédiction linéaire adaptative régressive (Backward Adaptative) et les coefficients n'ont plus besoin d'être transmis au récepteur. Pour avoir les coefficients du filtre

court-terme $\{a_k\}$ (appelés aussi coefficients LPC) du processus AR, la méthode classique des moindres carrés peut être utilisée. La variance ou l'énergie du signal erreur r[n] est minimisée sur une trame de parole. Deux grandes approches sont utilisées pour l'analyse LPC (Linear Prediction Coding) court-terme : la méthode d'autocorrélation et la méthode de covariance.

2.3 Méthode d'autocorrélation

Dans cette méthode c'est le signal parole qui est fenêtré alors que pour la méthode de covariance, le fenêtrage s'effectue sur le signal erreur (résiduel). Donc dans la méthode d'autocorrélation, le signal parole $s[n]$ est d'abord multiplié par une fenêtre d'analyse $w[n]$ de longueur finie afin d'obtenir un autre signal $s_w[n]$ qui est nul en dehors de la fenêtre. Dans le cas où la fenêtre de longueur N est positionnée en zéro, nous avons :

$$s_w[n] = w[n].s[n] \quad \text{pour} \quad 0 \le n \le N-1, \tag{2.8}$$
$$s_w[n] = 0 \qquad \text{ailleurs.}$$

Plusieurs fenêtres d'analyse de différentes formes ont été proposées, mais celle de Hamming qui est une fonction cosinus rehaussée, est la plus utilisée.

$$w[n] = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{N-1}\right), & 0 \le n \le N\text{-}1, \\ 0, & \text{ailleurs.} \end{cases} \tag{2.9}$$

L'énergie du résiduel peut donc être minimisée pour donner les coefficients du filtre linéaire de prédiction. Soit E l'énergie de l'erreur :

$$E = \sum_{n=-\infty}^{\infty} e^2[n] = \sum_{n=-\infty}^{\infty} \left(s_w[n] - \sum_{k=1}^{p} a_k s_w[n-k] \right)^2 \tag{2.10}$$

où $e[n]$ est le résiduel correspondant au signal fenêtré $s_w[n]$. Les valeurs de a_k qui minimisent E sont déterminés en annulant les dérivés partielles $\dfrac{\partial E}{\partial a_k}$ pour k=1,2,3,…,p. Cela donnera p équations linéaires avec p coefficients inconnus a_k :

$$\sum_{k=1}^{p} a_k \sum_{n=-\infty}^{\infty} s_w[n-i]s_w[n-k] = \sum_{n=-\infty}^{\infty} s_w[n-i]s_w[n], \quad 1 \le i \le p \tag{2.11}$$

La fonction d'autocorrélation du segment de parole pondéré $s_w[n]$ est donnée par :

$$R(i) = \sum_{n=i}^{N-1} s_w[n]s_w[n-i], \qquad 0 \le i \le p \tag{2.12}$$

où $R(i)$ est une fonction paire, $R(i) = R(-i)$. Alors les équations linéaires peuvent être écrites sous la forme :

$$\sum_{k=1}^{p} R(|i-k|)a_k = R(i), \qquad 1 \le i \le p \tag{2.13}$$

Sous la forme matricielle, l'ensemble des équations linéaires est représenté par **Ra = v** qui peut être réécrit en :

$$\begin{bmatrix} R(0) & R(1) & . & . & R(p-1) \\ R(1) & R(0) & . & . & R(p-2) \\ . & R(1) & . & . & . \\ . & . & . & . & . \\ R(p-1) & R(p-2) & . & . & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ a_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ . \\ . \\ R(p) \end{bmatrix} \tag{2.14}$$

La matrice résultante est une matrice Toeplitz. Cela facilite la résolution des équations de Yule-Walker pour les coefficients a_k avec des algorithmes de calcul rapides comme celui de Levinson-Durbin [21,24] ou celui de Schur [25]. C'est l'algorithme de Levinson-Durbin que nous utiliserons pour trouver les coefficients de prédiction. La structure Toeplitz garantit que $A(z)$ est à phase minimale (zéros et pôles à l'intérieur du cercle unité) [26]. Le filtre de synthèse $H(z) = 1/A(z)$ est donc stable. Cette propriété est un facteur motivant majeur pour l'utilisation de la méthode d'autocorrélation pour l'analyse LP.

2.4 Méthode de covariance

La minimisation dans la méthode de covariance est effectuée sur le signal erreur fenêtré de sorte que l'énergie à minimiser soit :

$$E = \sum_{n=-\infty}^{\infty} e_w^2[n] = \sum_{n=-\infty}^{\infty} e^2[n]w[n] \tag{2.15}$$

En annulant les dérivations partielles par rapport aux coefficients du filtre $\frac{\partial E}{\partial a_k} = 0$ pour k=1,2,3,…,p, on a p équations linéaires :

$$\sum_{k=1}^{p} \phi(i,k) a_k = \phi(i,0), \qquad 1 \le i \le p, \tag{2.16}$$

où la fonction de covariance $\phi(i,k)$ est définie par :

$$\phi(i,k) = \sum_{n=-\infty}^{\infty} w[n]\, s[n-i]\, s[n-k] \tag{2.17}$$

Sous forme matricielle, les p équations deviennent **Φ a = Ψ**, ou :

$$\begin{bmatrix} \phi(1,1) & \phi(1,2) & . & . & \phi(1,p) \\ \phi(2,1) & \phi(2,2) & . & . & \phi(2,p) \\ . & . & . & . & . \\ . & . & . & . & . \\ \phi(p,1) & \phi(p,2) & . & . & \phi(p,p) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ a_p \end{bmatrix} = \begin{bmatrix} \psi(1) \\ \psi(2) \\ . \\ . \\ \psi(p) \end{bmatrix} \tag{2.18}$$

où $\psi(i) = \phi(i,0)$ pour $1 \le i \le p$.

La matrice de covariance préserve sa propriété de symétrie et est définie positive mais n'est pas nécessairement Toeplitz, ce qui rend la méthode moins efficace en terme de calcul comparativement à la méthode d'autocorrélation. La matrice de covariance peut être décomposée en un produit de matrices triangulaire supérieure et inférieure :

$$\mathbf{\Phi = L\ U} \tag{2.19}$$

La décomposition de Cholesky est habituellement utilisée pour convertir la matrice de covariance en un produit de deux (02) matrices :

$$\mathbf{\Phi = C\ C^T} \tag{2.20}$$

où **C=L** et $\mathbf{C^T=U}$. On calcule le vecteur **a** en résolvant d'abord l'équation :

$$\mathbf{L\ y = \Psi} \tag{2.21}$$

Puis :

$$\mathbf{U\ a = y} \tag{2.22}$$

2.5 Algorithme de résolution

Nous admettons que la fonction d'autocorrélation R(k) est connue pour k=0,1,…, p. Il s'agit donc de résoudre le système linéaire (2.14), c'est-à-dire en fait inverser une matrice d'ordre p. Les méthodes algébriques classiques exigent pour cela un nombre d'opérations(multiplication+addition) de l'ordre de p^3, ce que l'on note par $O(p^3)$. L'algorithme qui va être décrit profite de la structure particulière (Toeplitz symétrique) de la matrice d'autocorrélation pour résoudre (2.14) par une récursion sur l'ordre de prédiction : autrement dit, il fournit toutes les solutions d'ordre m=1, 2, …, p. Le nombre d'opérations est seulement $O(p^2)$.

La variance de l'erreur de prédiction α_p sera obtenue également par une récurrence sur l'ordre m.

Algorithme de Levinson-Durbin.

Rappelons que la fonction d'autocorrélation est supposée connue et que pour un signal stationnaire, on a :

$$R(i,j) = R(|i-j|) = R(k) \tag{2.23}$$

Initialisation :

$$a_m(0) = 1, \quad (m=1, 2, \ldots, p) \quad \alpha_0 = R(0) = \sigma_x^2 \tag{2.24}$$

Récursion :

∗ pour m=1, 2, …, p

$$k_m = -\frac{1}{\alpha_{m-1}}\left[R(m) - \sum_{i=1}^{m-1} \alpha_{m-1}(i) R(m-i)\right] \tag{2.25}$$

∗ pour i=1, 2, …, m-1

$$\alpha_i(m) = \alpha_i(m-1) - k_m \alpha_{m-i}(m-1) \tag{2.26}$$

$$\alpha_m = \alpha_{m-1}\left(1 - k_m^2\right) \tag{2.27}$$

Les coefficients résultants, lorsque m=p, $\alpha_i(m)$ représentent les coefficients de prédiction d'un prédicteur linéaire d'ordre p :

$$a_i = \alpha_i(p) \qquad 1 \le i \le p$$

2.6 La Prédiction long-terme

Les segments de parole voisés montrent une forte corrélation long terme laquelle est maintenue dans le signal résiduel. Ces redondances peuvent être exploitées par l'utilisation d'un *prédicteur pitch*. Dans ce contexte, un filtre long terme peut être employé [27] :

$$P(z) = \beta z^{-D} \tag{2.28}$$

où β et D sont respectivement, le coefficient prédicteur et la période du pitch estimée en échantillons. Le signal erreur est exprimé par :

$$e(n) = r(n) - \beta r(n - D), \tag{2.29}$$

et est appelé signal résiduel pitch. En prenant la transformée en z des deux membres, le filtre d'analyse du pitch est donné par :

$$P_a(z) = 1 - \beta z^{-D}, \tag{2.30}$$

Dans le domaine temporel, le filtre d'analyse pitch soustrait de l'échantillon de parole courant l'échantillon (pondéré par β) situé à un retard égal à la période estimée. Dans le domaine fréquentiel, le filtre d'analyse pitch enlève la structure harmonique du signal d'entrée (le résiduel dans ce cas). L'analyse pitch n'aura pas d'effet utile sur le signal non voisé du moment que l'excitation non voisée est aléatoire (pas de structure harmonique). Le coefficient prédicteur β indique le degré de périodicité de la forme d'onde et prend des valeurs comprises entre 0 et 1. Ainsi β est proche de 0 pour une structure non périodique (et la valeur de la période D est donc sans signification) et égal pratiquement à l'unité pour de la parole voisée.

Au niveau du décodeur, le filtre de synthèse pitch est donnée par :

$$P_z(z) = \frac{1}{P_a(z)} = \frac{1}{1 - \beta z^{-D}}, \tag{2.31}$$

et est utilisé pour introduire la structure harmonique au niveau du signal de parole synthétisé.

2.7 Analyse par synthèse

Presque tous les codeurs standards de la parole font partie de la classe des codeurs utilisant l'analyse par synthèse par prédiction linéaire (LPAS coders). Dans cette classe de codeurs, on trouve les trois standards ITU (International Telecommunications Union) G.723.1, G.728, et le G.729 et tous les standards cellulaires digitaux en [28] :

♦ Europe : Système Global pour Télécommunications mobiles (GSM), débit total, débit moitié, et débit total amélioré.

♦ Amérique du Nord : Débit total et débit total amélioré pour les systèmes à accès multiple à division temporelle (TDMA) et les systèmes à accès multiple à division de code (CDMA).

♦ Japon : débit total et débit moitié.

♦ Le standard fédéral 1016 : codeur de la parole à 4.8Kb/s pour la téléphonie sécuritaire.

La parole décodée est obtenue en filtrant le signal produit par le générateur d'excitation à travers un filtre de synthèse prédicteur long-terme (LT) et un filtre de synthèse prédicteur court-terme (ST). Le signal d'excitation est obtenu en minimisant l'erreur quadratique moyenne sur un bloc d'échantillons. Le signal erreur est la différence entre les deux signaux perceptuels (original et synthétique). Le filtre court-terme modélise les corrélations court-terme (enveloppe spectrale) dans le signal de parole. C'est un filtre tout pôle. Les coefficients prédicteurs sont déterminés à partir du signal de parole en utilisant les techniques de prédiction linéaire (LP) et sont adaptés dans le temps, avec des débits allant de 30 jusqu'à 400 fois par seconde.

Le filtre prédicteur long-terme modélise les corrélations long-terme (structure fine spectrale) dans le signal de parole et possède comme paramètres : le délai (retard) et le gain. Pour les signaux périodiques, le délai correspond à la période du pitch (ou éventuellement un multiple de la période du pitch). Celui-ci est aléatoire pour les signaux non périodiques. Typiquement, les coefficients prédicteurs long-terme sont adaptés 100 à 200 fois par seconde.

Une structure alternative communément utilisée pour le filtre pitch est le **dictionnaire adaptatif**. Dans cette approche, le filtre de synthèse long-terme est remplacé par un dictionnaire contenant les excitations passées à différents délais. Les vecteurs résultants sont scrutés et celui qui minimisera l'erreur sera sélectionné. De plus, un facteur d'échelle optimal peut être déterminé pour le vecteur sélectionné.

Pour avoir un débit total minimum, le nombre moyen de bits par échantillon pour chaque trame d'excitation doit être petit. Dans le codeur à impulsions multiples, l'excitation est représentée comme une séquence d'impulsions localisées à des intervalles espacés de façon non uniforme [29]. La procédure d'analyse de l'excitation consiste à déterminer les amplitudes et les positions des impulsions. Trouver ces paramètres pour toutes les impulsions simultanément reste difficile à effectuer et de plus simples procédures comme la détermination des positions et amplitudes séquentiellement sont utilisées. Le nombre d'impulsions requis pour avoir une qualité de parole acceptable varie entre 4 et 6 impulsions toutes les 5 ms.

Les codeurs prédictifs linéaires excités par codes **(CELP)** [30], qui représentent la réalisation la plus commune dans le modèle d'analyse par synthèse, utilisent une autre approche pour réduire le nombre de bits par échantillon. Ici, l'encodeur comme le décodeur emmagasine la même collection des *C* séquences possibles de longueur *L* dans un dictionnaire. L'excitation pour chaque trame est complètement décrite par l'index d'un vecteur approprié dans le dictionnaire. L'index est trouvé par une recherche exhaustive sur tous les vecteurs possibles du dictionnaire, sélectionnant celui qui produit la plus petite erreur entre le signal décodé et le signal original.

Pour simplifier la recherche, la structure '' gain-forme'' pour le dictionnaire est généralement utilisée. Plus de simplifications peuvent être apportées en peuplant les vecteurs du dictionnaire avec une structure multiimpulsionnelle. En utilisant uniquement quelques impulsions (valeurs +1 et –1) pour chaque vecteur, on peut avoir des procédures de recherche efficaces. Cette partition de l'espace de l'excitation est connue sous le nom de dictionnaire algébrique et la méthode d'excitation qui en découle est dite *prédiction linéaire excitée par code algébrique* (ACELP) [31].

2.7.1 Le codeur CELP

2.7.1.1 *Introduction*

Le codeur CELP est un *quantificateur vectoriel prédictif*. Il est largement utilisé pour compresser les signaux de parole en bande téléphonique à des débits inférieurs à 16 kbit/s depuis son introduction par Atal et Schroeder en 1982 dans la version dite multi-impulsionnelle puis en en 1985 dans la version plus générale. Il est également utilisé pour compresser des signaux de parole en bande élargie.

2.7.1.2 *Principe des codeurs CELP*

On rappelle que le schéma de principe d'un codeur CELP est équivalent au schéma de modélisation standard d'un signal s(n) par un processus aléatoire stationnaire autorégressif $\hat{s}(n)$ comme le montre la figure 2.2 :

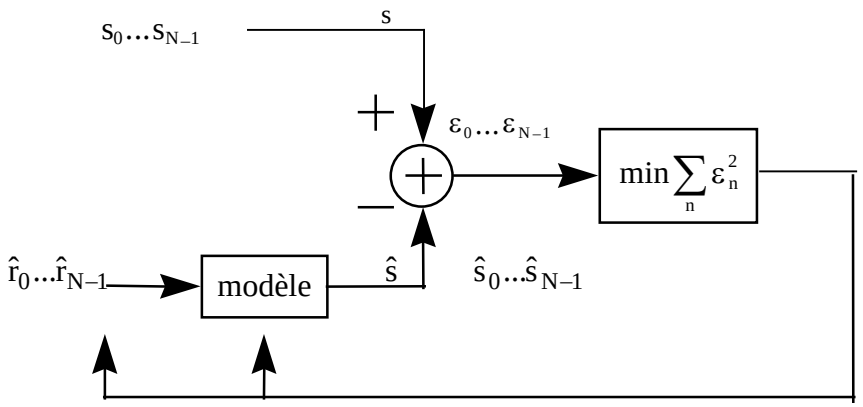


Fig.2.2 Principe de la modélisation paramétrique [22].

Comme nous l'avons déjà signalé, le signal de la parole n'est pas stationnaire. Ses propriétés statistiques varient au cours du temps. Alors, pour le traiter, on doit faire une approximation qui consiste à le considérer comme localement stationnaire sur des intervalles de temps de l'ordre de 10 à 30 ms.

Des fenêtres d'analyse sont alors introduites de longueur 80 à 240 échantillons si l'on choisit une fréquence d'échantillonnage de 8 kHz. On dispose ainsi de "N" échantillons qu'on utilisera pour extraire les caractéristiques du modèle. Ces dernières, codées à l'émetteur et transmises à travers le canal de transmission, sont décodées au récepteur.

On cherche à construire un signal synthétique $\hat{s}_0 \ldots \hat{s}_{N-1}$ le plus ressemblant possible au signal de départ $s_0 \ldots s_{N-1}$ sur l'ensemble de la fenêtre d'analyse.

Il faut donc définir le mode de construction du signal synthétique et préciser le critère de ressemblance. Une règle de construction possible est le passage d'un signal $\hat{r}_0 \ldots \hat{r}_{N-1}$ à travers un système. Le choix se rapporte en général sur des filtres linéaires, invariants (dans la fenêtre d'analyse) et causaux. On détermine ensuite les valeurs numériques des différents paramètres qui interviennent dans ce modèle. On minimise un critère de ressemblance entre le signal réel et le signal synthétique. Pour des raisons de simplicité, on choisit presque toujours un critère quadratique de la forme :

$$E = \sum_{n=0}^{N-1} (s_n - \hat{s}_n)^2 = \sum_{n=0}^{N-1} \varepsilon_n^2 \qquad (2.32)$$

Ici, il faut déterminer $\hat{r}_0 \ldots \hat{r}_{N-1}$ et les coefficients du filtre : on note que c'est un problème de déconvolution. Vu qu'on ne peut déterminer simultanément l'entrée et les

coefficients du filtre, une solution sous-optimale consiste à déterminer d'abord les coefficients du filtre en faisant des hypothèses très simplificatrices au niveau de l'entrée puis on raffine le modèle de l'entrée.

2.7.1.3 *Modèle de synthèse de la parole dans un codeur CELP*

La synthèse de la parole dans un codeur CELP [29], [30], [32] et [33] est identique à celle utilisée dans les codeurs prédictifs adaptatifs [34]. Elle consiste en deux filtres récursifs linéaires à coefficients variables dans le temps, ayant chacun un prédicteur dans la boucle de contre réaction figure (2.3). La première boucle contient un prédicteur long-terme qui génère les périodicités du pitch d'un son voisé, la seconde contient un prédicteur court-terme qui restaure l'enveloppe spectrale.

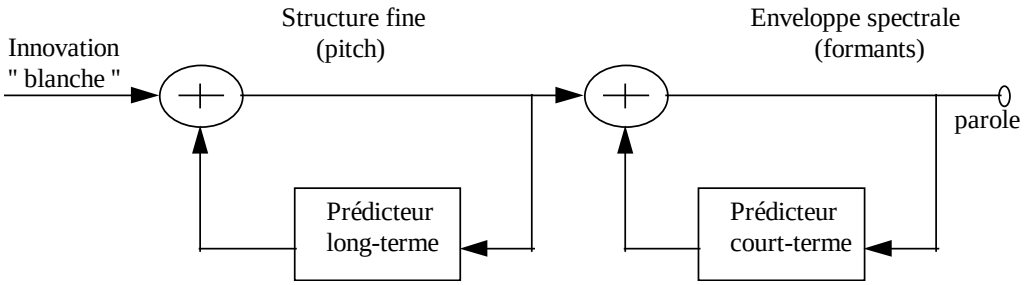


Fig.2.3 Modèle de synthèse de la parole [30].

Les deux prédicteurs sont déterminés en utilisant les méthodes décrites dans les références [35] et [36]. Le prédicteur court-terme a, par exemple, 16 coefficients et est déterminé en utilisant en analyse LPC la méthode de covariance stabilisée pondérée toutes les 10 ms [34] et [36]. Dans cette méthode, l'erreur de prédiction instantanée est pondérée par une fenêtre de Hamming de 20 ms et les coefficients sont obtenus en minimisant l'énergie de l'erreur pondérée.

La forme générale du prédicteur long-terme est :

$$\frac{1}{P(z)} = \frac{1}{1 - \sum_{j=-1}^{J} \beta_j z^{-(M+j)}} \tag{2.33}$$

où β_j : les coefficients gain.

M : la période pitch ($20 \leq M \leq 147$)

J : (prend les valeurs -1, 0, et 1).

Le décalage M approxime la périodicité du signal. Le gain β peut être interprété comme un indicateur du *niveau de périodicité* avec β approchant la valeur 1 pour des signaux *très périodiques*. Ces paramètres sont déterminés par analyse-synthèse et en boucle fermée.
Pour les sons voisés, la fréquence du pitch varie de 54 Hz à 400 Hz. Pour une fréquence d'échantillonnage de 8 kHz, l'échelle de décalage est comprise entre 20 et 147 échantillons (128 délais possibles) ce qui nécessite 7 bits pour le codage.
L'utilisation du prédicteur pitch est importante dans le codeur CELP. Sans lui, le dictionnaire aléatoire ne peut générer efficacement les composantes périodiques dans le signal d'excitation [37], [38] et [39]. En effet, cette périodicité correspond, physiologiquement, à la période de vibrations des cordes vocales.

2.7.2 Introduction d'un facteur perceptuel

Les fonctions de coût quadratiques se prêtent bien aux calculs : elles possèdent la propriété de fournir un système linéaire lorsque l'on dérive ce critère par rapport aux paramètres inconnus. Par contre, ce critère n'est pas forcément bien adapté à notre système auditif. Une correction perceptive est très largement utilisée pour pallier à cet inconvénient [40]. On rajoute une fonction de pondération, sous la forme d'un filtre de fonction de transfert $W(z)$, avant le critère de minimisation.

On sait que le bruit dû à la quantification est moins perceptible lorsque le signal a beaucoup d'énergie. On dit que le signal masque le bruit. Il n'est pas possible de jouer sur la puissance totale du bruit de quantification. Par contre il est possible de modifier sa forme spectrale. On cherche donc une fonction de pondération qui attribue moins d'importance aux zones fréquentielles énergétiques c'est-à-dire aux zones formantiques. On montre que la fonction de transfert $W(z) = A(z)/A(z/\gamma)$ avec $0 < \gamma < 1$ joue ce rôle. En effet, si on note

$$A(z) = 1 + a_1 z^{-1} + \ldots\ldots\ldots\ldots a_p z^{-p} = \prod_{i=1}^{p} (1 - p_i z^{-1}) \qquad (2.34)$$

On remarque que :

$$A(\frac{z}{\gamma}) = 1 + a_1 \gamma\, z^{-1} + \ldots\ldots\ldots\ldots a_p \gamma^p\, z^{-p} = \prod_{i=1}^{p} (1 - \gamma\, p_i z^{-1}) \qquad (2.35)$$

Le module de la réponse en fréquence du filtre 1/A(z/γ) présente des pics moins accentués que celui du filtre 1/A(z) puisque les pôles du filtre 1/A(z/γ) sont ramenés vers le centre du cercle unité par rapport à ceux du filtre 1/A(z). Le module de la réponse en fréquence du filtre $W(z) = A(z)/A(z/\gamma)$ a donc la forme souhaitée. Le tracé de la figure 2.4 montre, pour

trois fenêtres d'un signal correspondant à une transition d'un phonème voisé vers un phonème non-voisé, le module de la réponse en fréquence des filtres $1/A(z)$ et $W(z)$.

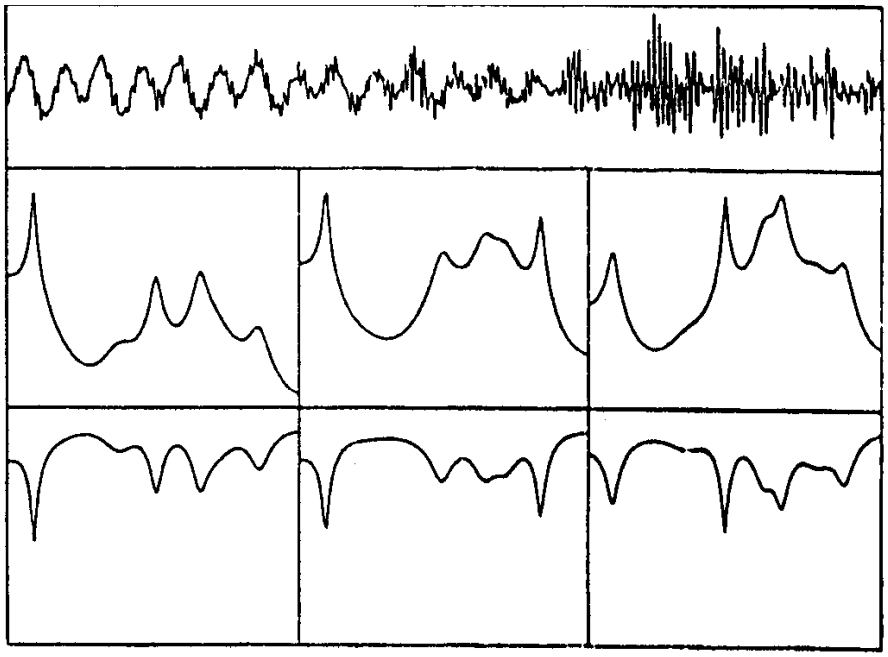


Fig.2.4 Spectre du signal et module de la réponse en fréquence de la fonction de pondération W(z) = A(z)/A(z/γ) pour trois fenêtres avec γ = 0.8.

2.7.3 La séquence optimale

Considérons par exemple, le codage d'une trame de signal parole échantillonnée à 8 kHz et de durée 2.5ms. Chaque trame consiste en 20 échantillons. Un débit binaire de 1/2 bits par échantillon correspond à 1024 séquences possibles de longueur 20 pour chaque bloc.

La procédure de sélection de la séquence optimale est illustrée en figure 2.5(a). Chaque élément du dictionnaire fournit 20 échantillons du signal innovation. Celui ci est mis à l'échelle par un facteur d'amplitude qui est constant et réactualisé à une nouvelle valeur à chaque cycle d'adaptation. Les échantillons ainsi mis à l'échelle sont filtrés séquentiellement à travers les deux filtres récursifs, pour introduire la périodicité des sons voisés et l'enveloppe spectrale. Les échantillons du signal de parole régénérés à la sortie du deuxième filtre sont comparés aux échantillons correspondant du signal de parole original pour former le signal différence.

Ce dernier, représentant l'erreur objective, est alors traité à travers le filtre de pondération perceptuel. Pour chaque innovation, on détermine le vecteur erreur puis on calcule sa valeur quadratique moyenne correspondante. L'innovation fournissant la valeur minimale de l'erreur perceptuelle est sélectionnée. Son indice est alors transmis avec les paramètres des filtres court-terme et long-terme et l'indice représentant la quantification des gains. La procédure est répétée pour chaque nouveau vecteur de parole.

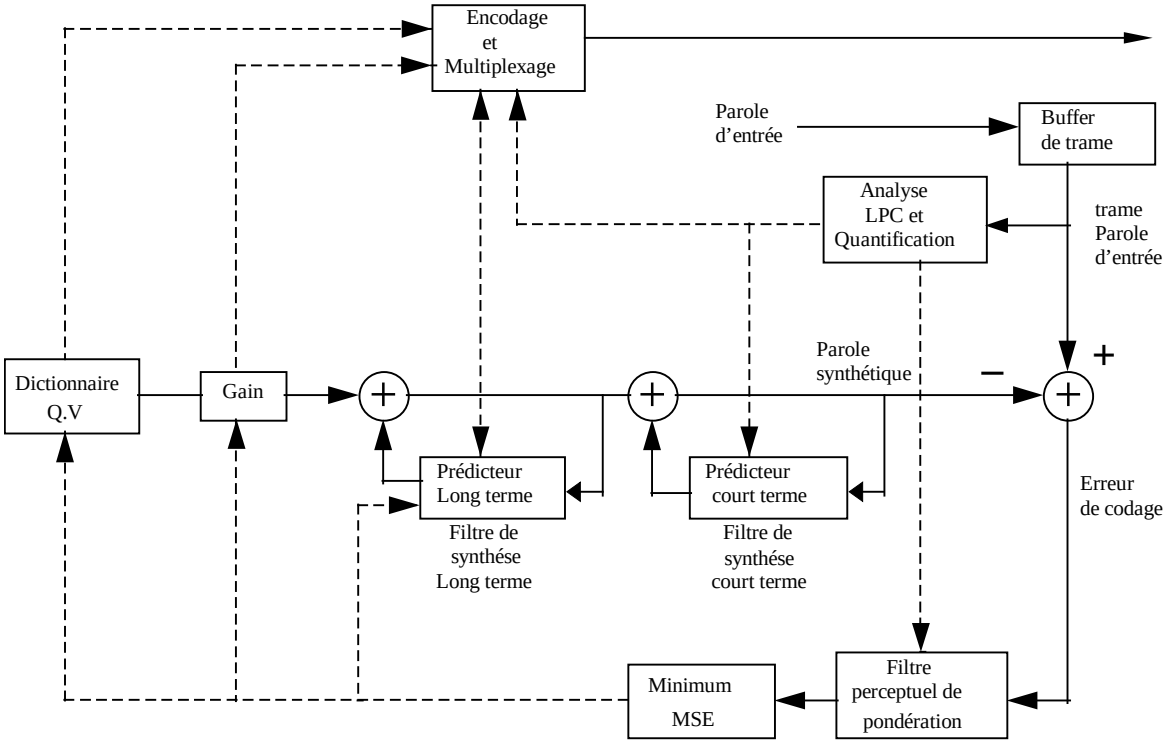


Fig.2.5 Procédure pour trouver la séquence optimale [41].

2.8 Conclusion

Dans le codeur CELP, le signal parole passe à travers les deux prédicteurs en cascade : le prédicteur long-terme qui génère les périodicités du pitch d'un son voisé et le prédicteur court-terme qui restaure l'enveloppe spectrale. Les codeurs CELP utilisent la quantification vectorielle. L'index du code vecteur qui produit la meilleure qualité du signal de parole est transmis avec le gain correspondant.

Le principe de recherche dans le dictionnaire consiste en une minimisation de l'erreur quadratique perceptuelle et ceci en utilisant la technique d'analyse par synthèse.

La recherche dans le dictionnaire nécessite une grande charge de calcul (dizaines de MFLOPS) mais avec le développement d'algorithmes rapides, le codeur CELP peut être implémenté en temps réel en utilisant des DSP modernes. Cette technique est actuellement l'une des plus utilisée pour obtenir une haute qualité de parole à très bas débit.

Chapitre 3

Codeur à faible retard LD-CELP

3.1 Introduction

A partir du CELP conventionnel, des modifications ont été apportées pour satisfaire les exigences d'avoir un faible retard et une excellente qualité de parole reconstruite. Pour cela, le type d'adaptation utilisé en analyse LPC est fait d'une manière régressive (backward-adaptative) pour éliminer l'opération de bufferisation d'une longue trame de parole. Le prédicteur pitch est éliminé et pour compenser sa contribution, l'ordre de prédiction du filtre de synthèse est augmenté de 10 à 50. Un filtre perceptuel de pondération est utilisé et une prédiction linéaire régressive sur les gains logarithmiques est faite pour suivre la dynamique du signal d'entrée.

3.2 Principe du LD-CELP

Le codeur LD-CELP est basé sur la version adaptative régressive du codeur CELP conventionnel [32] et [33] . L'essence ou le principe du CELP qui consiste à utiliser la méthode d'analyse par synthèse pour la recherche dans le dictionnaire est retenue dans le codeur LD-CELP. Cependant pour avoir une haute qualité avec un faible retard, la plupart des blocs du CELP doivent subir des modifications.

Dans le CELP conventionnel, les paramètres de prédiction (court-terme et long-terme), les gains, et la séquence d'excitation sont tous transmis au récepteur. Dans le codeur LD-CELP montré dans la figure 3.1, on transmet seulement la séquence d'excitation. Tous les autres paramètres sont adaptés en backward. Cela veut dire que les coefficients de prédiction ne vont pas être dérivés des échantillons de parole d'entrée mais plutôt des échantillons de parole précédemment quantifiés [10] et [40]. Etant donné que ces échantillons sont aussi disponibles au niveau du décodeur, ce dernier peut aussi déterminer les coefficients prédicteurs en utilisant la même procédure que celle de l'encodeur. Ainsi, il n'est plus nécessaire de transmettre des bits d'information externes (side information) pour spécifier ces coefficients. De plus, avec l'adaptation en backward, il n'est pas nécessaire de bufférisier les 20 ms du signal de parole d'entrée pour l'analyse LPC. Par conséquent, le vecteur d'excitation devient l'unité de base de bufférisation et le retard de codage sera égal à trois fois la taille du vecteur. Comme la dimension du vecteur est inférieure et de beaucoup au 20 ms, le retard de codage est ainsi hautement réduit [10].

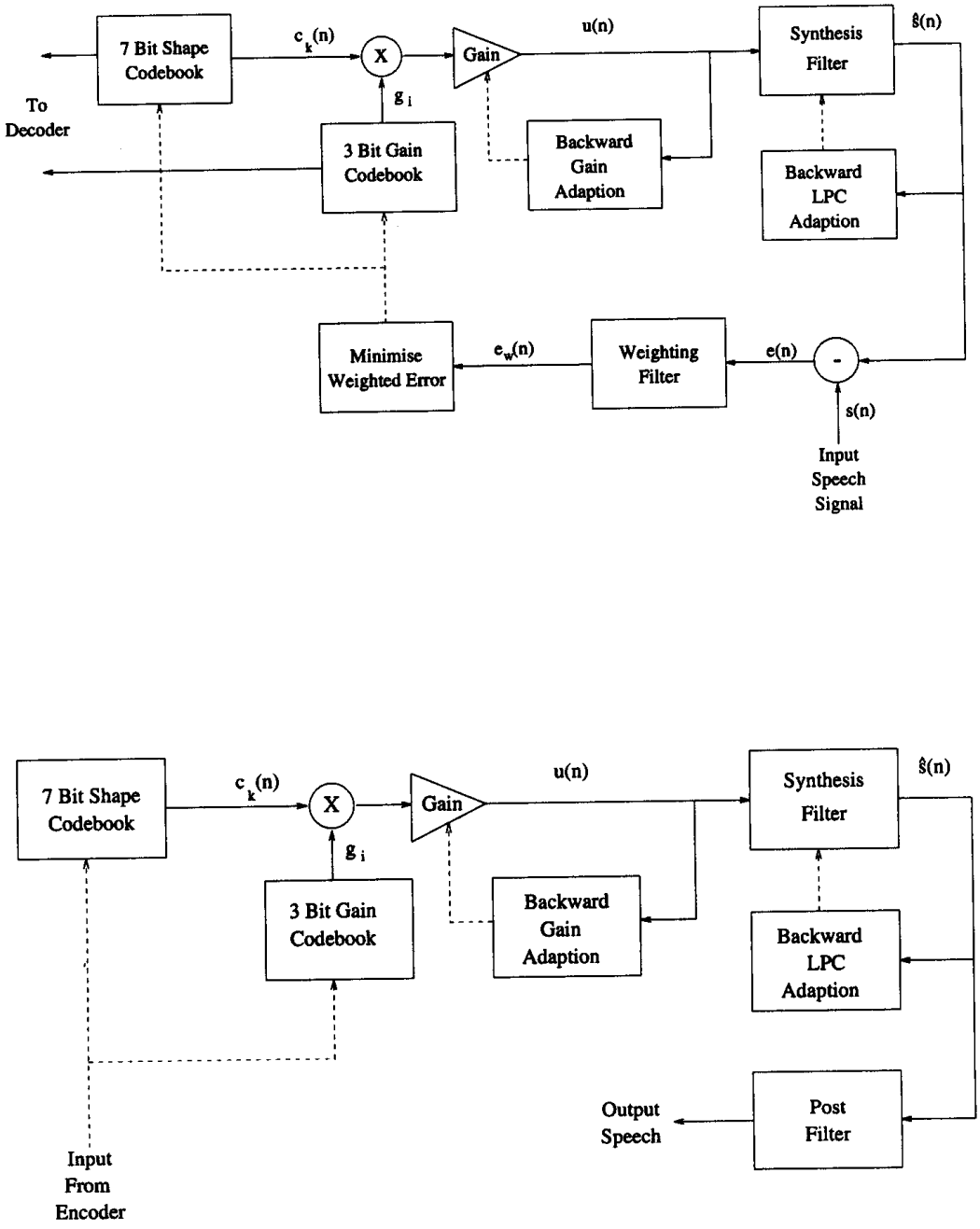


Fig.3.1 Codeur et Décodeur LD-CELP (Code Excited Linear Prediction) d'après [42].

3.3 Prédicteur LPC d'ordre supérieur

En développant l'algorithme du LD-CELP, on utilisa au début un prédicteur LPC d'ordre 10 et un prédicteur pitch trois états adaptés en backward . Bien que cette

configuration produisit une bonne qualité de parole pour un canal clair, l'adaptation backward du prédicteur pitch fut très sensible aux erreurs de canal. Cette mauvaise adaptation (divergence) due à la présence des bits erreurs a conduit les auteurs [40] à supprimer le prédicteur pitch ; cependant l'élimination de ce prédicteur a fait que la qualité de la parole prononcée par un locuteur féminin se dégrade de façon significative, bien que l'effet sur un locuteur masculin soit beaucoup moins notable. Pour compenser cette perte de qualité, l'ordre du prédicteur LPC a été augmenté de 10 à 50. La motivation de ce choix a été faite afin d'exploiter les redondances du pitch dans la parole prononcée par un locuteur féminin.

Nous avons trois justifications pour le choix de l'ordre 50 pour le filtre LPC :

a- l'analyse LPC adaptative régressive est apparue assez robuste aux erreurs de canal même après l'augmentation de l'ordre LPC à 50,

b- nous ne sommes plus obligés de transmettre des bits pour spécifier les 50 coefficients du modèle LPC vu qu'ils sont adaptés en backward,

c- le gain de prédiction est généralement saturé pour des ordres supérieurs à 50 : pour un locuteur masculin, ce gain se sature à l'ordre 20 alors que pour un locuteur féminin il n'atteint la saturation qu'à l'ordre 50 (figure 3.2).

Un autre avantage possible a été trouvé : en éliminant le prédicteur pitch et en utilisant à la place un prédicteur LPC d'ordre 50, le codeur devient moins spécifique pour le signal parole du moment qu'il ne tient compte d'aucune quasi-périodicité du pitch dans le signal d'entrée. Ce qui peut expliquer pourquoi le codeur fonctionne très bien avec d'autres types de signaux comme la musique ou bien les signaux de données modem.

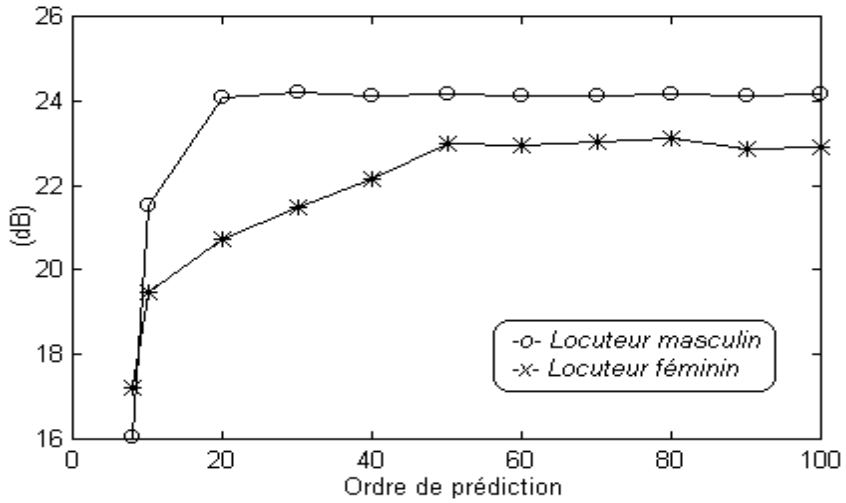


Fig 3.2 Evolution du gain de prédiction en fonction de prédiction du filtre de synthèse.

3.4 Opération de fenêtrage

Le choix de la fonction de fenêtrage joue un rôle important dans la capture des variations statistiques du signal d'entrée [42]. En général, pour l'analyse LPC, une fenêtre non récursive comme celle de Hamming est utilisée mais vue que cette méthode nécessite une opération de bufferisation en plus de la charge de calcul (multiplication et addition) pour l'estimation de la fonction d'autocorrélation, cette fenêtre a été remplacée par la version modifiée de la fenêtre de Barnwell [43]. Cette dernière représente la réponse impulsionnelle d'un filtre numérique récursif et possède une longueur infinie. Le calcul de la fonction d'autocorrélation se fait d'une façon récursive et vue que la matrice d'autocorrélation est de type Toeplitz, la résolution du système d'équations peut donc être accomplie par récursion (sans inversion de matrice) en utilisant la méthode de Levinson-Durbin.

La forme de la fenêtre w(n) doit être de telle sorte que le signal dans le passé immédiat soit pondéré plus que le signal dans le passé lointain. Ainsi le prédicteur agit selon les modes de changement de la stationnarité du signal de parole d'entrée.

Les paramètres importants dans le choix d'une fenêtre sont la forme et la longueur effective de cette fenêtre. Un autre facteur à prendre en considération est la complexité de calcul de l'algorithme de réactualisation. Une méthode pour réduire cette complexité dans la procédure de réactualisation est d'utiliser une fenêtre w(n) qu'on peut considérer comme la réponse impulsionnelle d'un filtre causal d'ordre fini. L'utilisation de ces fenêtres nous permet d'avoir la possibilité d'avoir des équations de réactualisation récursives [43].

Pour décrire la manière dont les coefficients d'autocorrélation sont calculés en utilisant la fenêtre hybride, notons par m l'indice du dernier échantillon dans une certaine trame et w(n) le n^{ieme} échantillon de la fonction fenêtre, de sorte que w(n)=0 pour n<0 et w(n) est indicée en backward dans le temps. Le signal de parole fenêtré $\hat{s}(n,m)$ où n est l'indice temps et m le numéro de la fenêtre est donné par :

$$\hat{S}(n,m) = s(n)\, w(m-n) \tag{3.1}$$

La fonction d'autocorrélation du signal fenêtré est déterminée à partir de :

$$R(k,m) = \sum_{n=-\infty}^{\infty} \hat{S}(n,m)\ \hat{S}(n+k,m) \tag{3.2}$$

où R(k,m) est le k^{ieme} coefficient d'autocorrélation pour la fenêtre m. Si la longueur de la fenêtre utilisée est finie alors les limites de la sommation dans (3.2) sont aussi finies. Ces coefficients d'autocorrélation sont alors utilisés comme entrées pour l'algorithme d'inversion de la matrice de Toeplitz afin de déterminer les paramètres du filtre LPC. Avec cette

approche, nous nous retrouvons confrontés à plusieurs problèmes lors de l'estimation des fonctions d'autocorrélation. D'abord, pour avoir en général une bonne qualité de la parole, les fenêtres doivent être recouvrantes (overlapping). Ainsi, plusieurs échantillons peuvent être utilisés pour plus d'une trame lors du calcul des coefficients d'autocorrélation. De plus, l'opération de recouvrement conduit à une augmentation considérable de bufférisation et de charge de calcul. Ces problèmes peuvent être évités si le besoin d'une longueur de fenêtre finie est écarté. D'où l'intérêt de l'utilisation d'une classe de fenêtres de longueur infinie. L'une des fenêtres de cette classe peut être formée à partir de la réponse impulsionnellle d'un filtre numérique de second ordre ayant deux pôles réels. La transformée en z correspondant à ce filtre est :

$$H(z) = \frac{1}{(1+\alpha z^{-1})(1+\beta z^{-1})} \tag{3.3}$$

où α et β représentent le lieu des pôles.

A partir des équations (3.1) et (3.2), les fonctions d'autocorrélations pour les séquences fenêtrées peuvent être écrites comme suit :

$$R(k,m) = \sum_{n=+\infty}^{n=-\infty} s(n)\,s(n+k)\,w(m-n)\,w(m-n-k) \tag{3.4}$$

Maintenant en définissant :

$$W(n,k) = w(n)\,w(n-k) \tag{3.5}$$

et :

$$S(n,k) = s(n)\,s(n+k) \tag{3.6}$$

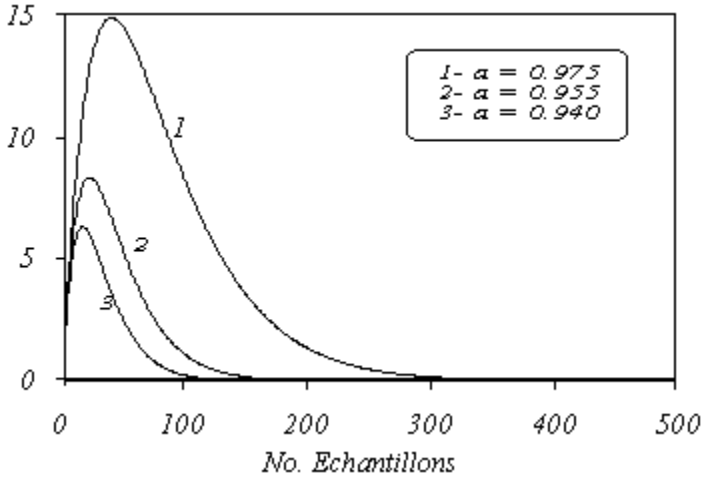


Fig 3.3 Exemples de réponses impulsionnelles pour des filtres à réponse impulsionnelle infinie avec un pôle réel double à $z = -\alpha$.

l'équation (3.4) devient :

$$R(k,m) = \sum_{n=-\infty}^{n=+\infty} S(n,k)\,W(m-n,k) \tag{3.7}$$

A partir de cette équation, on peut voir que le k^{ieme} coefficient d'autocorrélation peut être exprimé comme la convolution de la séquence $\{S(n,k)\}$ et la fonction $W(n,k)$. De plus, sachant que $W(n,k)$ est le produit de deux fonctions fenêtres, alors $W_k(z)$, la transformée en z de $W(n,k)$ est donnée par la convolution des transformées en z des deux fonctions fenêtres $w(n)$ et $w(n-k)$.

Maintenant si la fenêtre est la réponse impulsionnelle de longueur infinie d'un filtre numérique H(z) alors $W_k(z)$ peut être écrit comme :

$$W_k(z) = \frac{1}{2\pi j}\oint H(\nu)H\left(z/\nu\right)\nu^{-k-1}d\nu \tag{3.8}$$

Si H(z) est prise comme étant la fonction de transfert du filtre de second ordre donnée par (3.3), $W_k(z)$ devient :

$$W_k(z) = \frac{1}{2\pi j}\oint \frac{\nu^{-k-1}}{(1-\alpha\nu^{-1})(1-\beta\nu^{-1})(1-\alpha\nu/z)(1-\beta\nu/z)}d\nu \tag{3.9}$$

L'évaluation de cette expression donne :

$$W_k(z) = \frac{b(0,k) + b(1,k).z^{-1}}{1 - a(1,k).z^{-1} - a(2,k).z^{-2} - a(3,k).z^{-3}} \tag{3.10}$$

où :

$$b(0,k) = \frac{\alpha^{k+1} - \beta^{k+1}}{\alpha - \beta} \tag{3.11}$$

$$b(1,k) = \frac{\alpha^2\beta^{k+1} - \beta^2\alpha^{k+1}}{\alpha - \beta} \tag{3.12}$$

$$a(1,k) = \alpha^2 + \beta^2 + \alpha\beta$$

$$a(2,k) = -(\alpha^2\beta^2 + \alpha^3\beta + \alpha\beta^3) \qquad (3.13)$$

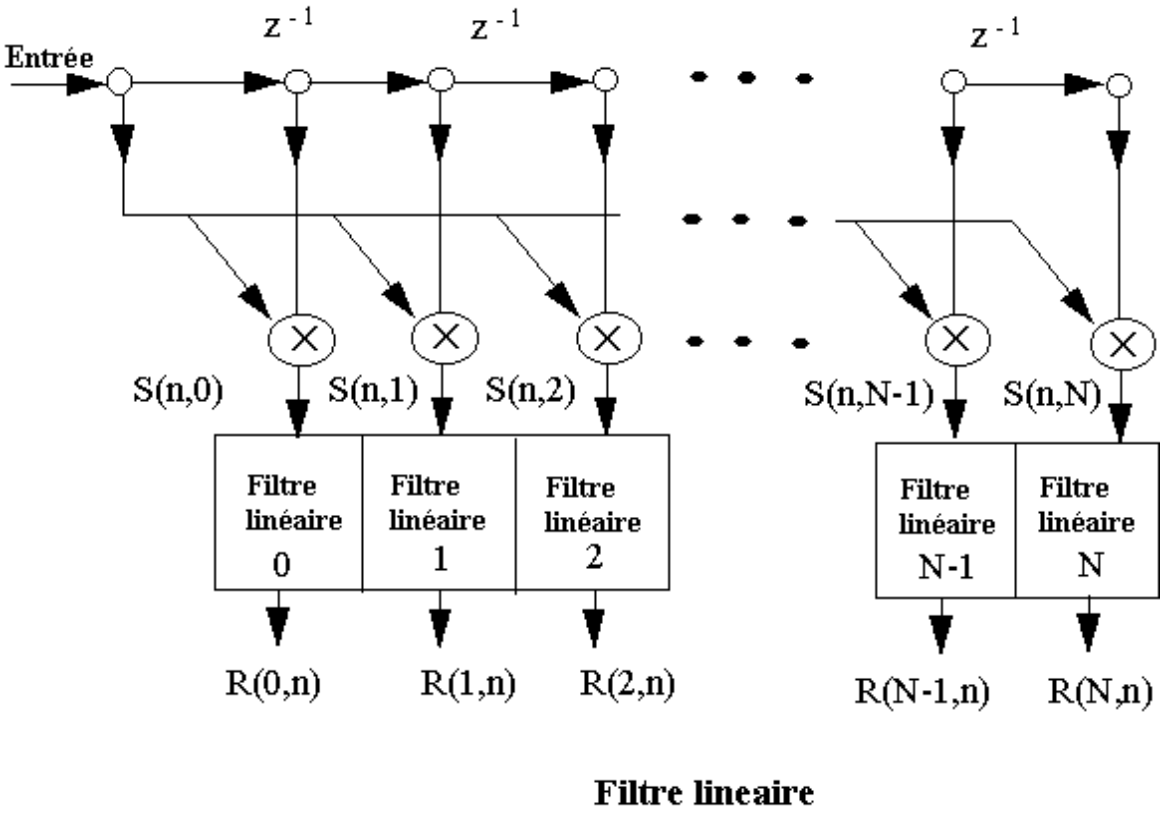
$$a(3,k) = \alpha^3\beta^3$$

Dans le cas où $\alpha = \beta$ (en faisant tendre β vers α) on aura :

$$\begin{aligned} b(0,k) &= (k+1)\,\alpha^k \\ b(1,k) &= -(k-1)\,\alpha^{k+2} \end{aligned} \qquad (3.14)$$

$$\begin{aligned} a(1,k) &= 3.\alpha^2 \\ a(2,k) &= -3.\alpha^4 \\ a(3,k) &= \alpha^6 \end{aligned} \qquad (3.15)$$

Ces équations montrent que les fonctions d'autocorrélation peuvent être calculées de façon récursive comme le montre la figure 3.4.



Filtre lineaire

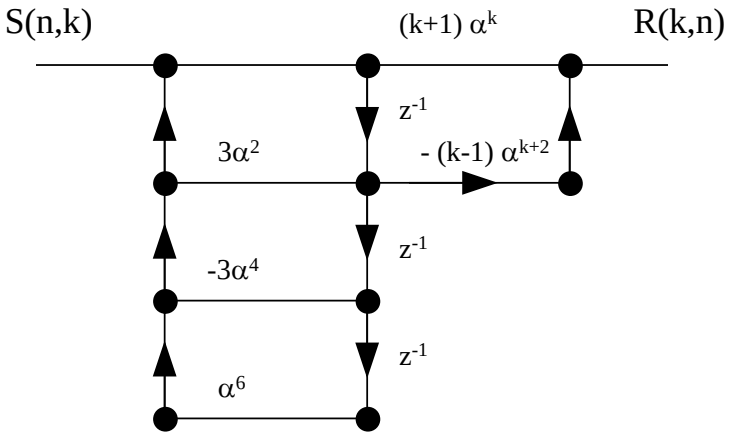


Fig 3.4 Structure pour le calcul récursif de la fonction d'autocorrélation estimée pour une analyse d'ordre N.

Analyse de la structure récursive :

Plusieurs points sont à noter concernant le système de la figure 3.4 :

* c'est un système point à point qui opère de façon identique sur chaque échantillon donc pas de bufferisation additionnelle autre que celle montrée à la figure 3.4 ;
* le paramètre α contrôle complètement la longueur de la fenêtre ;
* les deux multiplications dans la partie non récursive du filtre [$(k+1)\alpha^k$ et $(k-1)\alpha^{k+2}$] sont effectuées une seule fois par trame et non sur chaque échantillon ;
* les constantes multiplicatives dans les parties récursives des filtres linéaires sont identiques ce qui conduira à un stockage moins important et à des réalisations plus simples des filtres ;
* la logique de contrôle pour la structure entière est très simple ;
* du moment que toutes les informations concernant la fenêtre sont contenues dans les coefficients du filtre, alors un stockage extensive de la fonction fenêtre n'est pas nécessaire.

Les coefficients d'autocorrélation sont calculés récursivement en faisant passer le signal de parole à travers la structure spéciale du banc de filtres (figure 3.4).

Pour une analyse LPC d'ordre 50, il y a 51 filtres de forme directe d'ordre 3 dans le banc de filtres, un pour chaque coefficient d'autocorrélation. Cependant, l'obtention de ces coefficients par la méthode de Barnwell telle qu'elle est exposée, provoque des problèmes de précision numérique dans la séquence de l'analyse LPC d'ordre 50 dans le cas où l'on travaille en virgule fixe [10]. Ce problème est contourné en remplaçant le filtre de forme directe d'ordre 3 par des filtres de 1^{er} ordre en cascade [40].

Les fonctions de transfert de ces filtres de 1er ordre sont :

$$\frac{1}{1-\alpha^2 z^{-1}}, \frac{1}{1-\alpha^2 z^{-1}} \text{ et } \frac{(k+1)\alpha^k - (k-1)\alpha^{k+2} z^{-1}}{1-\alpha^2 z^{-1}} \tag{3.16}$$

où α est le paramètre de contrôle de la forme de la fenêtre.

3.5 Filtre perceptuel de pondération

Comme dans le codeur CELP conventionnel, un filtre perceptuel pour le masquage du bruit par les formants est utilisé.

La fonction de transfert de ce filtre pour le CELP conventionnel est de la forme :

$$W(z) = \frac{1-Q(z)}{1-Q(z/\gamma)} \tag{3.17}$$

où

$$Q(z) = \sum_{i=1}^{M} q_i z^{-i} \tag{3.18}$$

$$Q(z/\gamma) = \sum_{i=1}^{M} \gamma^i \, q_i \, z^{-i}, \qquad 0 < \gamma < 1 \tag{3.19}$$

q_i et M représentent respectivement les coefficients LPC quantifiés et l'ordre du prédicteur généralement pris égal à 10. Soit donc le prédicteur d'ordre 10, représenté par la fonction de transfert :

$$Q(z) = \sum_{i=1}^{10} q_i z^{-i} \tag{3.20}$$

où q_i représentent les coefficients du prédicteur. La fonction de transfert $W(z)$ du filtre perceptuel s'écrira :

$$W(z) = \frac{1-Q(z/\gamma_1)}{1-Q(z/\gamma_2)}, \quad 0 < \gamma_2 < \gamma_1 < 1 \tag{3.21}$$

avec :

$$Q(z/\gamma_1) = \sum_{i=1}^{10} \left(q_i \, \gamma_1^i\right) z^{-i} \tag{3.22}$$

$$Q(z/\gamma_2) = \sum_{i=1}^{10} \left(q_i \, \gamma_2^i\right) z^{-i} \tag{3.23}$$

Les valeurs optimales de γ_1 et γ_2 sont déterminées par des tests d'écoute.

Le filtre perceptuel est en général déplacé vers les deux branches avant l'opération de soustraction entre le signal original et son synthétique. Ceci peut être fait vu que le filtre est linéaire. La minimisation est alors équivalente à la minimisation de l'erreur quadratique moyenne de l'erreur entre les deux signaux (original et synthétique) pondérés comme montré dans la figure ci-dessous.

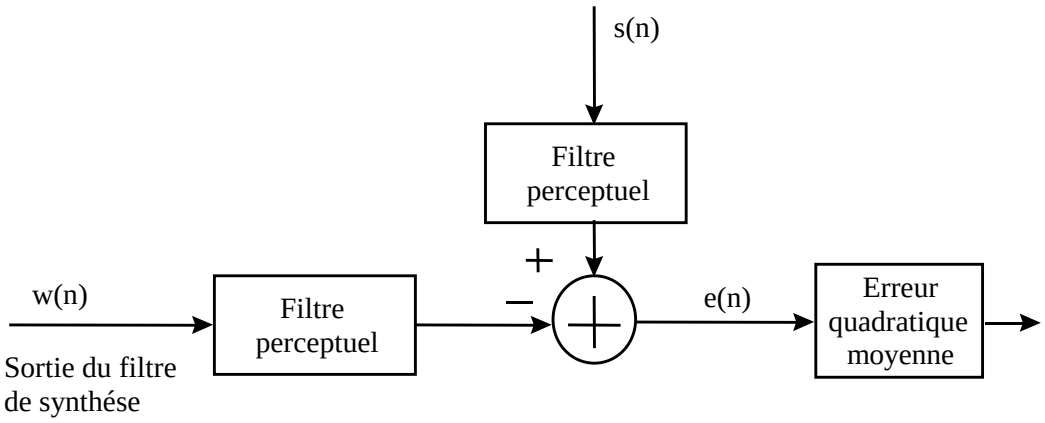
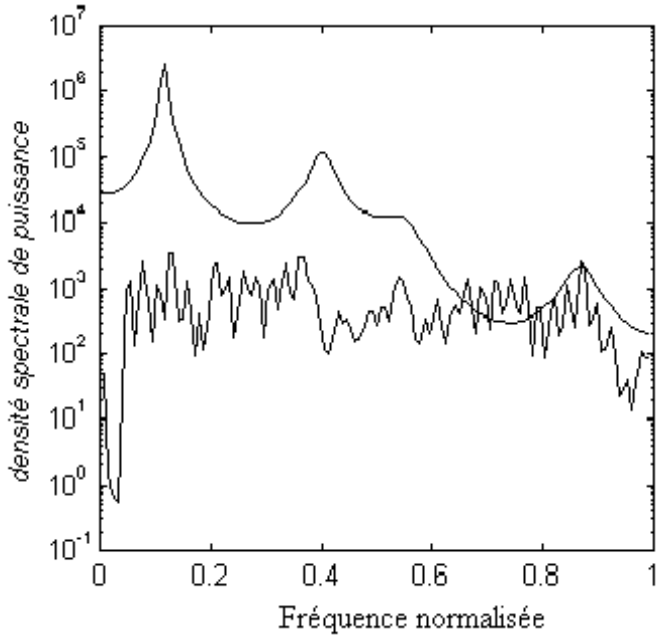
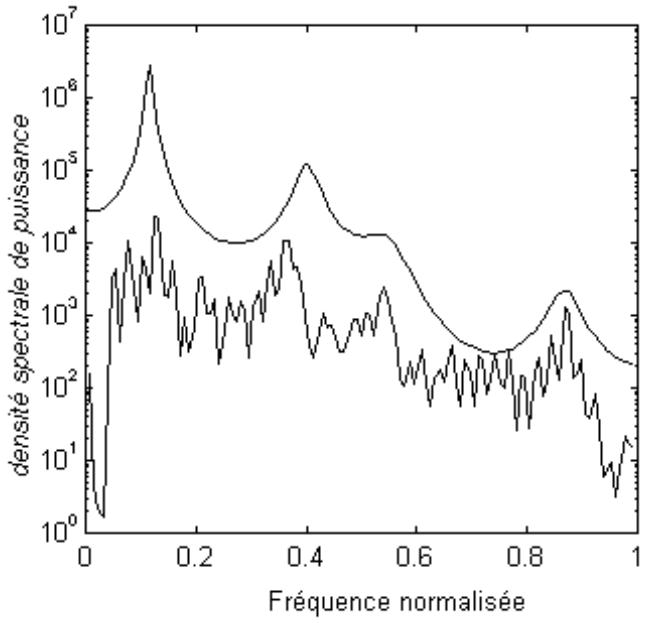


Fig 3.5 Déplacement du filtre perceptuel vers les deux branches d'entrées.

Le filtre de perception permet la redistribution de l'erreur dans le domaine spectral.



-a-



-b-

Fig 3.6 Densité spectrale de l'erreur

a- sans pondération b- avec pondération

3.6 Prédicteur de gain

Les codeurs de parole à faible retard utilisent la prédiction linéaire adaptative backward pour l'évaluation du gain d'excitation. Au niveau de l'encodeur, chaque vecteur est normalisé de façon appropriée avant l'opération d'encodage (quantification vectorielle). Les vecteurs normalisés ont une échelle dynamique réduite et peuvent donc être codés efficacement. Au niveau du récepteur, la sortie du décodeur est multipliée par le gain estimé [44]. Cette procédure permet de suivre assez fidèlement la puissance court-terme du signal parole.

Le vecteur d'excitation $e(n)$ mis à l'échelle s'écrit :

$$e(n) = \sigma(n).\, y(n) \tag{3.24}$$

où

$y(n)$ représente un des mot-codes excitation du dictionnaire à l'instant n et $\sigma(n)$ le gain d'excitation adaptatif en backward. Les écarts type (RMS) associés à $y(n)$ et $e(n)$ étant respectivement $\sigma_y(n)$ et $\sigma_e(n)$ nous avons l'égalité :

$$\sigma_e(n) = \sigma(n).\, \sigma_y(n) \tag{3.25}$$

Dans le domaine logarithmique l'équation (3.25) s'écrit :

$$\log[\sigma_e(n)] = \log[\sigma(n)] + \log[\sigma_y(n)] \tag{3.26}$$

Le but de ce prédicteur est d'obtenir $\sigma(n)$ aussi proche que possible de $\sigma_e(n)$ en exploitant les informations disponibles à l'instant n. Soit le prédicteur d'ordre 10 opérant sur la séquence $\log[\sigma_e(n-1)],\ \log[\sigma_e(n-2)],\ldots,\ \log[\sigma_e(n-10)]$ afin de prédire $\log[\sigma(n)]$:

$$\log[\sigma(n)] = \sum_{i=1}^{10} p_i.\log[\sigma_e(n-i)] \tag{3.27}$$

où p_i représentent les coefficients du prédicteur.

Donc :

$$\log[\sigma(n)] = \sum_{i=1}^{10} p_i.\log[\sigma(n-i)] + \sum_{i=1}^{10} p_i.\log[\sigma_y(n-i)] \tag{3.28}$$

D'après (3.28) nous pouvons noter que $\log[\sigma(n)]$ peut être vu comme la sortie d'un filtre pôle-zéro d'ordre 10 avec $\log[\sigma_y(n-1)]$ comme entrée. La réactualisation des coefficients p_i du prédicteur log-gain se fait à travers une analyse LPC régressive adaptative sur la séquence antérieure $\log[\sigma_e(n-i)]$, i=1,…,10.

L'efficacité de l'adaptation du gain en backward est montré à la figure 3.7. Sur celle-ci sont portées les fonctions densité de probabilité (PDF) des gains optimaux des vecteurs d'excitation (avec et sans adaptation). On constate que l'adaptation en gain produit une PDF avec des pics autour de 1 et avec une variance réduite ; ceci conduit à une meilleure quantification des gains optimaux.

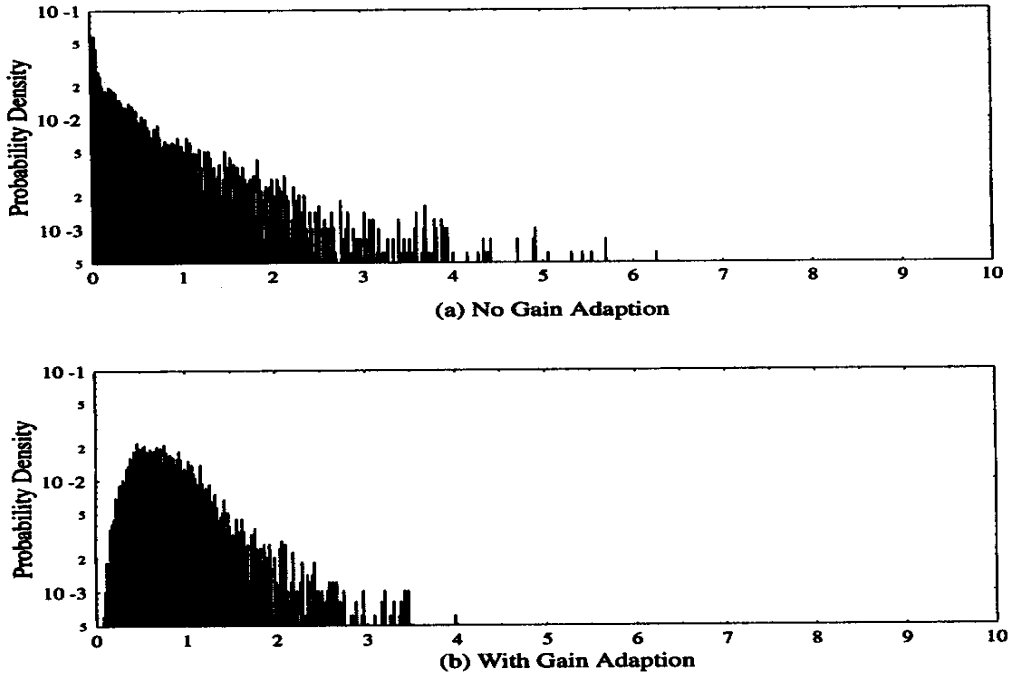


Fig 3.7 PDFs des gains normalisés avec et sans adaptation backward du gain d'après [42].

3.7 Conclusion

Se distinguant des codeurs CELP conventionnels qui transmettent cinq types d'informations différentes (adaptation en forward), le LD-CELP transmet seulement l'index de 10 bits de la quantification vectorielle du mot code c'est-à-dire de la meilleure excitation sélectionnée dans le dictionnaire. Le tableau ci-contre montre les deux types d'adaptation pour les cinq paramètres :

Paramètre	CELP conventionnel	LD-CELP
Coefficients LPC	Forward	Backward
Gain d'excitation	Forward	Backward
Vecteur d'excitation	Forward	Forward
Gain du prédicteur Pitch	Forward	Non utilisé
Période du prédicteur Pitch	Forward	Non utilisé

Tableau 3.1 Types d'adaptations pour le CELP conventionnel et le LD-CELP.

Afin d'avoir une haute qualité de parole avec un retard de codage assez faible, la plupart des blocs du codeur CELP conventionnel ont subi des modifications. Les principales caractéristiques du LD-CELP sont :

- un prédicteur LPC d'ordre supérieur avec une analyse LPC adaptative en **backward** ;
- adaptation régressive du gain d'excitation à l'aide d'un prédicteur log-gain adaptatif ;
- l'opération de fenêtrage pour les analyses LPC est effectuée par la fenêtre de BARNWELL modifiée qui fournit les coefficients d'autocorrélation de façon récursive ; cette fenêtre est en fait la réponse d'un filtre IIR d'ordre 2 ;
- l'utilisation d'un filtre perceptuel de pondération est utilisé pour le masquage du bruit de codage.

Chapitre 4

Algorithme du LD-CELP (G728)

Dans ce chapitre, une description détaillée de la mise en œuvre de l'algorithme du LD-CELP G728 est décrite. Chaque bloc du codeur/décodeur est présenté. Une attention particulière est donnée pour le module de recherche dans le dictionnaire d'excitation. Une méthode de recherche permettant une réduction de complexité est appliquée. La procédure de conception (optimisation en boucle fermée) du dictionnaire "forme" et "gain" est également décrite.

4.1 Encodeur LD-CELP

La figure ci-dessous représente le schéma bloc de l'encodeur LD-CELP d'après [45] . Pour chaque variable à décrire, k est l'indice de l'échantillon et les échantillons sont pris à des intervalles de 125 μs (f_e = 8 KHz).

Un bloc de 5 échantillons consécutifs dans un signal donné est dit vecteur de ce signal, par exemple 5 échantillons consécutifs de parole forment un vecteur parole, 5 échantillons d'excitation forment le vecteur d'excitation. Nous utilisons "n" pour désigner l'indice vecteur, qui est différent de l'indice de l'échantillon "k".

L'indice du dictionnaire qui est la quantification du vecteur d'excitation (QV) est la seule information qui est explicitement transmise de l'encodeur au décodeur. Trois autres types de paramètres sont réactualisés périodiquement :

* le gain d'excitation ;
* les coefficients du filtre de synthèse ;
* les coefficients du filtre perceptuel de pondération.

Ces paramètres sont dérivés d'une manière adaptative en backward des signaux déjà reçus avant l'encodage du vecteur du signal de parole courant. L'actualisation est effectuée de la manière suivante : le gain d'excitation est actualisé à chaque nouveau vecteur, alors que les coefficients des filtres de synthèse et perceptuel sont réactualisés une fois tous les 4 vecteurs (c'est à dire, tous les 20 échantillons ou toutes les 2,5 ms), ceci pour alléger les charges de calcul.

On n'omettra pas de mentionner que bien que la séquence de traitement dans l'algorithme a un cycle d'adaptation de 4 vecteurs (20 échantillons), la taille du vecteur de bufferisation de base est seulement de 1 vecteur (5 échantillons).

Une description de chaque bloc de l'encodeur est donné ci-dessous :

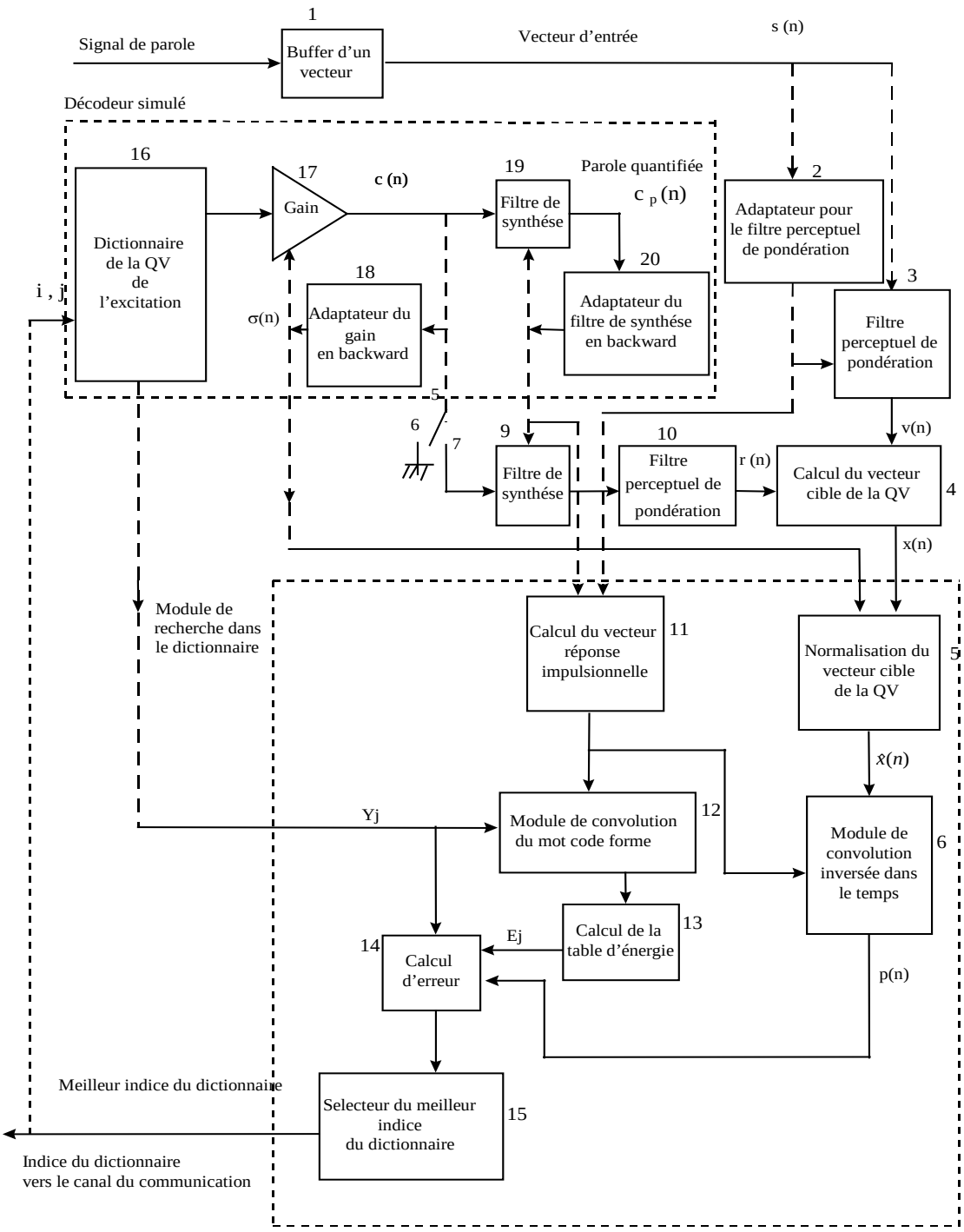


Fig 4.1 Schéma bloc détaillé de l'encodeur LD-CELP d'après [45].

4.1.1 Buffer

Ce bloc bufferise 5 échantillons consécutifs de la parole s(5n), s(5n+1), s(5n+2), s(5n+3), s(5n+4) pour former un vecteur parole de dimension 5. S(n) = [s(5n), s(5n+1), ...,s(5n+4)];

4.1.2 Adaptateur du filtre perceptuel de pondération (Bloc 2)

Cet adaptateur calcule les coefficients du filtre de pondération perceptuel une fois tous les 4 vecteurs en se basant sur l'analyse par prédiction linéaire et en utilisant la parole non quantifiée. Les coefficients sont maintenus constants entre les réactualisations. La figure 4.2 montre l'opération détaillée de ce bloc.

Le vecteur parole d'entrée s(n) (non quantifié) est passé au travers du module de fenêtrage récursif qui applique une fenêtre sur les vecteurs antérieurs de parole et calcule récursivement les 11 premiers coefficients d'autocorrélation. Le module de récursion Levinson-Durbin convertit ces coefficients d'autocorrélation en coefficients du prédicteur.

L'adaptateur du filtre perceptuel de pondération réactualise périodiquement les coefficients de W(z) selon les équations (3.22) et (3.23) puis transmet les coefficients au module de calcul du vecteur réponse impulsionnelle et au filtre perceptuel de pondération.

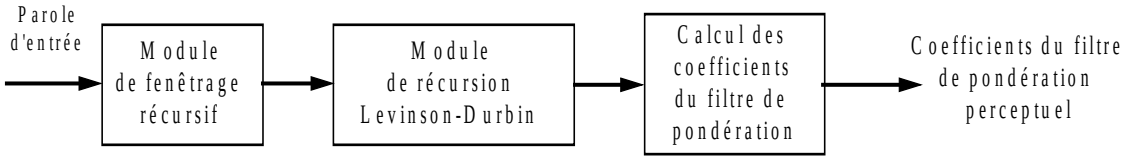


Fig 4.2 Adaptateur pour le filtre perceptuel.

4.1.3 Filtre perceptuel de pondération (Bloc 3)

Comme indiqué dans la figure 4.1, le vecteur parole d'entrée courant s(n) est passé au travers du filtre perceptuel de pondération, qui donnera un vecteur parole pondéré v(n). On remarque que, excepté durant l'initialisation, la mémoire du filtre ne doit pas être initialisée à zéro à n'importe quel instant. Autrement dit, cette dernière doit avoir un traitement spécial comme il sera décrit plus loin.

4.1.4 Filtre de synthèse (Blocs 9 et 19)

Dans la figure (4.1), il y a deux filtres de synthèse ayant des coefficients identiques. Ils sont réactualisés par l'adaptateur du filtre de synthèse backward (bloc 20). Chaque filtre de

synthèse est un filtre tout-pôle d'ordre 50. Sa fonction de transfert s'écrie :

$$F(z) = \frac{1}{1 - P(z)} \tag{4.1}$$

Où P(z) est la fonction de transfert du prédicteur LPC d'ordre 50.

Après l'obtention du vecteur parole pondéré v(n), le vecteur r(n) qui est une réponse à entrée nulle, doit être généré en utilisant le filtre de synthèse (bloc 9) et le filtre de pondération perceptuel (bloc 10). Pour accomplir ceci, on ouvre d'abord l'interrupteur 5, (le diriger vers le nœud 6). Ceci implique que le signal allant du nœud 7 vers le filtre de synthèse 9 est nul. On laisse alors ce dernier et le filtre de perceptuel (10) "tournés" durant 5 échantillons (1 vecteur). La sortie du filtre perceptuel 10 est la réponse à une entrée nulle r(n).

Il est à noter que sauf après initialisation, la mémoire des filtres 9 et 10 n'est en général pas nulle. Le vecteur sortie r(n) est en général différent de zéro en général, bien que l'entrée du filtre venant du nœud 7 vers le filtre de synthèse 9 soit nulle. Ce vecteur représente l'effet de la mémoire du filtre.

4.1.5 Calcul du vecteur cible (Target) (bloc4)

Ce bloc retranche le vecteur r(n) qui est une réponse à entrée nulle du vecteur parole pondéré v(n) afin d'obtenir le vecteur cible (target) t(n). Ce dernier sera utilisé, dans le module de recherche dans le dictionnaire.

4.1.6 Adaptateur du filtre de synthèse (bloc 20)

L'adaptation du filtre s'effectue en backward. Cet adaptateur prend le vecteur parole quantifié c(n) comme entrée et donne en sortie les coefficients des filtres de synthèse 9 et 19. Il réactualise donc les coefficients de ces deux filtres.

Une version de cet adaptateur est donnée en figure 4.3.

Les opérations du module de fenêtrage et du module de Levinson-Durbin sont identiques à ceux de la figure 4.2 à l'exception des 3 différences suivantes:

* le signal d'entrée est maintenant de la parole quantifiée au lieu de la parole originale;
* l'ordre de prédiction est de 50 au lieu de 10;
* la longueur effective de la fenêtre de Barnwell est plus grande.

Ces filtres de synthèse ont comme fonction de transfert:

$$F(z) = \frac{1}{1 - P'(z)} \tag{4.2}$$

où P'(z) représente la fonction de transfert du prédicteur LPC modifié.

$$P'(z) = \sum_{i=1}^{50} a_i' \, z^{-i} \tag{4.3}$$

avec:

$$a_i' = \lambda^i \, a_i \qquad i=1,2,\ldots,50. \tag{4.4}$$

Les a_i représentent les coefficients du filtre prédicteur LPC d'ordre 50 :

$$P(z) = \sum_{i=1}^{50} a_i z^{-i} \tag{4.5}$$

Donc pour augmenter la robustesse aux erreurs du canal, les coefficients a_i sont modifiés de façon à ce que les pics dans le spectre LPC résultant aient des largeurs de bande un peu plus larges. Le module d'expansion de la largeur de bande transforme les coefficients a_i en un nouveau ensemble de coefficients a_i' suivant la relation (4.4). Le coefficient λ est calculé par :

$$\lambda = e^{-\left(\frac{\pi \Delta \omega}{8000}\right)} = 0.9883 \tag{4.6}$$

Les coefficients modifiés sont pris comme entrées des filtres de synthèse 9 et 19 et du module de calcul du vecteur réponse impulsionnelle.

Comme pour le filtre perceptuel, les filtres de synthèses 9 et 19 sont réactualisés une fois tous les 4 vecteurs. Les actualisations sont basées sur la parole quantifiée et au dernier vecteur du cycle d'adaptation antérieur.

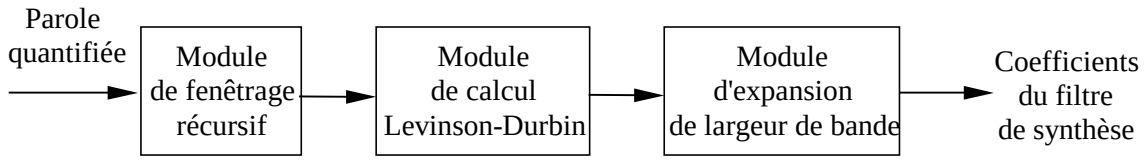


Fig 4.3 Adaptateur du filtre de Synthèse

4.1.7 Adaptateur gain du vecteur d'excitation (bloc 18)

Ce module réactualise le gain d'excitation $\sigma(n)$ pour chaque vecteur à l'instant n. Le gain d'excitation $\sigma(n)$ est un facteur d'échelle utilisé pour le vecteur d'excitation y(n) sélectionné. Ce module prend le vecteur d'excitation mis à l'échelle c(n) comme entrée et donne $\sigma(n)$ (gain d'excitation) comme sortie. Il « prédit » le gain de c(n) en se basant sur les gains de c(n-1), c(n-2), ... Une prédiction linéaire adaptative dans le domaine de gain logarithmique est utilisée.

L'adaptateur gain opère de la façon suivante : l'unité de décalage d'un vecteur fournit le vecteur d'excitation mis à l'échelle antérieure c(n-1). Le module de calcul RMS (Root Mean Square) calcule la valeur RMS (en dB) du vecteur c(n-1). Puis, celle en dB du RMS c(n-1).

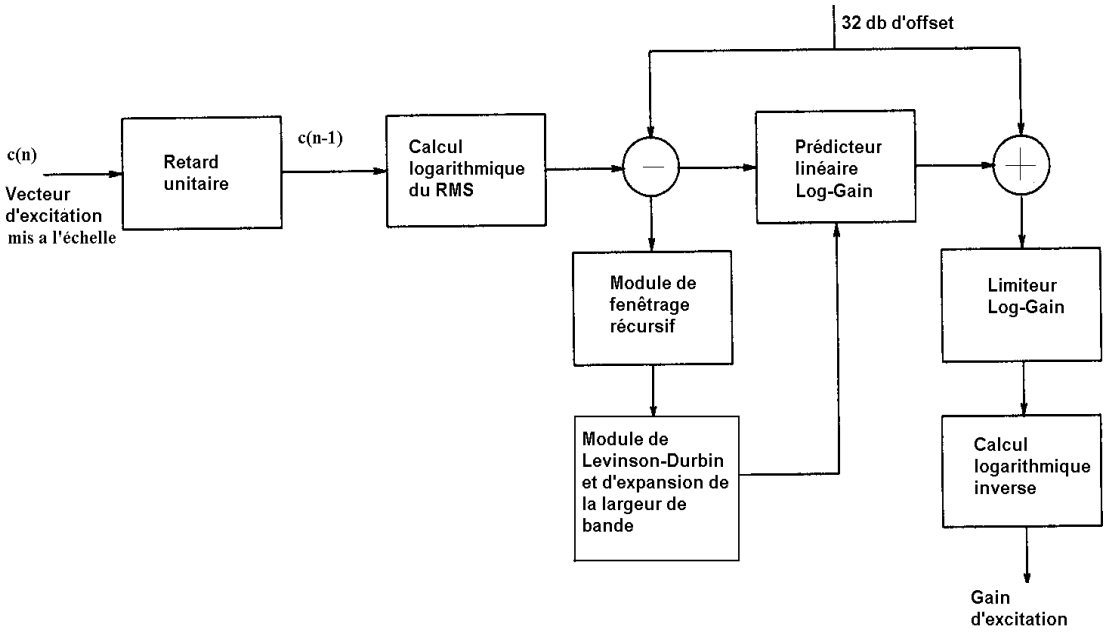


Fig 4.4 Schéma du prédicteur du gain d'excitation du G728 [42].

Une valeur d'offset log-gain de 32 dB est retranchée de cette valeur (logarithme du RMS). cette valeur est environ égale à l'énergie moyenne du gain d'excitation (en dB) durant la parole voisée.

Le gain logarithmique δ(n-1) ayant l'offset retranché est alors utilisé par le module de fenêtrage récursif et celui de calcul de récursion Levinson-Durbin. Ces blocs opèrent exactement de la même façon que ceux dans le module de l'adaptateur du filtre perceptuel de pondération, excepté que la longueur effective de la fenêtre est moins longue et que le signal analysé est un gain logarithmique au lieu de la parole d'entrée. Il faut souligner qu'une seule valeur est calculée pour tous les 5 échantillons de parole.

Le module de récursion Levinson-Durbin donne les coefficients du prédicteur linéaire d'ordre 10 ayant comme fonction de transfert :

$$R(z) = \sum_{i=1}^{10} \overline{\alpha}_i z^{-i} \tag{4.7}$$

Le module d'expansion de largeur de bande déplace les racines de ce polynôme radialement vers l'origine du plan z. Le prédicteur gain à largeur de bande élargie résultant a une fonction de transfert :

$$R(z) = \sum_{i=1}^{10} \alpha_i z^{-i} \tag{4.8}$$

Où les coefficients α_i sont : $\alpha_i = (0.9)^i \overline{\alpha}_i$

Cette expansion de largeur de bande permet à l'adaptateur gain d'être plus robuste aux erreurs de canal. Les α_i sont alors utilisés comme coefficients prédicteur linéaire log-gain.

Le prédicteur est actualisé une fois par cycle d'actualisation. Le prédicteur tend à prédire δ(n) en se basant sur une combinaison linéaire de δ(n-1), δ(n-2), ..., δ(n-10). La version prédite de δ(n) est notée $\overline{\delta}(n)$ est donnée par :

$$\overline{\delta}(n) = \sum_{i=1}^{10} \alpha_i \delta(n-i). \tag{4.9}$$

Après l'obtention de $\overline{\delta}(n)$, on ajoute la valeur d'offset log-gain de 32 dB. Le limiteur log-gain, alors, vérifie si sa valeur résultante n'est pas trop grande ou trop petite et limite cette valeur si nécessaire. Les limites inférieures et supérieures sont mise respectivement à 0 dB et 60 dB. La sortie de ce limiteur log-gain est ensuite convertie au domaine linéaire. Le limiteur gain permet d'avoir un gain dans le domaine linéaire entre 1 et 1000.

4.2 Recherche de l'excitation optimale

Pour réduire la complexité de recherche dans un dictionnaire de 10 bits (1024 mots codes), nous avons adopté la structure "gain-shape". Le dictionnaire est ainsi décomposé en deux dictionnaires: un de 7 bits (128 mot-code) pour la quantification de la forme, et un autre de 3 bits (8 mot-codes) pour la quantification du gain.

Ensuite, nous avons utilisé la technique du **ZIR** (**Z**ero **I**nput **R**esponse) qui consiste à prendre la mémoire des filtres en cascade (synthèse et perceptuel) et la soustraire de la branche haute (figure 4.5). Ceci, nous a permis d'effectuer les opérations de filtrage (filtres sans mémoire) sous forme de calcul matriciel.

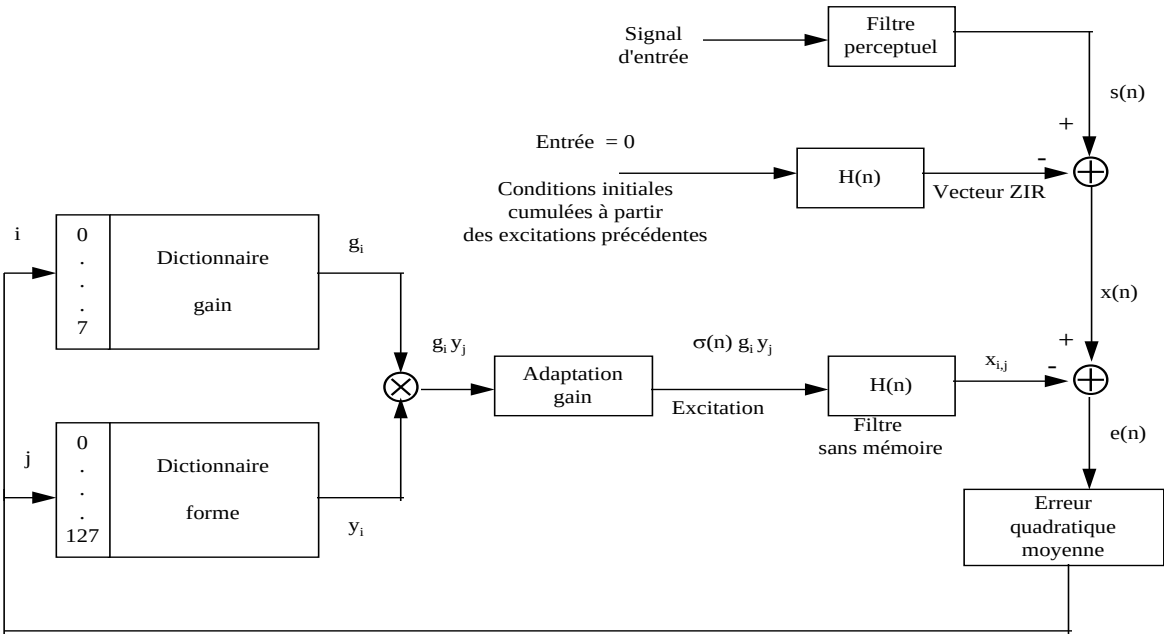


Fig 4.5 Introduction de la technique ZIR et procédure de sélection de la meilleure excitation.

En principe, le module de recherche dans le dictionnaire met à l'échelle chacun des 1024 mots codes candidats par le gain d'excitation courant $\sigma(n)$ et filtre les 1024 vecteurs obtenus par les 2 filtres en cascade F(z) et W(z). La mémoire du filtre est initialisée à zéro à chaque fois que le module présente un nouveau mot code au filtre résultant H(z):

$$H(z) = F(z).W(z). \qquad (4.10)$$

Soit y_j le j^{ieme} mot code dans le dictionnaire forme (j varie de 0 à 127), et soit g_i le i^{ieme} niveau dans le dictionnaire gain (i varie de 0 à 7).

Soit H(n) la réponse impulsionnelle du filtre H(z). Quand le mot code spécifié par les indices i et j des dictionnaires est à l'entrée du filtre en cascade H(z), la sortie du filtre est exprimée par :

$$x_{i,j} = H.\sigma(n).g_i y_j . \tag{4.11}$$

où H est une matrice dont les éléments sont la réponse impulsionnelle des filtres en cascade.

$$H = \begin{bmatrix} h(0) & 0 & 0 & 0 & 0 \\ h(1) & h(0) & 0 & 0 & 0 \\ h(2) & h(1) & h(0) & 0 & 0 \\ h(3) & h(2) & h(1) & h(0) & 0 \\ h(4) & h(3) & h(2) & h(1) & h(0) \end{bmatrix} \tag{4.12}$$

La recherche dans le dictionnaire consiste à chercher la meilleure combinaison des indices i et j qui minimise la distorsion **MSE** (**M**ean **S**quared **E**rror) suivante :

$$D = \left| x(n) - x_{ij} \right|^2 = \sigma^2(n) \left| \overline{x}(n) - g_i \, H \, y_j \right|^2 \tag{4.13}$$

Où $\overline{x}(n) = x(n)/\sigma(n)$ est le vecteur cible QV à gain normalisé.

Après développement on obtient :

$$D = \sigma^2(n) \left[\left|\overline{x}(n)\right|^2 - 2\, g_i \, \overline{x}^T(n) \, H \, y_j + g_i^2 \left| H \, y_j \right|^2 \right] \tag{4.14}$$

Les termes $\left|\overline{x}(n)\right|^2$ et la valeur de $\sigma^2(n)$ sont fixes durant la recherche, minimiser D est équivalent à la minimisation de:

$$\overline{D} = -2 g_i \, p^T(n) \, y_j + g_i^2 E_j \tag{4.15}$$

où

$$p(n) = H^T \overline{x}(n) \; ; \tag{4.16}$$

$$E_j = \left| H y_j \right|^2 \tag{4.17}$$

E_j est l'énergie du j^{eme} mot code forme filtré et ne dépend pas du vecteur cible $\overline{x}(n)$.

Notons aussi que le mot code forme y_j est fixe et que la matrice H dépend seulement des

coefficients du filtre de synthèse et du filtre perceptuel, qui sont fixes durant une période de 4 vecteurs. Par conséquent E_j est aussi fixe durant cette même période.

Quand les deux filtres sont réactualisés, nous pouvons calculer et mémoriser les 128 termes d'énergie possible E_j, j = 0,1,...,127 (correspondants aux 128 mots code forme) et utiliser ces termes d'énergie d'une façon périodique pour la recherche dans le dictionnaire durant les 4 vecteurs parole suivants. Ceci, nous permet de réduire la complexité de recherche dans le dictionnaire.

Pour plus de réduction dans les calculs, nous pouvons pré-calculer et mémoriser les 2 vecteurs ; $b_i = 2g_i$ et $c_i = g_i^2$ pour i = 0,1,...,127, Ces deux vecteurs sont fixes vu que g_i est fixe.

La relation (4.15) devient alors:

$$\overline{D} = -b_i p_j + c_i E_j \qquad (4.18)$$

où $p_j = p^T(n)\ y_j$. (1.19)

Nous remarquons qu'une fois les tables E_j, b_i et c_i sont pré-calculées et mémorisées, le produit $p_j = p^T(n)\ y_j$, qui dépend de j, prend la majorité du taux de calcul dans la détermination de $\overline{D}$. La procédure de recherche dans le dictionnaire consiste alors à scruter dans le dictionnaire forme et identifier le meilleur indice i du gain pour chaque mot code forme y_j.

Il existe plusieurs manières de trouver le meilleur indice i du gain pour un mot code forme y_j donné :

- la première, et la plus évidente manière, est d'évaluer les 8 valeurs possibles de i et sélectionner l'indice i qui correspond à la plus petite valeur D. Cependant, cela demande 2 multiplications pour chaque i ;
- une seconde manière de procéder consiste à évaluer le gain optimal $\overline{g}=\frac{p_j}{E_j}$ en premier, et quantifier ce gain $\overline{g}$ à l'un des 8 niveaux de gain (g_0,.....,g_7) dans le dictionnaire gain de 3 bits. Le meilleur indice i est celui du niveau de gain g_i qui est le plus proche de $\overline{g}$. Seulement cette manière de procéder nécessite une opération de division pour chacun des mots codes, et la division est très déconseillée pour être implantée dans les cartes DSP ;

♦ une troisième approche, qui est une modification de la seconde, est particulièrement efficace à l'implantation DSP. La quantification de $\overline{g}$ peut être une série de comparaison entre $\overline{g}$ et les «frontières de cellule de quantification», qui sont les points milieu entre les niveaux de gain adjacents. Soit d_i le point milieu entre le niveau gain g_i et g_{i+1} qui ont le même signe. Alors tester « $\overline{g} < d_i$» est équivalent à tester $p_j < d_i E_j$. Ainsi, en utilisant ce test, on évite l'opération de division et on n'effectue qu'une opération de multiplication pour chaque indice i, cette approche est utilisée dans la recherche dans le dictionnaire. Les frontières de la cellule de quantification gain d_i sont peuvent ainsi être pré-calculées et mémorisées dans une table. Pour 8 niveaux de gain, on a seulement 7 valeurs limites d_0, d_1, d_2, d_3, d_4 et d_5 et d_6 qui sont utilisées.

Une fois que les meilleurs indices "i" et "j" sont identifiés, ils sont concaténés pour former la sortie du module de recherche dans le dictionnaire sous la forme d'un indice de 10 bits.

4.3 Quantification vectorielle de l'excitation

A 16 kbits/s nous avons 10 bits pour représenter chaque vecteur d'excitation de dimension 5 et comme l'analyse LPC est effectuée en backward, ces bits sont entièrement utilisés pour coder le signal d'excitation correspondant à l'entrée du filtre de synthèse. La complexité de recherche à travers un dictionnaire de 10 bits avec 1024 mot-codes indépendants est assez élevée. Pour réduire cette complexité, nous avons introduit la structure "gain-shape" et décomposé le dictionnaire en un produit de dictionnaire gain à 3 bits et un dictionnaire forme (shape) à 7 bits.

Au lieu que le dictionnaire forme ne soit "peuplé" par des nombres aléatoires gaussiens comme dans le CELP conventionnel, ce dictionnaire est optimisé en boucle fermée par un algorithme de conception basé sur le critère de perception d'erreur pondérée utilisé par l'encodeur LD-CELP. Ainsi, les contributions d'adaptation du prédicteur LPC et du gain sont prises en considération dans la conception.

L'algorithme de conception est similaire à celui de LBG (Linde Buzo Gray) [19] mais vue que la conception du dictionnaire est basée sur le critère perceptuel de pondération de l'erreur, l'encodeur entier est utilisé à chaque itération pour encoder la séquence d'apprentissage.

Les vecteurs d'apprentissage sont classés selon la règle de codage d'erreur minimum et les

mots codes sont réactualisés selon les conditions du centroïde dérivées ci-dessous [40].

Le dictionnaire "gain" est constitué d'un bit signe et de 2 bits pour le niveau d'amplitude. Le bit signe a pour rôle de doubler la taille du dictionnaire "forme" sans pour autant doubler la complexité de recherche dans celui-ci. Définissons :

$g(n)$ le niveau d'amplitude du gain du dictionnaire,

$\xi(n)$ le signe du gain,

N_j l'ensemble des indices temporels pour lesquels y_j est sélectionné comme meilleur mot code du dictionnaire ''forme'' durant le codage de la séquence d'apprentissage des vecteurs de parole d'entrée x_n , {n=1,…,N}.

Il s'agit de trouver le (i+1)^{ieme} nouveau dictionnaire forme qui minimise la i^{ieme} distorsion moyenne :

$$D(i) = \sum_{n=1}^{N} d(x_n, y_n(i)) \qquad (4..20)$$

La séquence de vecteurs cibles sont classés en K cellules, (N_j , j = 1, ..., K). L'équation est décomposée en K termes, un terme pour chaque cellule particulière ;

$$\begin{aligned} D = & \sum_{n\in N_1} \left\| x(n) - \xi(n)\sigma(n)g(n)H(n)y_1 \right\|^2 \\ & + \sum_{n\in N2} \left\| x(n) - \xi(n)\sigma(n)g(n)H(n)y_2 \right\|^2 \\ & + \ldots + \sum_{n\in N_K} \left\| x(n) - \xi(n)\sigma(n)g(n)H(n)y_K \right\|^2 \end{aligned} \qquad (4.21)$$

Chaque sommation englobe tous les vecteurs cibles qui sont classés avec le même mot-code ''forme''. La minimisation de l'équation selon le mot code correspondant est équivalente à minimiser séparément chaque terme de sommation, vu qu'un mot-code forme n'apparaît seulement que dans une seule sommation particulière.

La distorsion totale accumulée de la j^{ieme} classe correspondant à "y_j" est donnée par :

$$D_j = \sum_{n\in N_j} \left\| x(n) - \xi(n)\sigma(n)g(n)H(n)y_j \right\|^2 \qquad (4.22)$$

En prenant la dérivée partielle selon y_j et en l'annulant, nous obtenons :

$$\frac{\partial D_j}{\partial y_j} = -2 \sum_{n \in N_j} \sigma(n)\xi(n)g(n)H^T(n)x(n) + 2 \sum_{n \in N_j} \sigma^2(n)g^2(n)H^T(n)H(n)y_j = 0 \quad (4.23)$$

Alors, le nouveau centroïde y_j^* de la j^{ieme} classe qui minimise D_j satisfait l'équation suivante :

$$[\sum_{n \in N_j} \sigma^2(n)g^2(n)H(n)^T H(n)]\, y_j^* = \sum_{n \in N_j} \xi(n)\sigma(n)g(n)H^T(n)x(n) \quad (4.24)$$

Les deux sommations de cette équation normale sont accumulées séparément pour chacun des 128 mots de codes "forme" pour toute la séquence d'apprentissage.

On résout les 128 équations normales résultantes pour obtenir les 128 nouveaux centroïdes qui remplaceront ceux du dictionnaire (ancien). La séquence d'entraînement est ensuite codée avec ce nouveau dictionnaire. Cette procédure de réactualisation est répétée jusqu'à l'obtention d'un dictionnaire satisfaisant. Le problème majeur est que la conception en boucle fermée n'est pas garantie à converger du moment que la séquence d'apprentissage change d'une itération à une autre. En pratique, la distorsion globale décroît toujours durant les premières itérations, ce qui se traduit par une réduction significative dans la distorsion globale.

4.4 Conception du dictionnaire ″forme″

Lors de la conception du dictionnaire "forme", nous avons été amenés à choisir correctement le dictionnaire initial, ce dictionnaire joue un rôle essentiel dans la convergence de l'algorithme de conception vers un dictionnaire optimal en un nombre d'itérations non exhaustif.

Le dictionnaire d'excitation dont les vecteurs modélisent l'excitation du filtre de synthèse, lequel dans le cas idéal est représenté par le résiduel, a été peuplé par 128

vecteurs extraits des résiduels de la base de données (séquence d'apprentissage). La

nouvelle technique d'initialisation de l'algorithme GLA a été utilisée pour obtenir les 128 vecteurs mots codes de dimension 5 [20].

L'expression du résiduel est :

$$r_n = \sum_{k=0}^{p} a_k s_{n-k} \qquad n=0,\ldots, N-1 \quad a_0=1 \tag{4.25}$$

avec

s_n : signal parole (base de données) et a_k : les coefficients de prédiction.

Le calcul du résiduel a été fait en prenant un ordre de prédiction égal à 10, une trame de 240 échantillons sans recouvrement et une fenêtre de Hamming comme fenêtre de pondération. Les coefficients de prédiction ont été calculés en utilisant l'algorithme de Levinson-Durbin.

Une fois que la séquence résidu de la base de données parole a été obtenue, on applique la technique d'initialisation de l'algorithme GLA [20] pour obtenir les 128 vecteurs mot-code de dimension 5. Ces 128 mot-codes constitueront le dictionnaire initial utilisé pour démarrer la conception du dictionnaire "forme".

Tout l'encodeur LD-CELP est utilisé lors de l'encodage de la séquence d'apprentissage.

Les équations (4.24) à résoudre ont la forme suivante :

$$\mathbf{A.x = B} \tag{4.26}$$

Où A est une matrice carrée (5,5), B est un vecteur colonne (5,1) et x un vecteur colonne (5,1) de valeurs inconnues qu'il s'agit de déterminer.

La méthode d'obtention de la solution est de multiplier chaque coté de l'équation 4.26 par $\mathbf{A^{-1}}$:

$$\mathbf{x = A^{-1}.B} \tag{4.27}$$

Seulement pour la résolution des équations linéaires, cette solution est loin d'être la plus efficace et la plus stable numériquement.

En observant la structure de la matrice dans l'équation (4.24), on remarque que A peut se mettre sous la forme H^T H où H est une matrice carrée, triangulaire ayant les éléments de la diagonale principale différents de zéro. La matrice A possède une structure d'une matrice hermitienne. Nous avons utilisé la décomposition de Cholesky pour résoudre ces équations.

Tous les dictionnaires intermédiaires sont sauvegardés afin de choisir le dictionnaire donnant les meilleures performances.

La base de donnée de parole est constituée d'une série de huit phrases phonétiquement équilibrées de langue anglaise et française (quatre phrases prononcées par des locuteurs masculins et les quatre autres par des locuteurs féminins) d'une durée totale 30 secondes , soit 240 000 échantillons.

Le meilleur dictionnaire est celui donnant le **meilleur rapport signal à bruit segmental**.

4.5 Conception du dictionnaire ″gain″

Durant la conception du dictionnaire "forme", le gain utilisé pour la recherche du meilleur mot code est le gain optimal (non quantifié) déterminé par la formule suivante :

$$g_k = \frac{\left(\bar{x}^T H y_k\right)}{\left\| H y_k \right\|^2} \tag{4.28}$$

Pour procéder à la conception du dictionnaire du gain, nous avons créé une base de données constituée de gains optimaux obtenus lors du codage de la séquence d'apprentissage donnant le meilleur RSB segmental. Nous appliquons ensuite l'algorithme GLA pour avoir un dictionnaire de 3 bits dont 1 bit pour le signe.

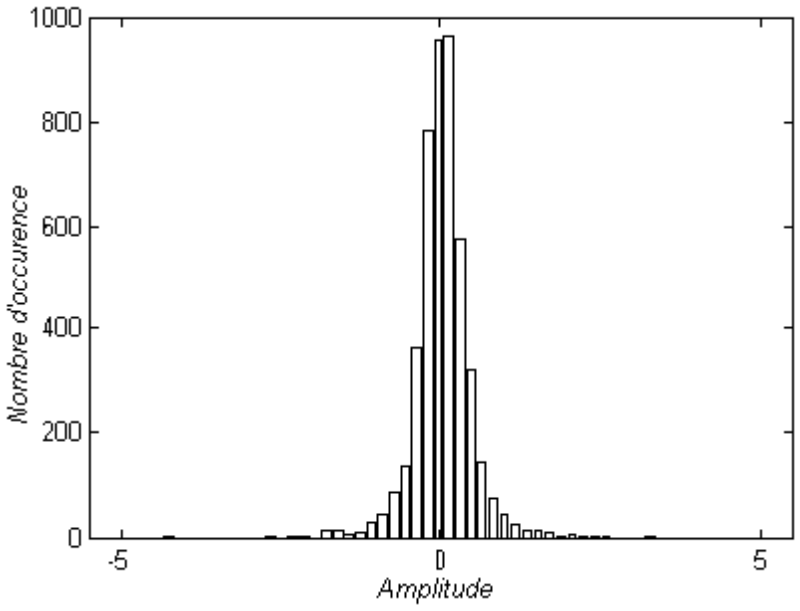


Fig 4.6 Histogramme du gain optimal obtenu durant la conception du dictionnaire "forme" donnant le meilleur RSB segmental.

Après plusieurs itérations, la convergence est atteinte. Les dictionnaires "gain" et "forme" obtenus sont sauvegardés dans un fichier de donnée et sont utilisés par l'encodeur LD-CELP.
Les mesures objectives et subjectives montrent que le codeur LD-CELP 16 kb/s atteint une qualité excellente (SNR segmental autour de 19 dB) avec un faible retard de codage(<2 ms).

4.6 Conclusion

Dans ce chapitre, nous avons donné une description technique de la mise en oeuvre des différents blocs du codeur/décodeur LD-CELP 16 kbits/s. Ce codeur possède un retard de codage dans un sens inférieur à 2 ms et cela grâce à l'adaptation en backward du prédicteur LPC ainsi que du gain d'excitation et en utilisant une taille réduite du vecteur d'excitation (5 échantillons). Le prédicteur pitch (pour les sons voisés) n'est pas utilisé mais est compensé par l'augmentation de l'ordre du prédicteur LPC de 10 (CELP conventionnel) à 50. Le gain d'excitation est réactualisé par un prédicteur linéaire adaptatif d'ordre 10 basé sur les gains logarithmiques des vecteurs d'excitation antérieurs. Le signal d'excitation est optimisé en boucle fermée. Durant la conception en boucle fermée, les contributions d'adaptation des prédicteurs LPC et gain sont prises en compte dans la conception. Enfin un filtre perceptuel de pondération est utilisé pour améliorer la qualité de la parole. Le codeur LD-CELP 16 kb/s permet d'avoir un signal de parole synthétique d'excellente qualité.

Chapitre 5

Le LD-CELP 8 KBITS/S

A partir du codeur (équivalent du G728) réalisé au CDTA (débit=16 kbits/s), notre travail consisterait à réduire le débit de moitié c'est-à-dire 8kbits/s. Le codeur G728 utilise 10 bits pour représenter chaque 5 échantillons de vecteur. Il est évident que pour réduire le débit de ce codeur, on doit soit réduire le nombre de bits utilisé pour chaque vecteur soit augmenter le nombre d'échantillons par vecteur. Si on garde la taille du vecteur fixe à 5 échantillons, à 8kbits/s on aura seulement 5 bits pour représenter les excitations forme et gain. Cela donnera un codeur ayant des performances inacceptables. Par conséquent pour réduire le débit de notre codeur nous avons augmenté la taille du vecteur.

5.1 Le LD-CELP 8 kb/s avec des trames de 10 échantillons

Pour le signal d'excitation un quantificateur vectoriel de 10 bits (7 bits forme, 3 bits gain) est utilisé. Le débit du codeur est modifié en changeant la taille du vecteur vs de vs=5 pour 16 kbits/s à vs=10 pour 8 kbits/s. Pour le G728, le filtre de synthèse, le filtre de pondération et le prédicteur de gain sont tous réactualisés une fois tous les quatre(04) vecteurs. Avec une taille de vecteur de 5, cela impliquera une réactualisation des filtres tous les 20 échantillons ou 2.5ms. Généralement plus la fréquence de réactualisation est grande et plus le codeur est performant. Cependant réactualisant les coefficients des filtres plus fréquemment entraînera une augmentation significative de la complexité du codeur. Ainsi avec l'augmentation de la taille du vecteur nous avons décidé de garder la période de réactualisation des filtres à 20 échantillons. Donc à 8 kbits/s la taille du vecteur est de 10 échantillons et les filtres sont réactualisés tous les 2 vecteurs.

Afin d'évaluer les performances de notre codeur, nous avons procédé à l'encodage d'une série de phrases de parole à bande étroite, phonétiquement équilibrées et de fréquence d'échantillonnage de 8 kHz. Ces phrases n'appartiennent pas à la séquence d'apprentissage utilisée pour la conception des dictionnaires ″forme″ et ″gain″ :

M1 : ″Annie s'ennuie loin de mes parents″ . **M2** : ″La voiture s'est arrêtée au feu rouge″.

M3 : ″Oh dear, the speaker apologized, I have been out of synchronization″.

M4 : "Dés que le tambour bat les gens accourent".

F1 : ″la vaisselle propre est mise sur l'évier″. **F2** : ″Huit satellites ont été mobilisés″.

F3 : ″I must have reread the article three times before I realized what was bothering me″.

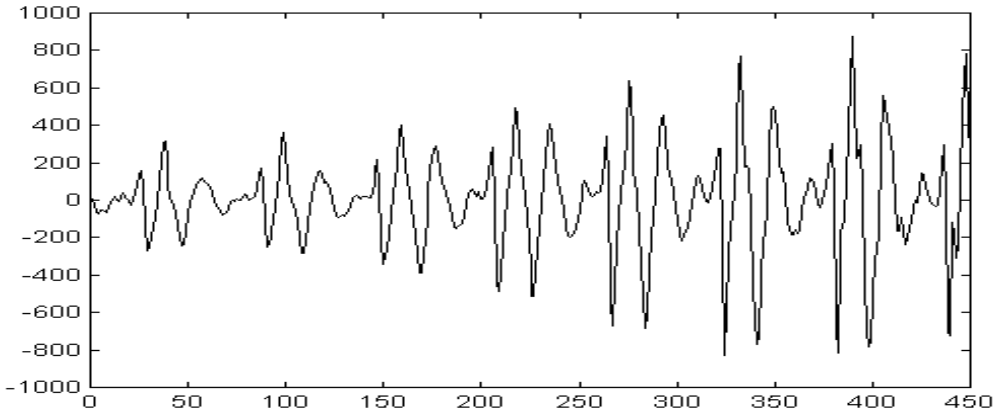
F4 : " American don't make a move anymore without checking first with greater trust".

La figure 5.1. nous donne une comparaison d'un segment de forme d'onde de parole tiré de la phrase M3.

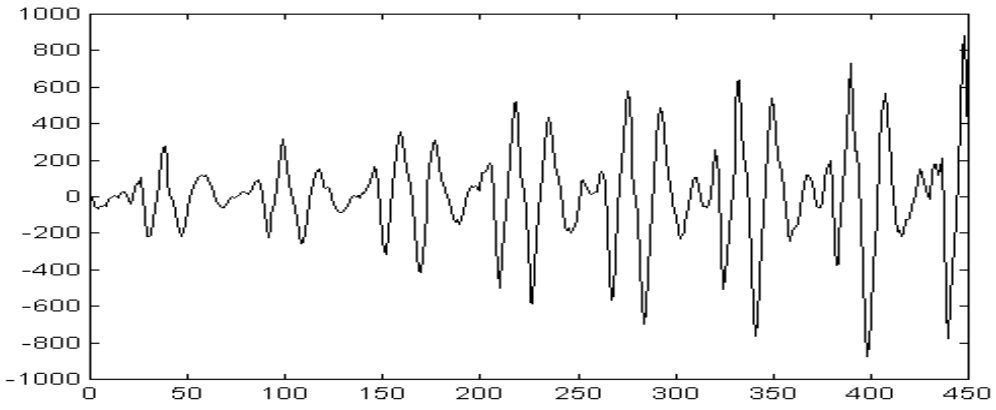
La figure 5.2. nous donne un exemple de codage de la phrase F2, son signal synthétique obtenu, le signal excitation ainsi que le signal erreur.

Le tableau 5.1 donne les performances objectives du codeur. Nous notons une réduction significative du rapport signal à bruit segmental (SNRSEG) comparativement au LD-CELP 16kb/s. Cette détérioration significative des performances était prévisible du moment qu'à 8kb/s nous travaillons à 1 bit par échantillon (10 bits pour 10 échantillons) alors qu'à 16kb/s, nous avons 2 bits par dimension (10 bits pour 5 échantillons).

Les moyennes des RSB et RSB segmentaux obtenues (pour les 8 phrases prises en même temps) sont respectivement égales à 15.11 dB et 12.88 dB . Les tests d'écoute ont confirmé l'efficacité du codeur/décodeur pour l'obtention de la bonne qualité.

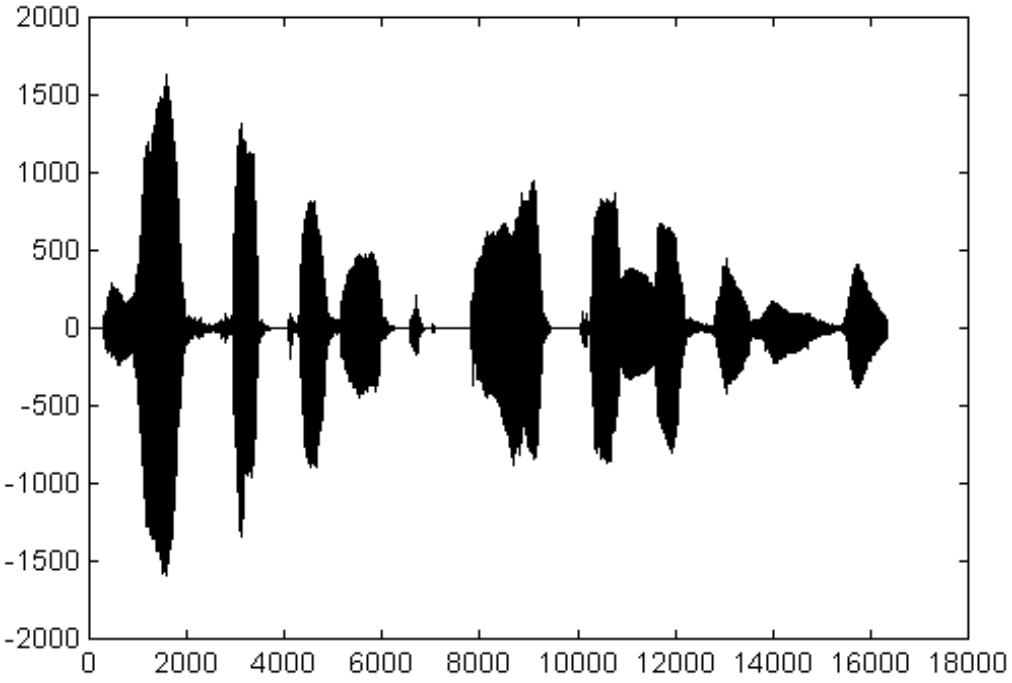


Signal original

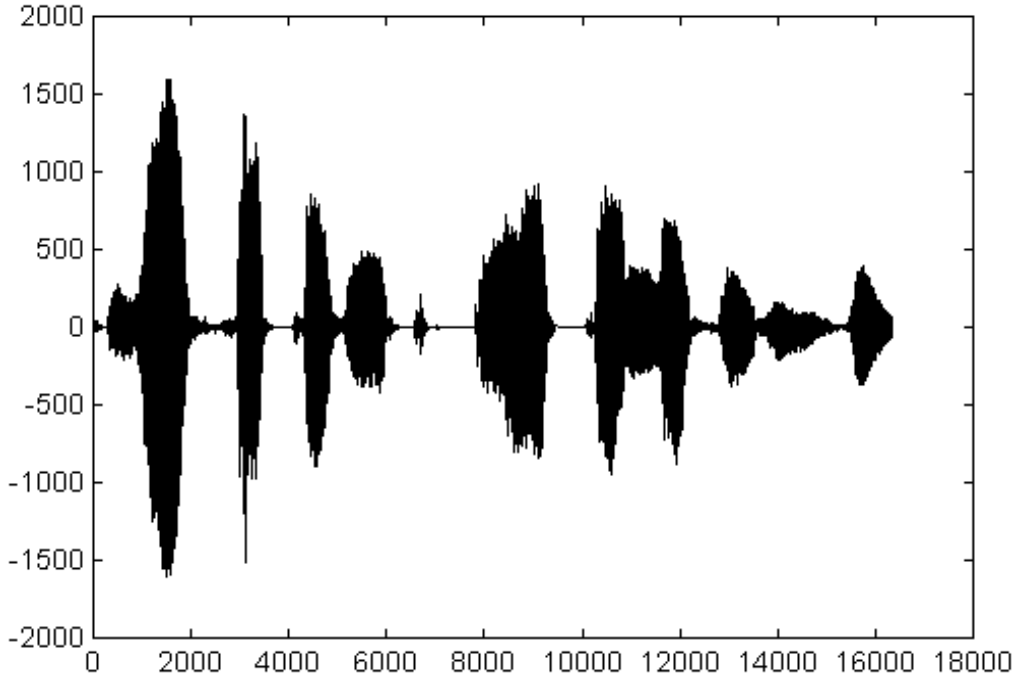


Signal synthétique

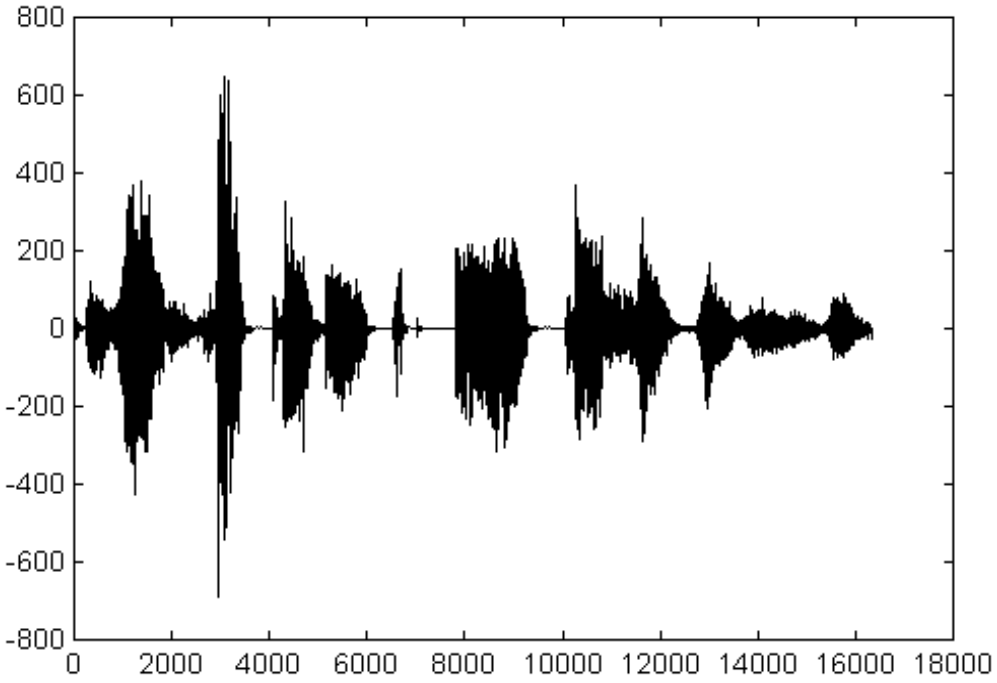
Fig 5.1 Comparaison de forme d'onde d'un segment de parole de la phrase M3.



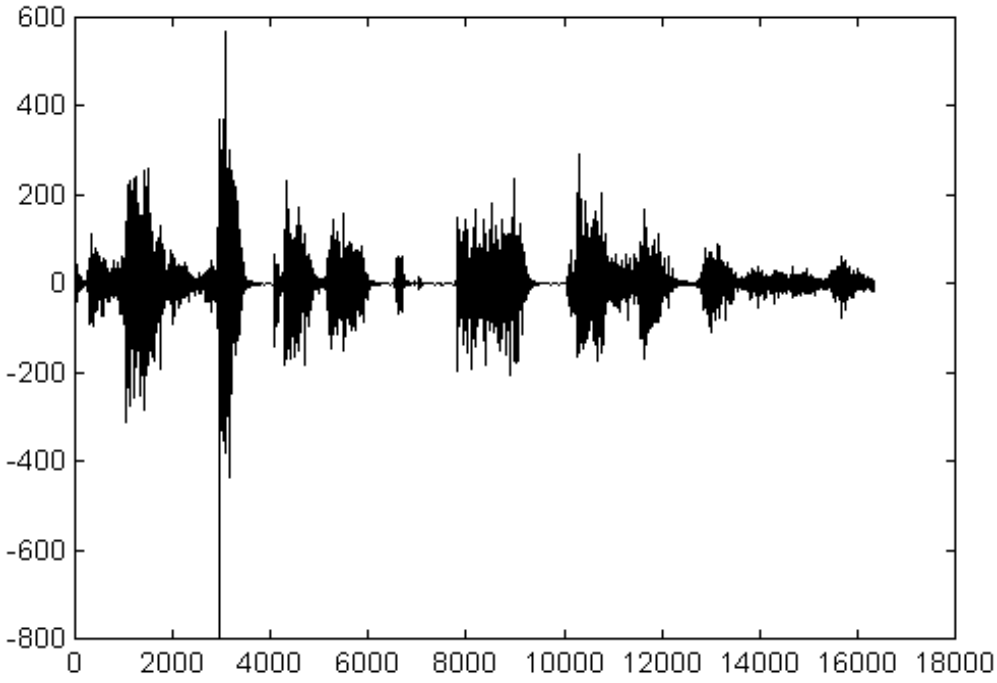
(a)- Signal original



(b)- Signal synthétique



(c) - Signal d'erreur pondéré



(d)- Signal d'excitation

Fig 5.2 Exemple de phrase codée F2 **a)** signal original **b)** signal synthétique **c)** signal d'erreur pondéré **d)** signal d'excitation .

Phrases	SNR(dB)	SNRSEG(dB)
M1	13.44	12.35
M2	13.10	11.09
M3	13.30	11.44
M4	13.51	12.52
F1	15.00	12.44
F2	16.90	14.24
F3	16.03	14.25
F4	15.95	13.38

Tableau 5.1 Performances du codeur LD-CELP 8kb/s
(7 bits pour la forme + 3 bits pour le gain et sans prédicteur long-terme LTP).

Le dictionnaire stochastique a été optimisé en **boucle fermée**.

5.2 Le Low Delay 8 kb/s avec prédicteur long-terme (LTP) adaptatif

5.2.1 Paradigme du dictionnaire adaptatif

Le filtre de synthèse pitch modélise les corrélations long-terme du signal de parole en répétant les segments de l'excitation antérieure reconstruite dite aussi excitation passée. Cette dernière est définie ici comme étant l'excitation reconstruite avant la trame ou sous-trame courante. Idéalement, l'excitation passée est retardée d'un intervalle égal à la période du pitch comme cela est montré à la figure 5.3 :

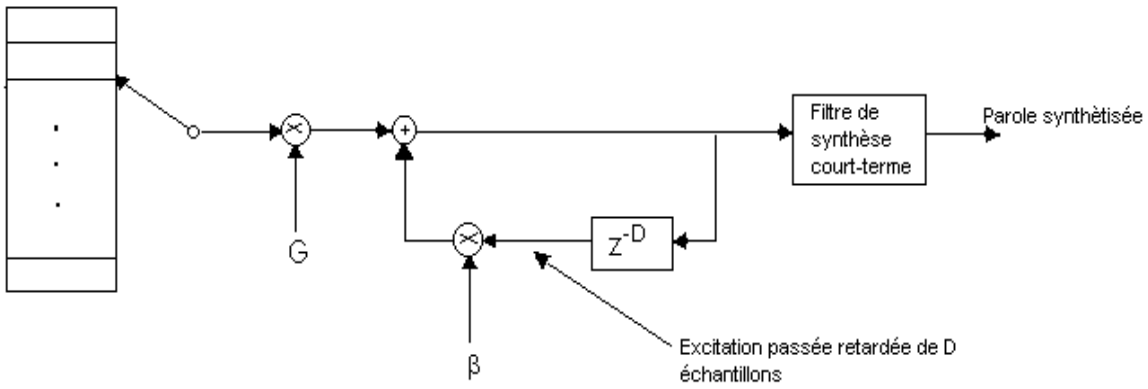


Fig. 5.3 Synthèse Pitch dans les codeurs CELP vue comme une opération de filtrage.

Cette opération peut être vue comme une sélection d'un vecteur à partir d'un **dictionnaire adaptatif** [46], comme illustré à la figure 5.4. Dans cette interprétation, L'excitation est une combinaison linéaire d'une contribution d'un dictionnaire adaptatif et d'un dictionnaire fixe.

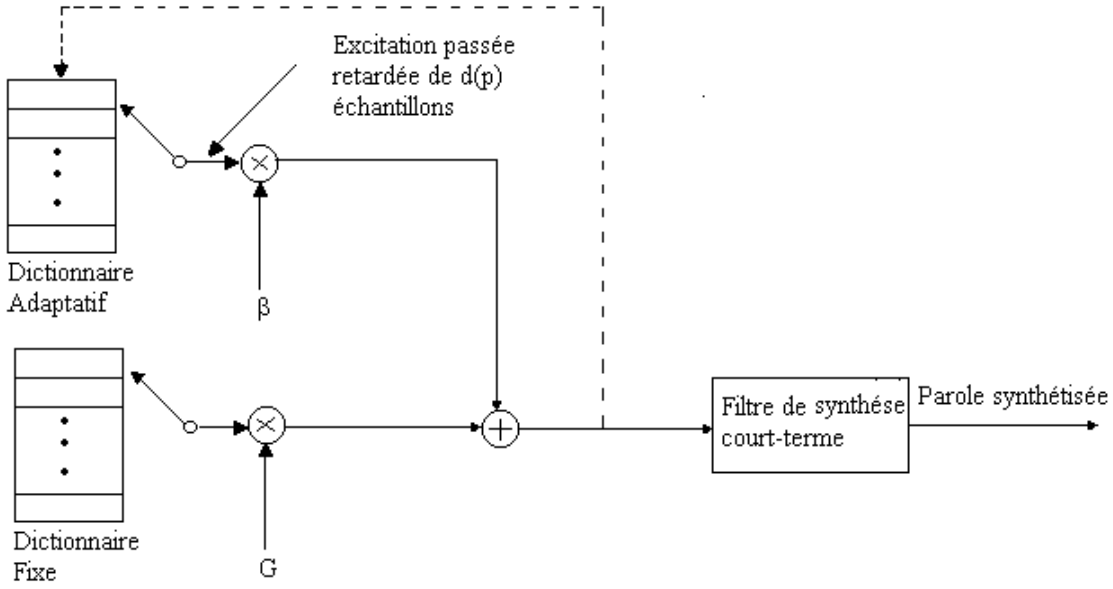


Fig. 5.4 Analyse par synthèse dans un codeur CELP avec dictionnaire adaptatif.

5.2.2 Définition et réalisation

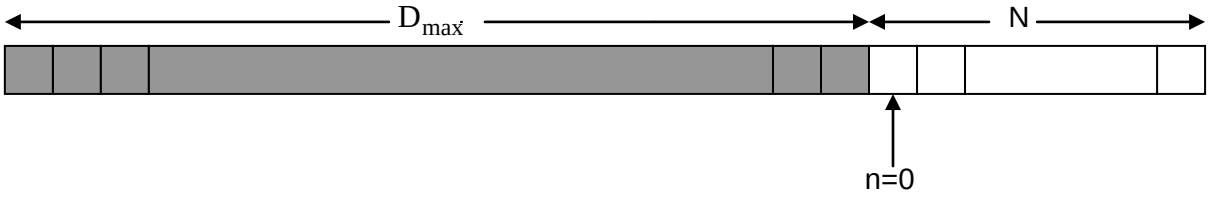
Le dictionnaire adaptatif emmagasine les vecteurs (blocs d'échantillons) représentant un segment de l'excitation passée à un délai spécifique. Le vecteur du dictionnaire adaptatif v[n], n=0,…,N-1, est construit à partir du signal d'excitation e[n] défini pour $-D_{max} \le n < 0$,

où D_{max} est le retard maximal alloué au dictionnaire adaptatif (période pitch), typiquement 147 échantillons pour une fréquence d'échantillonnage de 8 kHz. Donc le vecteur v[n] du dictionnaire adaptatif pour $n \ge 0$ est obtenu récursivement par :

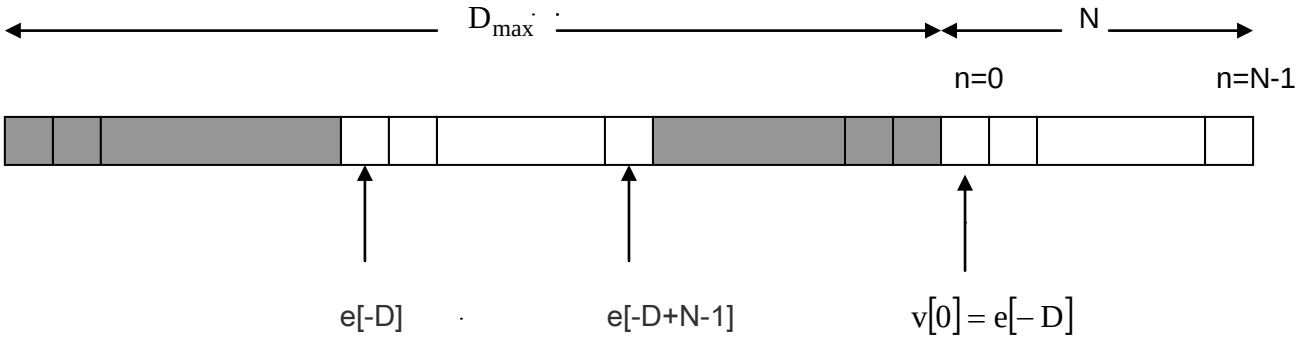
$$v[n] = e[n - d(p)] \qquad \text{pour } n = 0,1,...,N-1, \qquad (5.1)$$

où d(p) est le retard ou délai pour l'entrée indexée à p du dictionnaire adaptatif. La séquence de données v[n], $n = 0,1,...,N-1,$ forme l'entrée du dictionnaire adaptatif pour le retard d(p). En pratique, un champ d'échantillons est spécifié, d'une taille égale au moins à $D_{max} + N$.

Les premiers D_{max} échantillons représentent l'excitation construite passée et les N échantillons suivants représentent l'excitation pour la trame courante (figures 5.5 et 5.6).



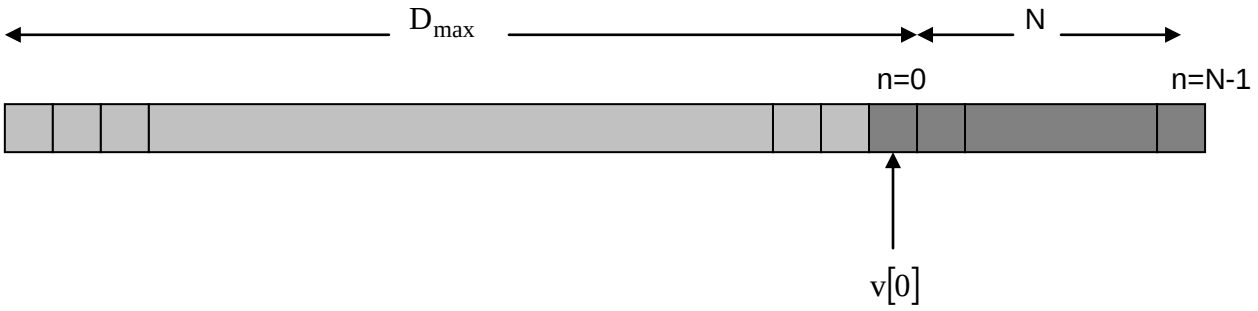
(a) Etat du dictionnaire adaptatif avant sélection de l'excitation



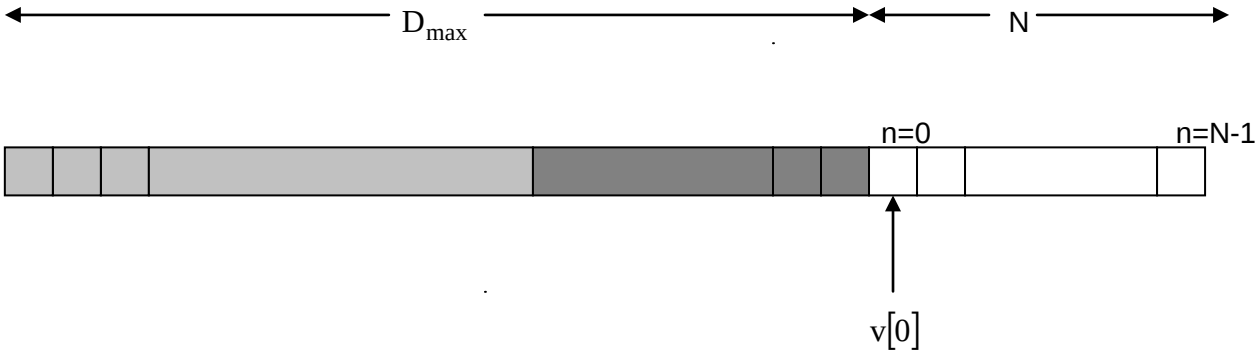
(b) Sélection d'un segment à partir d'une excitation construite passée ($D \geq N$).

Fig. 5.5 Une alternative pour la procédure de sélection du dictionnaire adaptatif.

Comme cela est montré à la figure 5.6, le dictionnaire adaptatif est réactualisé toutes les sous-trames en décalant son contenu par N échantillons vers la gauche. Ce décalage est effectué après que l'excitation résultante soit copiée à la sous-trame courante. L'excitation résultante est définie ici comme étant la somme de la contribution mise à échelle du dictionnaire adaptatif et de celle du dictionnaire fixe.



(a) Excitation résultante.



(b) Réactualisation du dictionnaire adaptatif

Fig 5.6 Procédure de réactualisation du dictionnaire adaptatif.

Si la période du pitch est supérieure à la taille de la sous-trame (ce qui est le cas pour nous), la procédure du dictionnaire adaptatif est similaire à la prédiction du pitch en boucle fermée (approche *filtrage*). Dans le cas où la période du pitch est inférieure à la taille de la sous-trame nous rencontrons des problèmes de calcul. Il existe plusieurs alternatives pour traiter cela :

1- Dans l'approche du filtrage, les échantillons sont calculés de façon récursive.

2- Dans la procédure du dictionnaire adaptatif, ce cas est traité différemment [46]. La séquence représentant la contribution du dictionnaire adaptatif (sans la mise à échelle) se répète d'elle même lorsque le retard d(p) est inférieur la longueur N de la sous-trame.

La procédure du dictionnaire adaptatif diffère de l'approche de filtrage essentiellement dans la mise à l'échelle relative des échantillons répétés. Dans la procédure du dictionnaire adaptatif, on attribue aux échantillons répétés la même pondération β que le reste des échantillons dans la séquence alors que dans l'approche de filtrage, les échantillons correspondants sont mis à l'échelle par β^2.

5.2.3 Détermination de l'excitation

Nous allons décrire une procédure pratique pour la sélection de l'excitation qui va attaquer le filtre de synthèse court-terme. C'est une méthode analytique qui permet de trouver le vecteur optimal ainsi que le gain du dictionnaire adaptatif. L'excitation finale résulte de la contribution des dictionnaires fixe et adaptatif.

Notons $s_w = [s[0],...,s[N-1]]^T$ le vecteur de parole pondéré et $e = [e[0],...,e[N-1]]^T$ le vecteur excitation résultant. Notons aussi v_p un vecteur du dictionnaire adaptatif d'index p où

$$v_p[n] = e[n - d(p)] = e[n - D], \qquad n=0,\ldots,N-1, \qquad (5.2)$$

et β le facteur de mise à l'échelle correspondant. De plus soit respectivement c_i et G le vecteur du dictionnaire fixe indexé par i et son gain associé. Donc l'excitation résultante est donnée par :

$$e_{ip} = \beta v_p + G c_i . \qquad (5.3)$$

Les vecteurs ainsi que les facteurs de mise à l'échelle sélectionnés sont ceux qui offrent l'erreur quadratique moyenne minimale. Pour avoir des performances optimales, les dictionnaires fixe et adaptatif doivent être scrutés simultanément. Cependant, afin de réduire la charge de calcul qui en découle, la recherche est souvent effectuée séquentiellement en commençant d'abord par le dictionnaire adaptatif. Ce choix est justifié du fait que le dictionnaire adaptatif fournit normalement la plus grande contribution à l'excitation résultante dans la parole voisée. Finalement, l'excitation optimale résultante est exprimée par :

$$e_{opt} = \beta_{opt} v_{opt} + G_{opt} c_{opt} . \qquad (5.4)$$

Pour sélectionner une entrée à partir du dictionnaire adaptatif, chaque vecteur v_p est filtré à travers le filtre de synthèse pondéré court-terme.

Soit $\hat{s}_p$ le vecteur parole synthétique pondéré pour un index p du dictionnaire adaptatif et $\hat{s}_0$ la réponse à entrée nulle du filtre de synthèse court-terme pondéré. Donc l'opération de filtrage peut être décrite comme :

$$\hat{s}_p = \beta H v_p + \hat{s}_0 \tag{5.5}$$

où

$$H=\begin{bmatrix} h_0 & 0 & 0 & . & . & . & 0 \\ h_1 & h_0 & 0 & . & . & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ h_{N-2} & h_{N-3} & h_{N-4} & . & . & . & 0 \\ h_{N-1} & h_{N-2} & h_{N-3} & . & . & . & h_0 \end{bmatrix}$$

et $\{h_0, h_1, h_2, \ldots, h_{N-1}\}$ représente la réponse impulsionnelle du filtre de synthèse pondéré court-terme. La différence quadratique moyenne entre les vecteurs parole (original et synthétique) pondérés doit être minimisée. Le critère d'erreur s'écrit :

$$\begin{aligned} \varepsilon_0 &= \left| s_w - \hat{s}_p \right|^2 \\ &= \beta^2 v_p^T H^T H v_p - 2\beta v_p^T H^T (s_w - \hat{s}_0) + \left| s_w - \hat{s}_0 \right|^2, \end{aligned} \tag{5.6}$$

où le dernier terme reste constant durant la recherche. Le gain optimal pour un vecteur particulier du dictionnaire peut être déterminé en minimisant ε_0 par rapport à β. Ceci est fait en dérivant ε_0 par rapport à β puis en annulant l'équation résultante ce qui donne :

$$\beta_{opt} = \frac{v_p^T H^T (s_w - \hat{s}_0)}{v_p^T H^T H v_p}. \tag{5.7}$$

En remplaçant la valeur optimale de β dans l'équation (5.6) et en omettant le terme constant, du moment qu'il ne dépend d'aucun paramètre à optimiser, le critère d'erreur est réduit à :

$$\varepsilon_1 = - \frac{\left[v_p^T H^T (s_w - \hat{s}_0)\right]^2}{v_p^T H^T H v_p}. \qquad (5.8)$$

Le vecteur $(s_w - \hat{s}_0)$ est appelé **vecteur cible**. La valeur de ε_1 est calculée pour chaque vecteur du dictionnaire. Sachant que pour chaque vecteur du dictionnaire adaptatif on a un retard lui correspondant, le délai qui donne la plus petite valeur de ε_1 est sélectionné.

En terme d'échantillons, les équations (5.7) et (5.8) peuvent être réécrites de la manière suivante. Soit s_t le vecteur cible. Donc le numérateur de l'équation (5.7) peut s'écrire comme $v_p^T H^T s_t$ ou bien $(Hv_p)^T s_t$. Mais Hv_p représente le vecteur parole pondéré synthétisé pour une entrée d'indice p du dictionnaire adaptatif noté $\hat{s}_p'$ Ainsi, nous avons :

$$\beta_{opt} = \frac{(Hv_p)^T s_t}{(Hv_p)^T Hv_p} = \frac{\sum_{n=0}^{N-1} s_t[n]\hat{s}_p'[n]}{\sum_{n=0}^{N-1} \hat{s}_p'[n]\hat{s}_p'[n]}, \qquad (5.9)$$

et

$$\varepsilon_1 = -\frac{\left[(Hv_p)^T s_t\right]^2}{(Hv_p)^T Hv_p} = -\frac{\left[\sum_{n=0}^{N-1} s_t[n]\hat{s}_p'[n]\right]^2}{\sum_{n=0}^{N-1} \hat{s}_p'[n]\hat{s}_p'[n]}. \qquad (5.10)$$

La recherche dans le dictionnaire fixe s'effectue de la même manière. La seule différence est que la contribution du dictionnaire adaptatif est maintenant prise en compte.

5.2.4 Le low delay 8 k/bs avec prédicteur long-terme adaptatif en backward

Dans un système adaptatif en forward, les coefficients du filtre de synthèse court-terme sont déterminés en minimisant l'énergie du signal résiduel trouvé en filtrant la parole originale à travers le filtre de synthèse inverse. De la même façon pour le prédicteur long-terme en boucle ouverte, nous minimisons l'énergie du signal résiduel long-terme lequel est déterminé en filtrant le résiduel court-terme r(n) à travers le prédicteur long-terme inverse. La minimisation de l'énergie du signal résiduel long-terme E_{LT} par un prédicteur d'ordre 1 est donnée par :

$$E_{LT} = \sum_{n=1}^{ns} \left(r(n) - \beta r(n-L)\right)^2 . \qquad (5.11)$$

où ns représente la longueur de la séquence de signal r(n) sur laquelle nous déterminons la période L (ns=20 dans notre cas). Le meilleur délai L est trouvé en calculant

$$X = \frac{\left(\sum_{n=1}^{ns} r(n)\, r(n-L)\right)^2}{\sum_{n=1}^{ns} r^2(n-L)} \qquad (5.12)$$

pour tous les retards possibles et choisir la valeur de L qui maximise X. Le meilleur facteur de gain β est donné par

$$\beta = \frac{\sum_{n=1}^{ns} r(n)\, r(n-L)}{\sum_{n=1}^{ns} r^2(n-L)} . \qquad (5.13)$$

Dans le système adaptatif *backward* le signal de parole original s(n) n'est pas disponible ce qui fait qu'à sa place on utilise le signal de parole précédemment reconstruit $\hat{s}(n)$ pour estimer les coefficients du filtre de synthèse court-terme. Ces coefficients peuvent être utilisés pour filtrer $\hat{s}(n)$ à travers le filtre inverse et obtenir le signal "résiduel reconstruit" $\hat{r}(n)$. Ce signal résiduel peut être utilisé dans les équations (5.12) et (5.13) pour calculer le délai et le gain du prédicteur long-terme. Alternativement nous pouvons utiliser le *signal d'excitation passé* u(n) dans les équations (5.12) et (5.13) et c'est ce qui été fait dans notre travail. Cette approche est légèrement plus simple par rapport à l'utilisation du signal résiduel reconstruit du fait que le filtrage inverse de $\hat{s}(n)$ pour calculer $\hat{r}(n)$ n'est plus nécessaire et on peut facilement constater que les deux approches donnent des résultats presque identiques.

Le meilleur retard (ou délai) est donc calculé en maximisant l'expression

$$X = \frac{\left(\sum_{n=1}^{ns} u(n)\, u(n-L)\right)^2}{\sum_{n=1}^{ns} u^2(n-L)} \qquad (5.14)$$

sur un intervalle de retards allant de 20 à 147 pour chaque trame. Le gain du prédicteur LTP est donnée par

$$\beta = \frac{\sum_{n=1}^{ns} u(n)\,u(n-L)}{\sum_{n=1}^{ns} u^2(n-L)} \quad . \tag{5.15}$$

5.3 Résultats et tracé des signaux originaux et synthétiques.

Dans cette section, on décrit les améliorations apportées à notre 8kb/s LD-CELP en rajoutant un prédicteur long-terme adaptatif backward. Ce travail est motivé du fait de la présence de corrélations long-terme dans le résiduel du filtre de synthèse. Nous avons donc augmenté la taille de la trame à 20 échantillons (2.5 ms de bufférisation). La complexité de recherche à travers un dictionnaire de 20 bits est réduite en adoptant la structure ''forme-gain''. Le dictionnaire de 20 bits est donc décomposé en deux dictionnaires :le premier avec 14 bits pour la quantification de la forme et le deuxième avec 6 bits (1 bit pour le signe et 5 bits pour l'amplitude) pour la quantification du gain. De plus, pour éviter l'opération de stockage, nous avons utilisé dans un premier temps un dictionnaire *forme* algébrique de 14 bits.

5.3.1 Le LD-CELP avec dictionnaire algébrique et LPC=50.

Le dictionnaire est donné par le tableau suivant :

Numéro de l'impulsion i	Amplitude	Position possible
1	**(-1 , +1)**	**1 , 6 , 11 , 16**
2	**-1**	**2 , 7 , 12 , 17**
3	**(-1 , +1)**	**3 , 8 , 13 , 18**
4	**(-1 , +1)**	**4 , 9 , 14 , 19**
5	**(+1 , +1)**	**5 , 10 , 15 , 20**

Tableau 5.2 Amplitudes et positions des impulsions pour le dictionnaire forme 14 bits Low-Delay ACELP.

Chaque vecteur de 20 échantillons possède un signal donné par 5 impulsions non nulles d'amplitude +1 ou –1 dont les positions sont montrées dans le tableau ci-contre 5.2. Les positions d'impulsion 1,3,4 et 5 sont codées sur 3 bits chacune car prenant une fois l'amplitude +1 et l'autre fois –1 alors que la position 2 l'est sur 2 bits seulement (avec l'amplitude –1). Donc tous ont 2 bits pour la position mais les impulsions 1,3,4 et 5

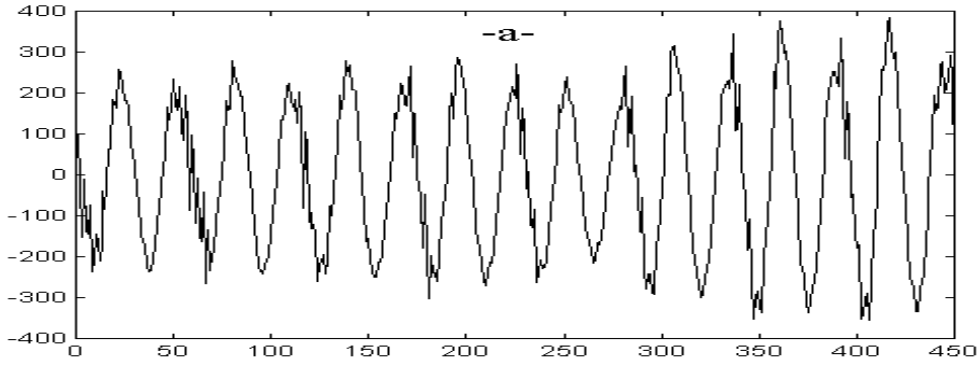
nécessitent chacune 1 bit supplémentaire pour le signe ce qui donne un total de 14 bits pour le dictionnaire. La position adéquate des impulsions est déterminée en utilisant une série de 9 boucles imbriquées donnant ainsi un algorithme de recherche dans le dictionnaire très efficace.

Le tableau 5.3 donne les performances objectives de ce codeur avec les trames de 20 échantillons. L'ordre du prédicteur LPC backward étant égal à 50 et les phrases étant les mêmes que celle du paragraphe précédent.

Phrases	**SNR(dB)**	**SNRSEG(dB)**
M1	15.14	14.17
M2	14.81	12.18
M3	14.61	12.58
M4	14.93	13.94
F1	16.16	13.40
F2	18.09	15.03
F3	17.26	15.37
F4	17.30	14.32

Tableau 5.3 Performances objectives du LD-CELP 8kb/s avec des trames de 20 échantillons, un dictionnaire algébrique de 14 bits et LPC=50.

La figure 5.7. nous donne une comparaison d'un segment de forme d'onde de parole tiré de la phrase M1.



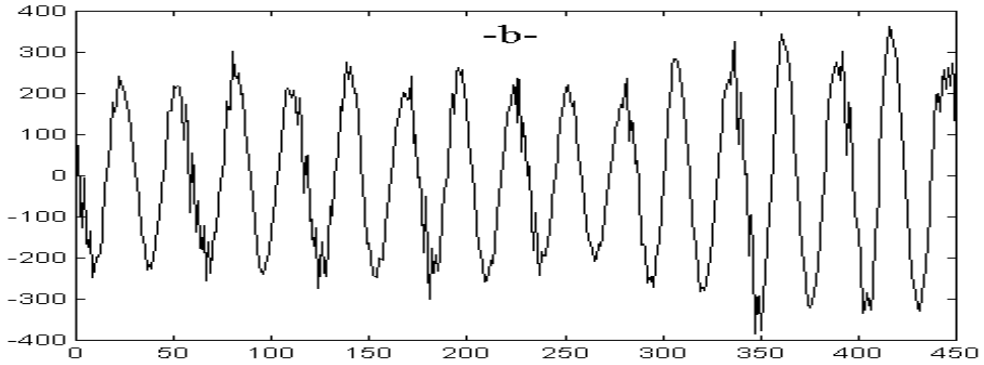
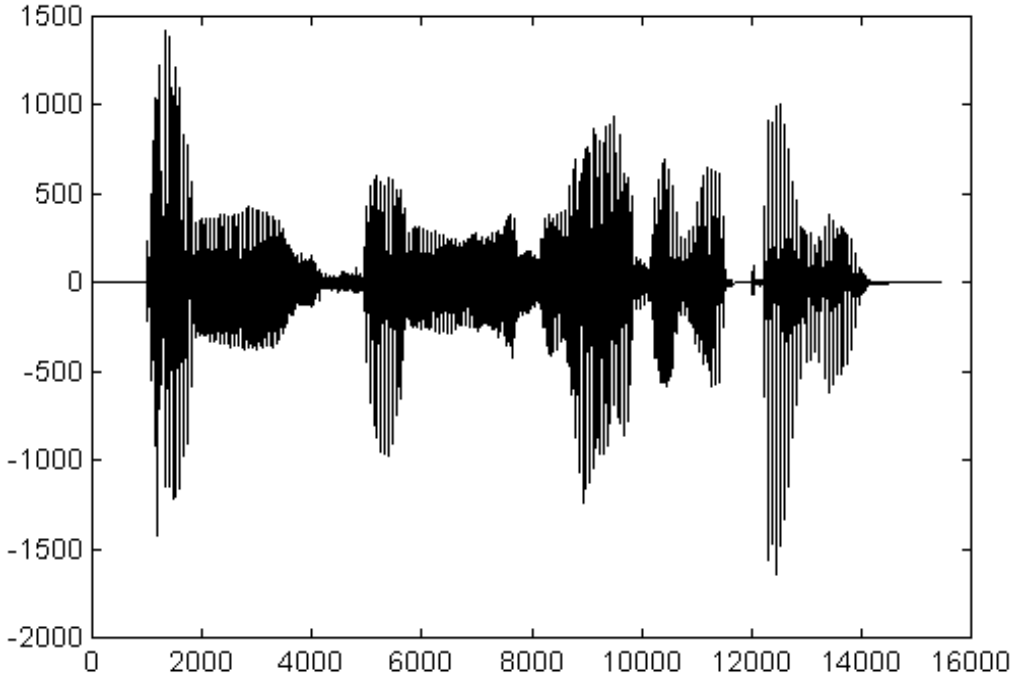
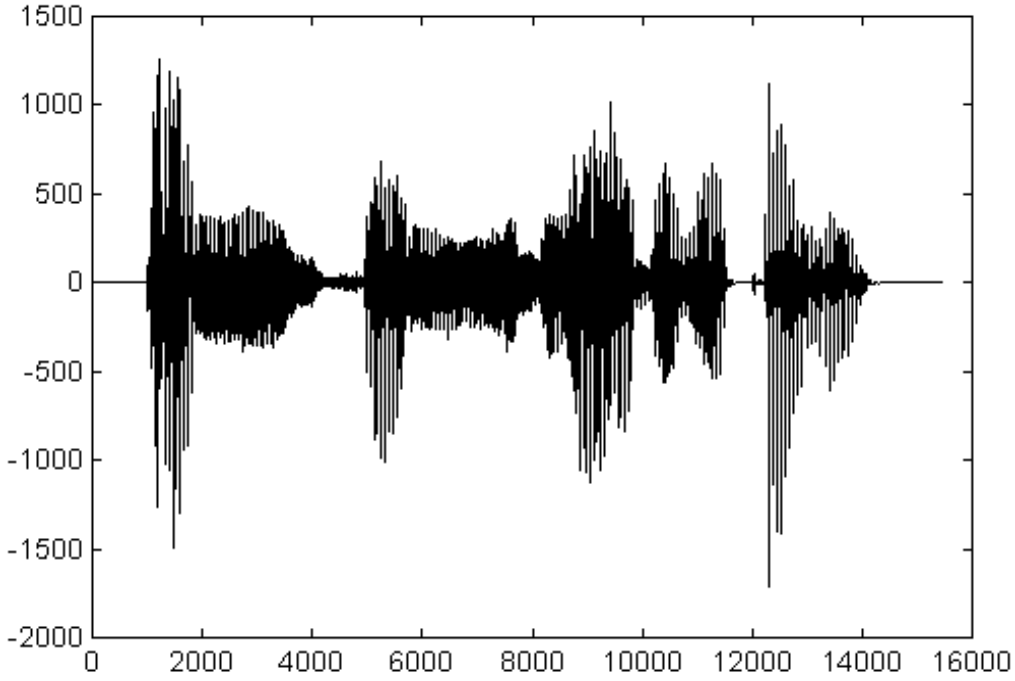


Fig 5.7 Comparaison de forme d'onde d'un segment de parole de la phrase M1.
(a)- signal original. (b)- signal synthétique

La figure 5.8. nous donne un exemple de codage de la phrase M1 et le signal synthétique obtenu.



-a-



-b-

Fig 5.8 Exemple de phrase codée M1 **a)** signal original. b) signal synthétique.

La figure 5.9 nous donne l'évolution du RSB en dB par rapport à la puissance du signal original pour la phrase M1.

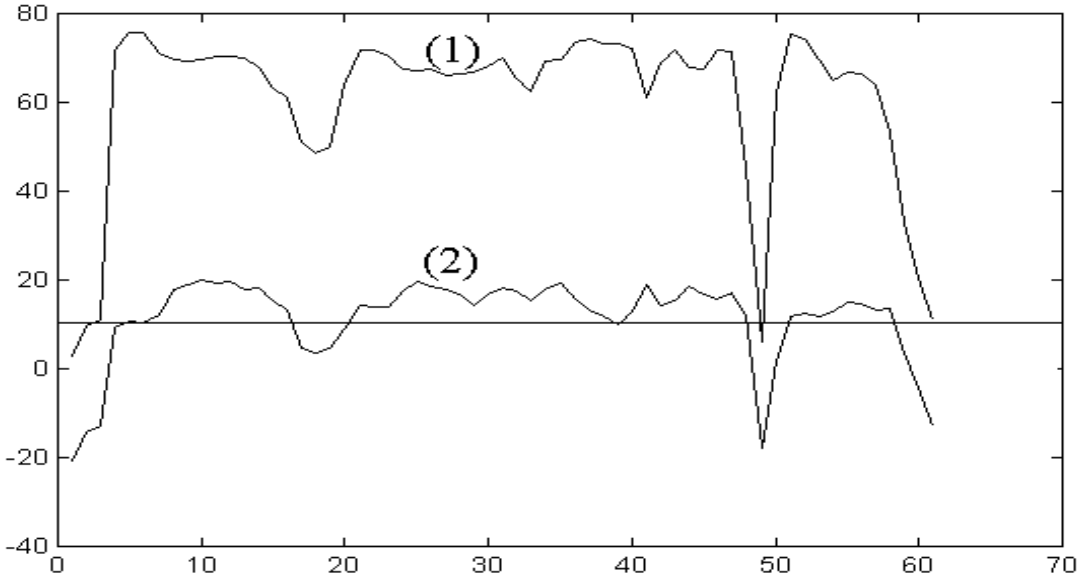


Fig 5.9 *(*1) Puissance du signal de parole. (2) Evolution du RSB en fonction du temps (trames de 240 échantillons).

5.3.2 Le LD-CELP avec dictionnaire algébrique et LPC=20.

Afin de résoudre le problème de la corrélation long-terme, nous avons procédé de la manière suivante : réduire d'abord l'ordre du prédicteur LPC de 50 à 20 puis rajouter un prédicteur long-terme adaptatif en backward. Donc en réduisant l'ordre du prédicteur LPC à 20, nous obtenons les résultats résumés dans le tableau ci-contre (le dictionnaire *forme* étant le dictionnaire algébrique de 14 bits) :

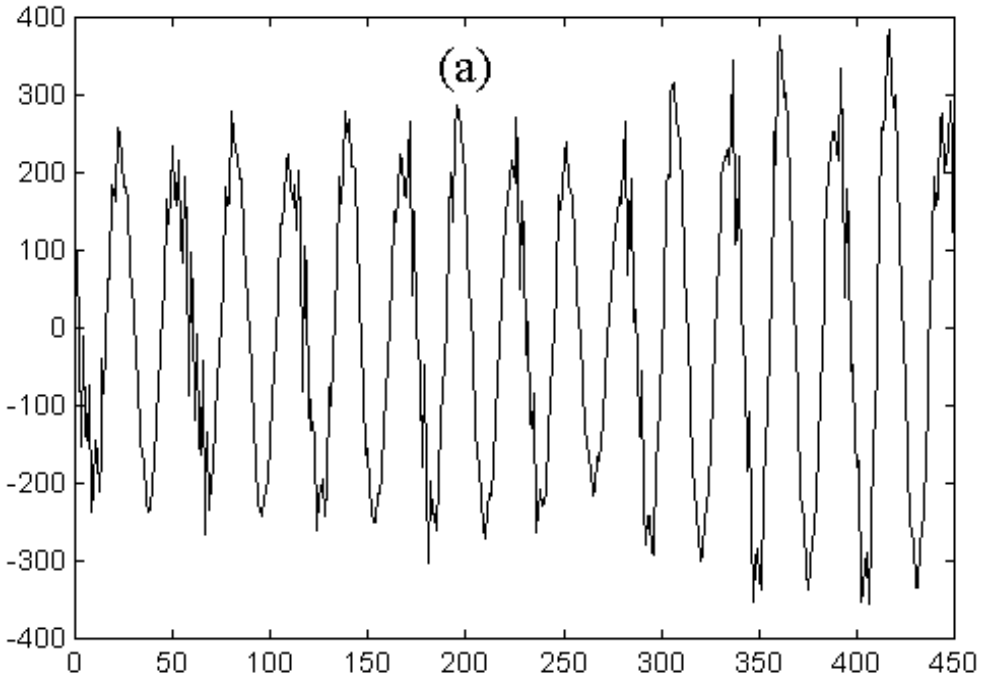
Phrases	**SNR(dB)**	**SNRSEG(dB)**
M1	14.62	13.80
M2	14.76	12.10
M3	14.50	12.39
M4	14.66	13.85
F1	15.62	12.88
F2	17.01	14.24
F3	16.45	14.50
F4	16.34	13.68

Tableau 5.4 Performances objectives du LD-CELP 8kb/s avec des trames de 20 échantillons, un dictionnaire algébrique de 14 bits et LPC=20.

Nous observons donc à partir des deux tableaux 5.3 et 5.4 que lorsque l'ordre de prédiction du filtre LPC est réduit de 50 à 20, nous avons une petite réduction en terme de performance. Le SNR segmental est en moyenne diminué de 0.44 dB. A partir de là nous concluons donc que pour avoir une bonne qualité de parole à 8 kb/s avec la contrainte du faible retard nous devons, exploiter les redondances du pitch de manière explicite. Pour cela, nous avons réduit l'ordre du prédicteur LPC à 20 et utilisé un prédicteur long-terme adaptatif backward.

La figure 5.10. nous donne une comparaison d'un segment de forme d'onde de parole tiré de la phrase M1 avec LPC=20.

La figure 5.11. nous donne un exemple de codage de la phrase M1 et son signal synthétique obtenu. La figure 5.12 nous donne l'évolution du rapport signal à bruit en fonction du temps de la phrase M1 en prenant des trames de 240 échantillons.


(a)
400
300
200
100
0
-100
-200
-300
-400
0
50
100
150
200
250
300
350
400
450

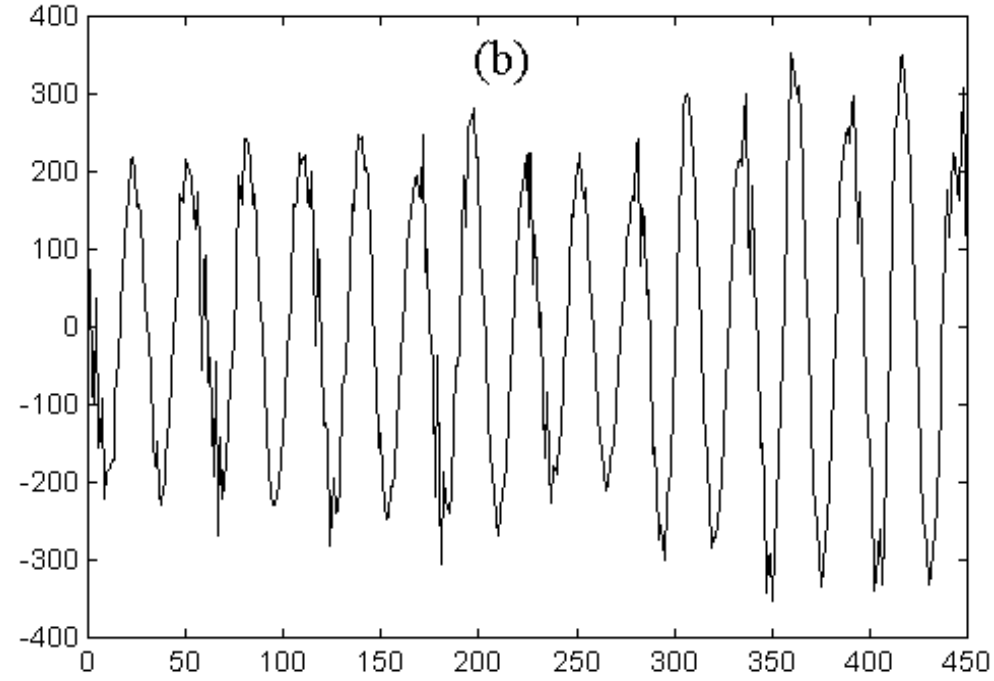
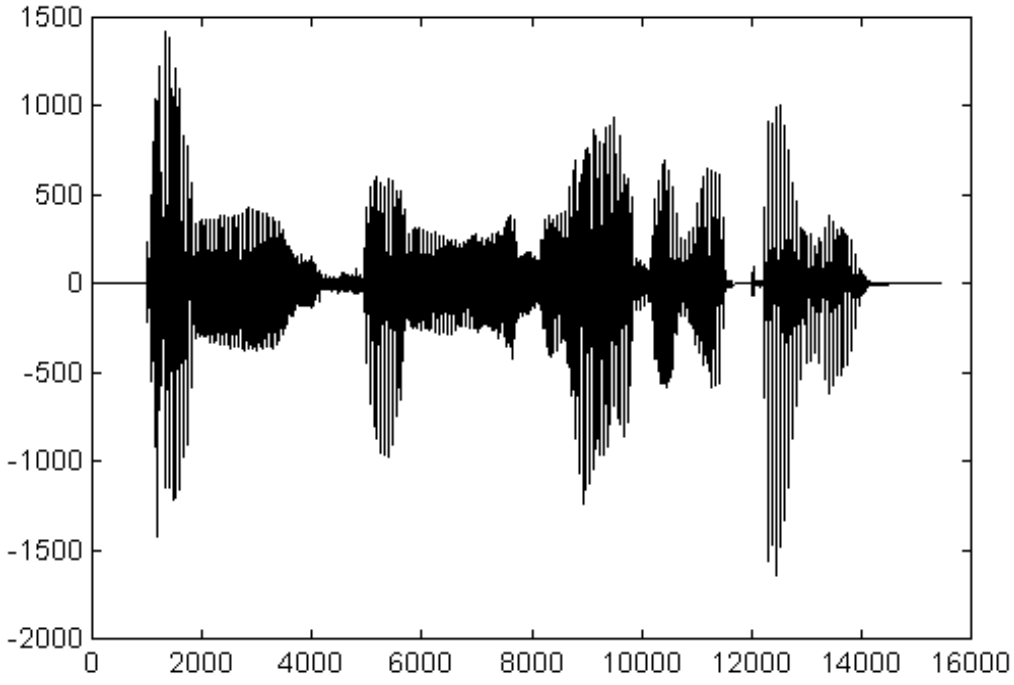
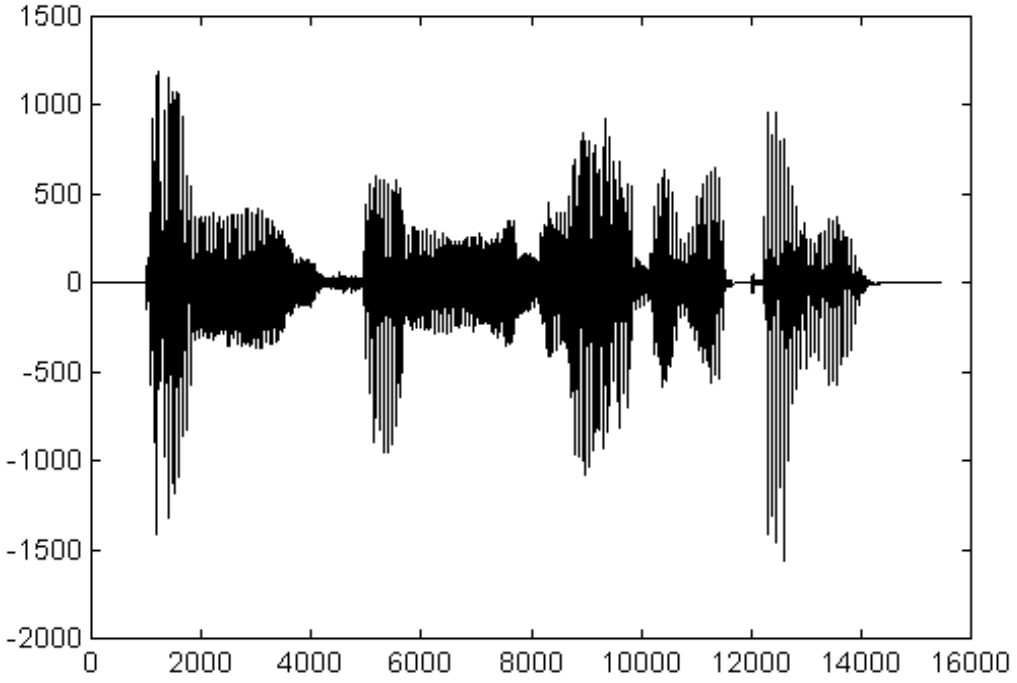


Fig 5.10 Comparaison de forme d'onde d'un segment de parole de la phrase M1. (a)- signal original. (b)- signal synthétique. (LPC=20).



-a-



-b-

Fig 5.11 Exemple de phrase codée M1 **a)** signal original **b)** signal synthétique. (LPC=20).

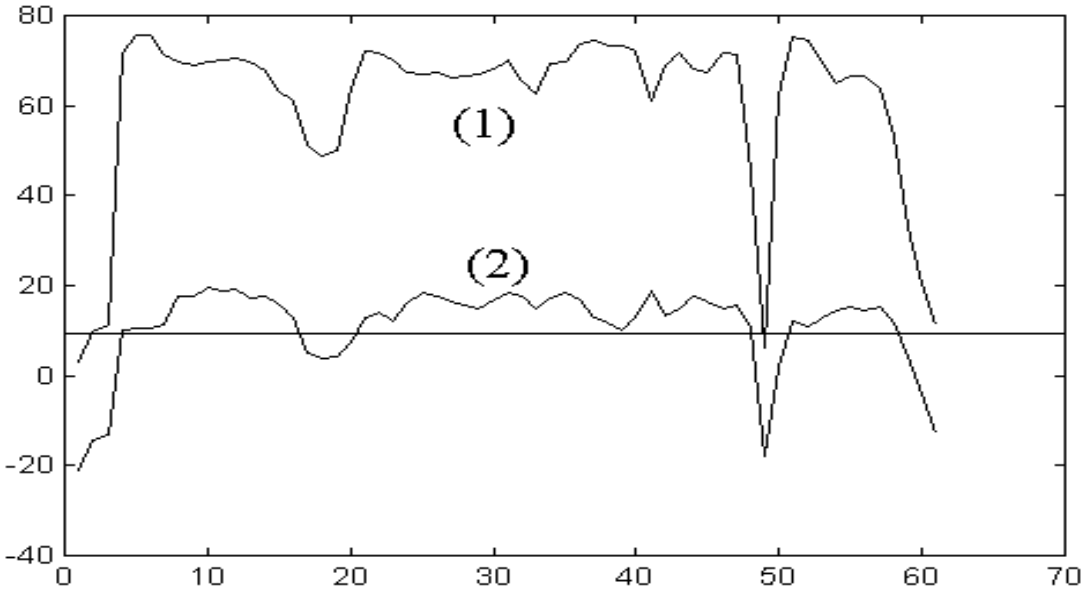


Fig 5.12 *(*1) Puissance du signal de parole (2) Evolution du RSB (trames de 240 échantillons) (LPC=20).

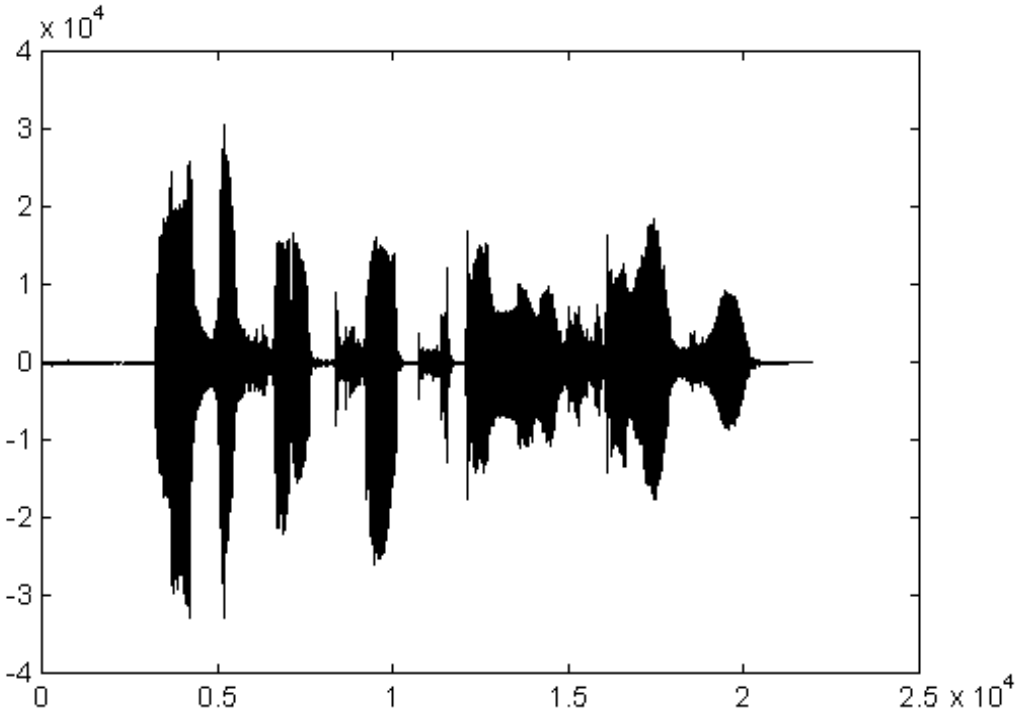
5.3.3 Le LD-CELP avec dictionnaire stochastique et LPC=50.

Le dictionnaire de 7 bits du paragraphe précédent (trames de 10 échantillons) qui a donné de bonnes performances dans le cas des vecteurs de dimension 10, a été utilisé pour la conception d'un dictionnaire *stochastique* de taille 14 bits (2^{14} ou 16384 vecteurs) : nous effectuons le produit du dictionnaire de 7 bits avec lui-même. Par exemple le codage des deux phrases M1 et F1 avec ce dictionnaire stochastique a donné comme résultats :

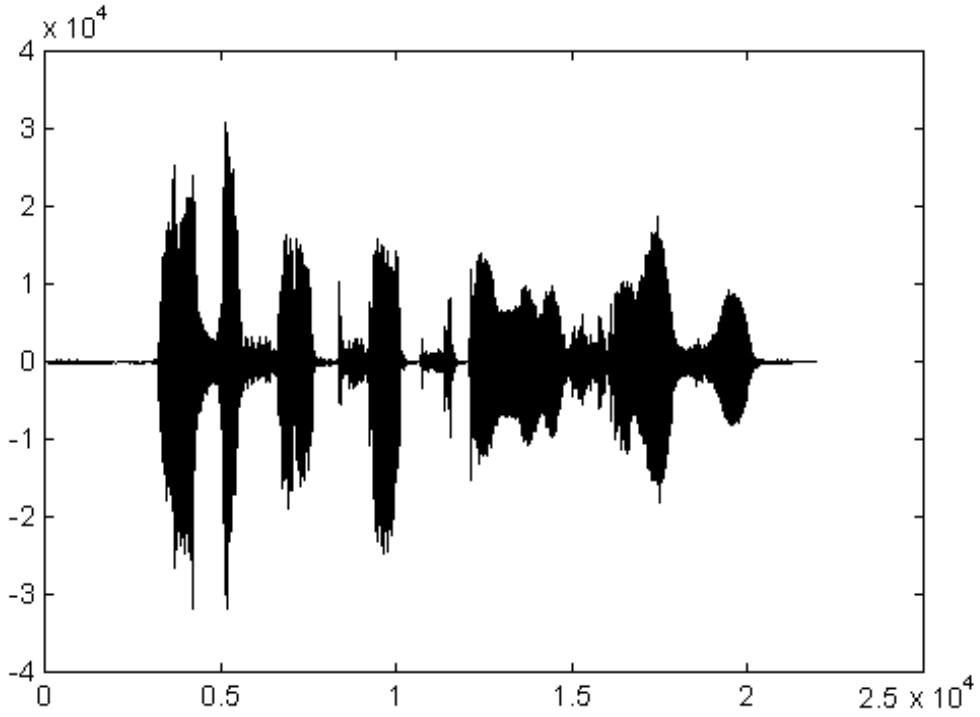
Phrases	SNR(dB)	SNRSEG(dB)
M1	**13.33**	**11.29**
F1	**14.04**	**10.17**

Tableau 5.5 Performances objectives du LD-CELP 8kb/s avec des trames de 20 échantillons et un dictionnaire stochastique de 14 bits (7 bits $\times$ 7 bits).

La figure 5.13. nous donne la représentation du signal original F1 ainsi que son synthétique. La figure 5.14. nous donne une comparaison d'un segment de forme d'onde de parole tiré de la même phrase F1.

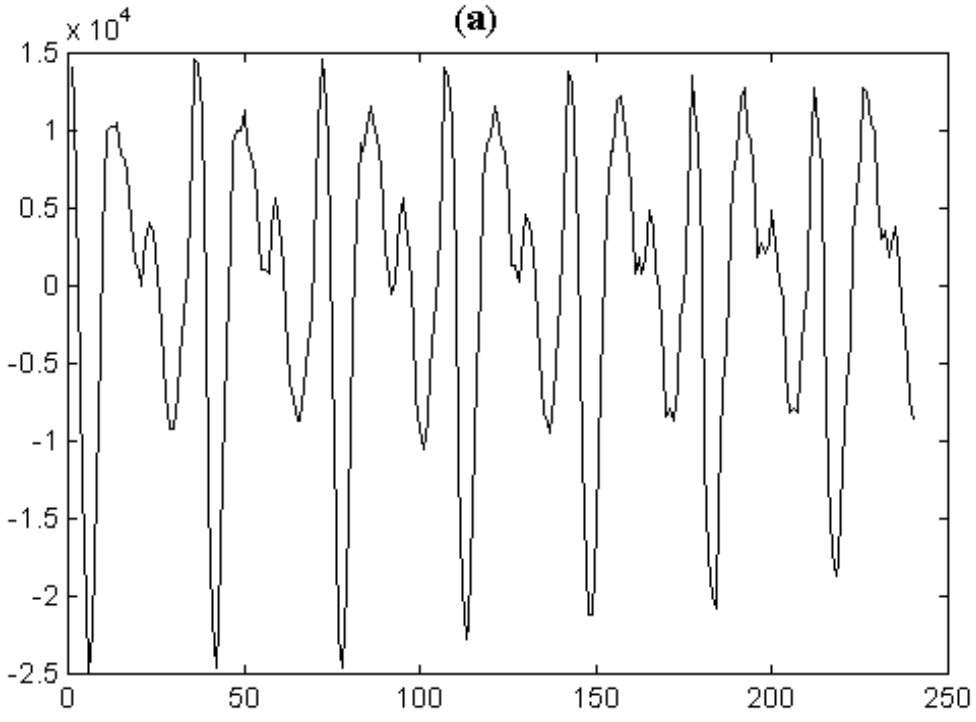


-a-



-b-

Fig. 5.13 Codage de F1. (a) signal original. (b) signal synthétique.



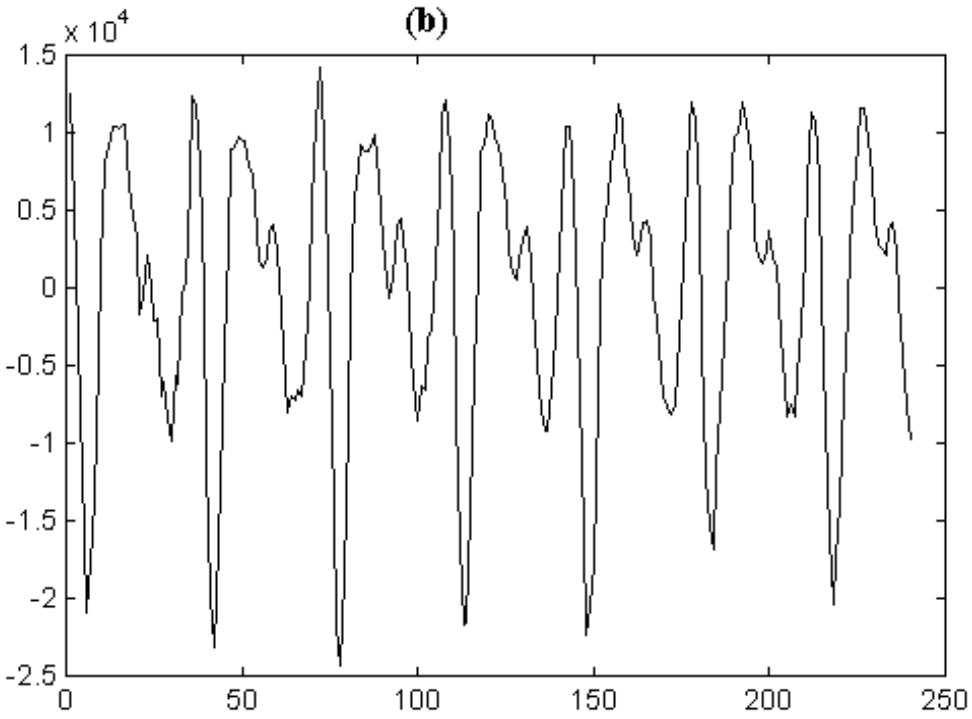


Fig. 5.14 Comparaison de forme d'onde d'un segment de parole de la phrase F1. (a) signal original. (b) signal synthétique.

Comparativement au dictionnaire algébrique de 14 bits, le dictionnaire stochastique est sensiblement moins performant (le SNR segmental est en moyenne inférieur de 2.5 dB). Ceci était prévisible car il aurait fallu chercher le dictionnaire stochastique optimisé de 14 bits et non pas faire le produit de 2 dictionnaires de 7 bits. Ceci n'a pas été fait vu la complexité (taille du dictionnaire grande).

5.3.4 Le LD-CELP avec dictionnaire algébrique et pitch backward.

Donc comme nous l'avons souligné au paragraphe précédent, une bonne qualité de parole à 8 kb/s ne peut être obtenue que si l'on exploite de manière *explicite* les redondances à long terme se trouvant dans le signal de parole [47]. Nous avons ainsi inséré dans la structure de notre codeur un prédicteur *pitch* adaptatif en backward comme cela est montré à la figure ci-contre :

Backward Pitch Predictor

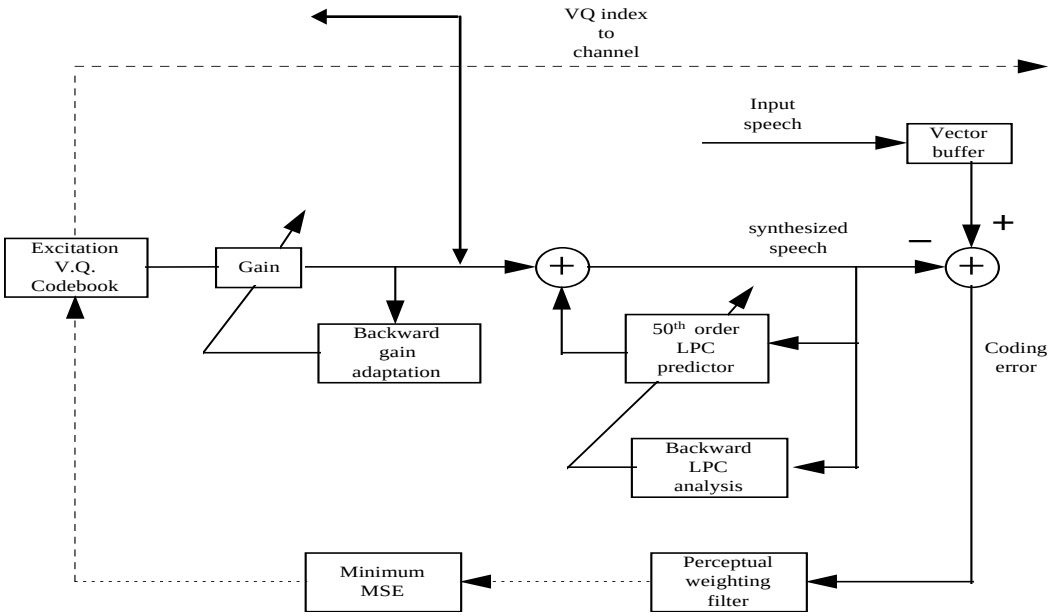


Fig 5.15 structure du codeur avec prédicteur pitch adaptatif backward.

Ce prédicteur est en fait comme mentionné dans les paragraphes (5.2.1) et (5.2.2) un dictionnaire adaptatif dont le contenu varie. La valeur du pitch et du gain du fait de l'adaptation en backward *n'auront pas à être transmis vers le récepteur* et sont données par les équations (5.14) et (5.15). Comme nous l'avons cité précédemment, dans notre système adaptatif backward, le signal de parole original n'est pas disponible et c'est le signal de parole reconstruit qui est utilisé pour déterminer les paramètres du pitch.

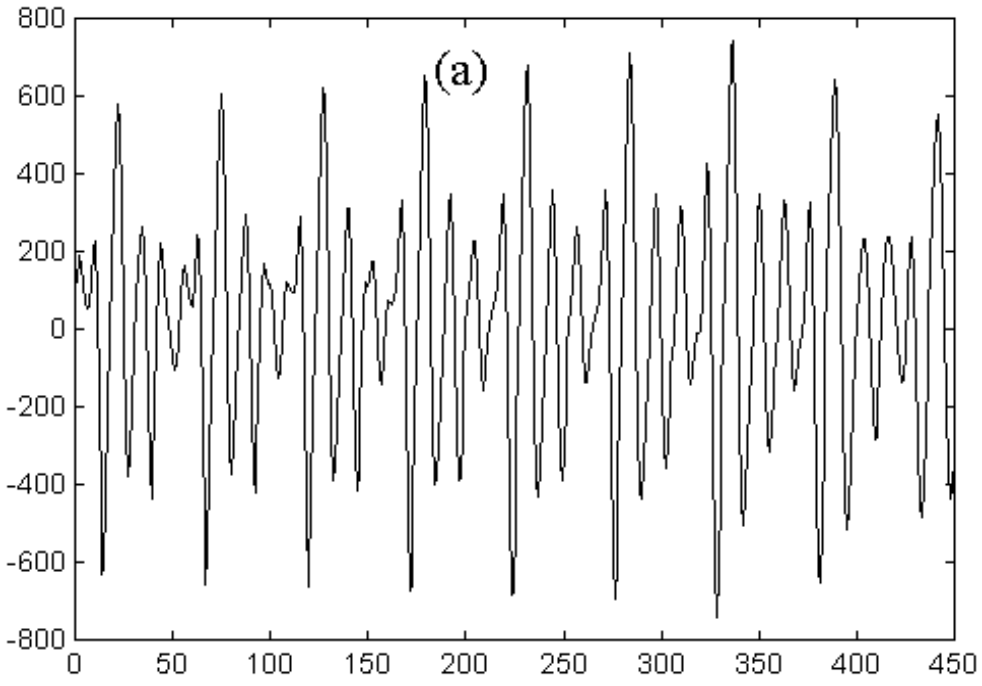
Le SNR et SNR segmental de ce codeur sont donnés dans le tableau ci-contre :

Phrases	**SNR(dB)**	**SNRSEG(dB)**
M1	14.76	13.91
M2	14.83	12.25
M3	14.63	12.52
M4	14.77	13.91
F1	15.65	12.92
F2	17.32	14.44
F3	16.56	14.68
F4	16.55	13.82

Tableau 5.6 Performances objectives du LD-CELP 8kb/s avec des trames de 20 échantillons, un dictionnaire algébrique de 14 bits, LPC=20 et prédicteur pitch backward.

A partir donc des deux tableaux (5.4) et (5.6) nous voyons que le prédicteur pitch backward améliore les performances du LD-CELP 8 kb/s de 0.12 dB en moyenne seulement et ceci est du à la réaction du bruit de quantification (Noise Feedback) et à la disparité du temps (Time mismatch) dans l'estimation des paramètres du pitch.

La figure 5.16. nous donne une comparaison d'un segment de forme d'onde de parole tiré de la phrase F4.
La figure 5.17. nous donne un exemple de codage de la phrase F4 et son signal synthétique obtenu. La figure 5.18 nous donne l'évolution du rapport signal à bruit en fonction du temps de la phrase F4 en prenant des trames de 240 échantillons.



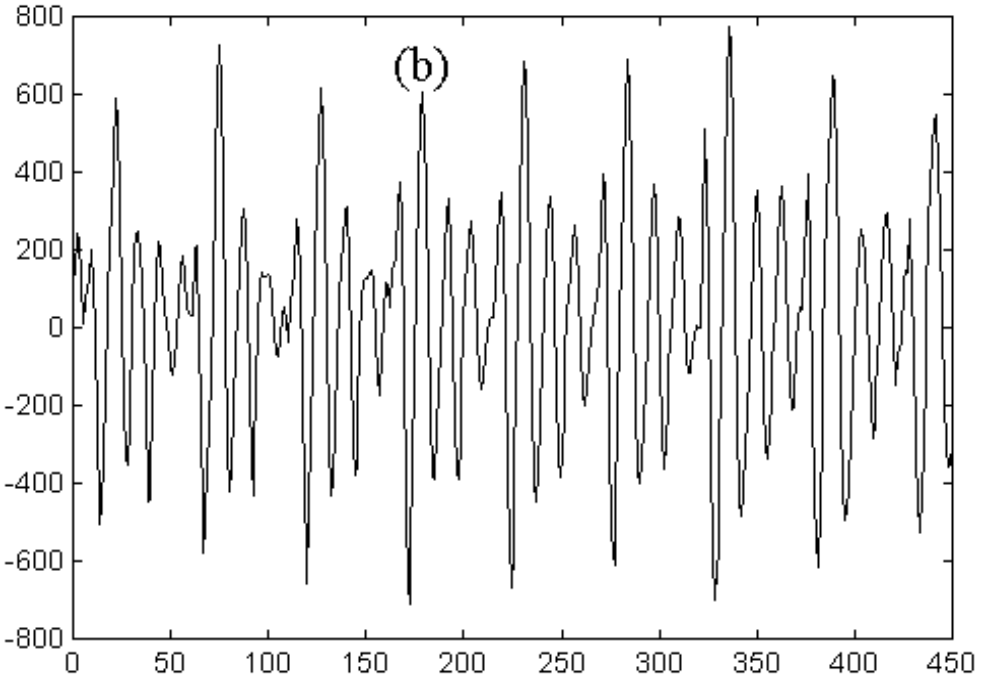
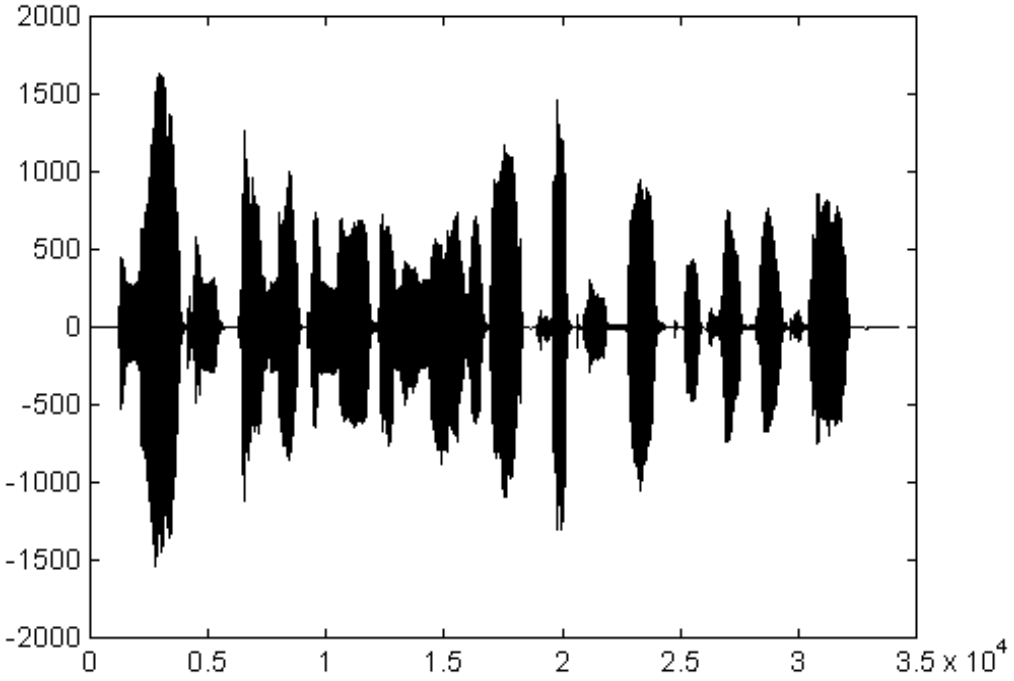
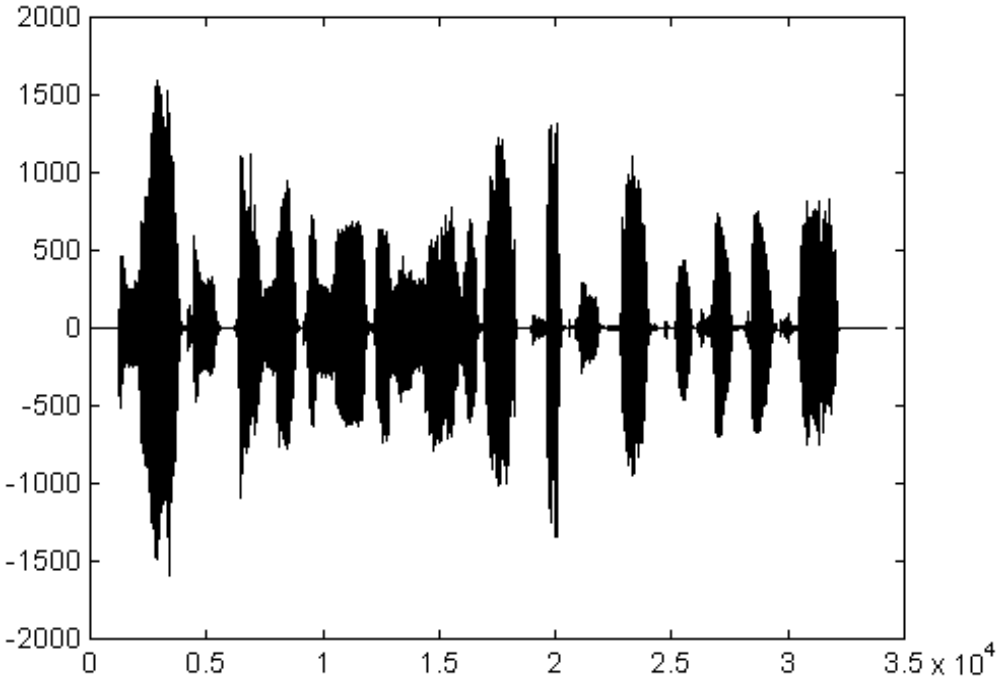


Fig 5.16 Comparaison de forme d'onde d'un segment de parole de la phrase F4. (a)- signal original. (b)- signal synthétique.



-a-



-b-

Fig 5.17 Codage de la phrase F4 **a)** signal original **b)** signal synthétique.

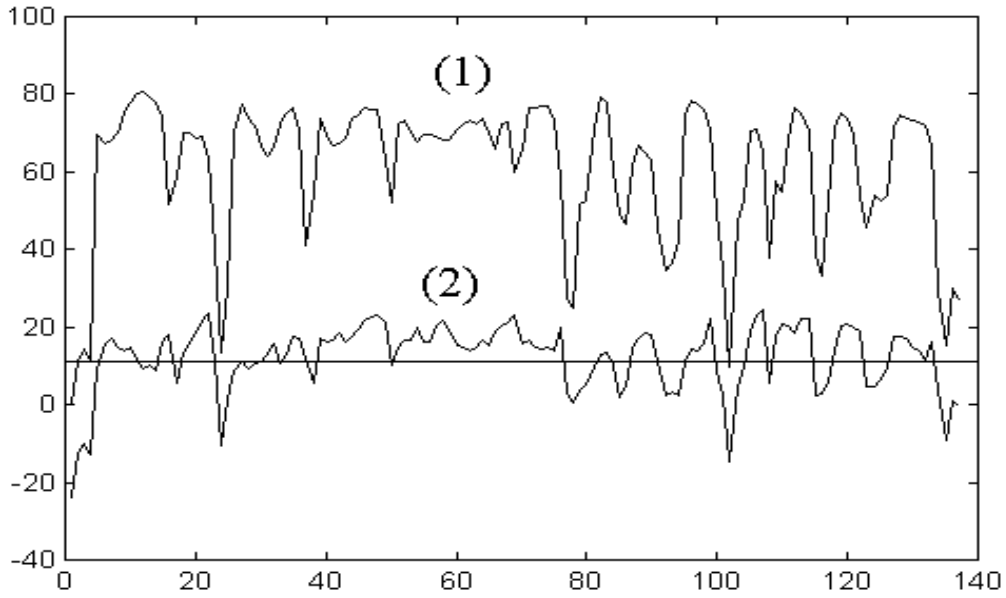


Fig 5.18 *(*1) Puissance du signal de parole (2) Evolution du RSB (trames de 240 échantillons)

L'adaptation en backward a l'avantage de réduire de manière importante le délai ou retard du codeur et ôter l'information sur les coefficients des filtres qui doit être transmise. Cette information externe est habituellement égale à 25% du débit du codeur et donc il est utile qu'elle soit enlevée [42]. Cependant l'adaptation en backward a l'inconvénient de produire des coefficient de filtre qui sont relativement dégradés en comparaison avec ceux des codecs travaillant en adaptation forward. Cette dégradation comme nous l'avons souligné plus haut est due à deux facteurs [48]:

1- **Réaction du bruit (Noise Feedback)** : Dans un système adaptatif en backward, les coefficients du filtre sont obtenus à partir du signal quantifié et donc il y a retour (réaction) du bruit de quantification dans l'analyse LPC. Ceci va dégrader la performance des coefficients produits.

2- **Disparité du temps (Time Mismatch)** : Dans un système adaptatif en forward, les coefficients du filtre pour la trame courante sont dérivés du signal de parole d'entrée pour la trame courante. Par contre dans une structure backward, nous avons seulement le signal disponible à partir de la trame précédente. Nous aurons donc une disparité du temps entre la trame courante et les coefficients utilisés pour cette trame.

L'effet de *réaction du bruit de quantification* augmente considérablement lorsque l'on travaille à des débits réduits.

5.3.5 Le LD-CELP avec dictionnaire algébrique et prédicteur pitch forward.

Pour résoudre le problème de la disparité du temps rencontré dans l'adaptation du pitch en backward, nous avons utilisé une structure *forward* pour ce dernier. Ayant 20 bits pour coder des sous-trames de 20 échantillons (débit à 8 kb/s), nous sommes donc contraint d'allouer 10 bits pour les paramètres du filtre long-terme du moment que ces derniers seront transmis au récepteur. Nous aurons ainsi dix (10) bits pour la quantification de l'excitation du dictionnaire fixe $Gc_j(n)$ (7 bits pour le vecteur *forme* $c_j(n)$ et 3 bits pour le *gain* G correspondant) et dix (10) bits pour les paramètres de l'excitation pitch $\beta r(n-M)$ c'est-à-dire pour le dictionnaire adaptatif (7 bits pour le *délai* M, et 3 bits pour le *gain* β). Donc le dictionnaire forme passe de 14 bits à 7 bits soit une réduction de moitié.

Un dictionnaire stochastique de 7 bits a été utilisé pour la quantification de l'excitation

$Gc_j(n)$. Il contient 128 vecteurs optimisés en boucle fermée.

Le tableau 5.7 donne les performances objectives de ce codeur :

Phrases	**SNR(dB)**	**SNRSEG(dB)**
M1	14.67	13.56
M2	14.18	11.93
M3	14.09	12.13
M4	14.31	13.38
F1	15.52	12.21
F2	17.37	14.64
F3	16.78	15.40
F4	16.77	14.53

Tableau 5.7 Performances objectives du LD-CELP 8kb/s avec des trames de 20 échantillons, un dictionnaire stochastique de 7bits, LPC=20 et prédicteur pitch forward.

Nous constatons que les performances du codeur sont sensiblement les mêmes que celles du codeur avec adaptation du pitch en backward (paragraphe précédent) dans lequel le dictionnaire fixe était de 14 bits. Ceci étant logique dans la mesure où la réduction de la taille du dictionnaire fixe de 14 bits à 7 bits a été compensée par l'adjonction d'un prédicteur pitch adaptatif **forward** c'est-à-dire que les paramètres du pitch, le **gain** β et le **délai** M ne sont plus extraits à partir du signal synthétique reconstruit mais à partir du signal original. Ainsi donc, le problème de la disparité du temps (time mismatch) ne se pose plus. On remarque aussi que le temps de codage est réduit et ceci étant du au fait que la taille du dictionnaire fixe est de 7 bits seulement au lieu de 14 bits.

La figure 5.19 nous donne une comparaison d'un segment de forme d'onde de parole tiré de la phrase F4.

La figure 5.20 nous donne un exemple de codage de la même phrase F4 avec son signal synthétique. La figure 5.21 montre l'évolution du RSB obtenu en fonction du temps par rapport à la puissance du signal parole.

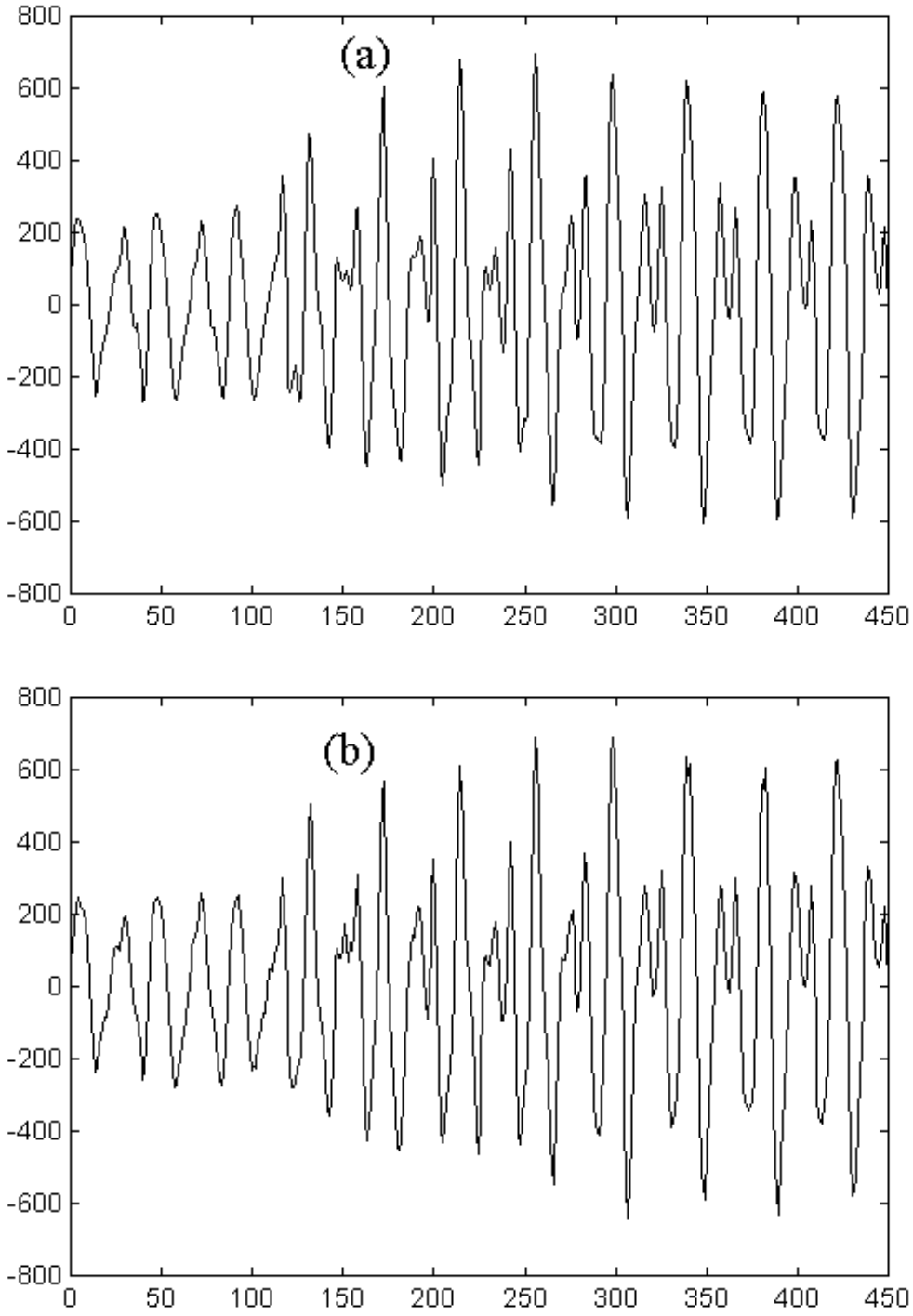
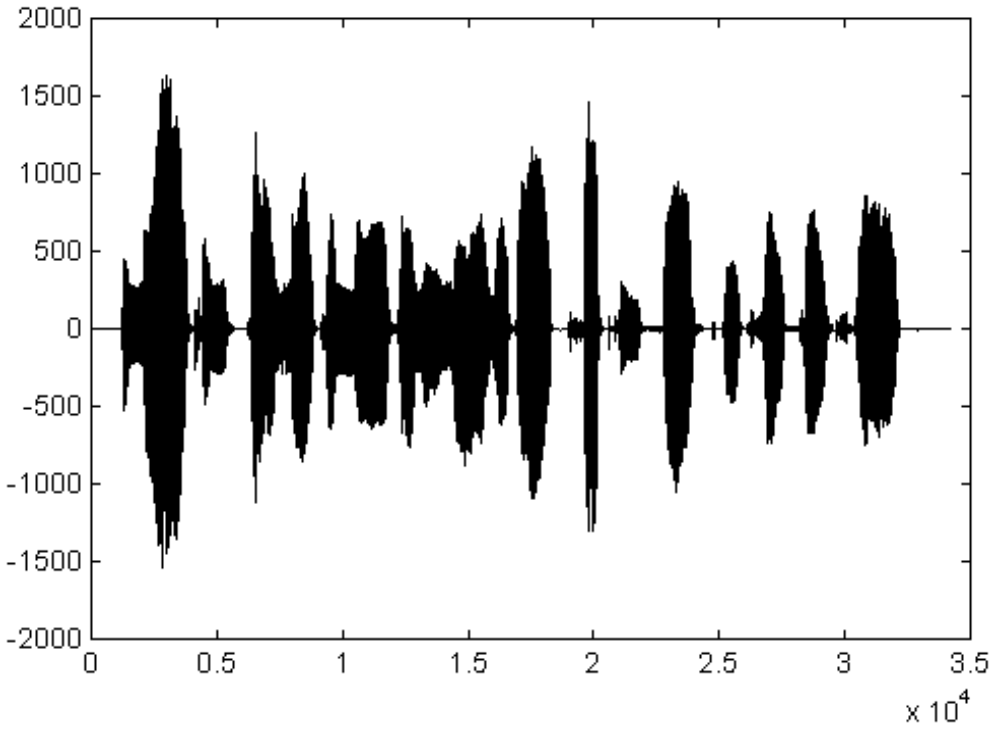
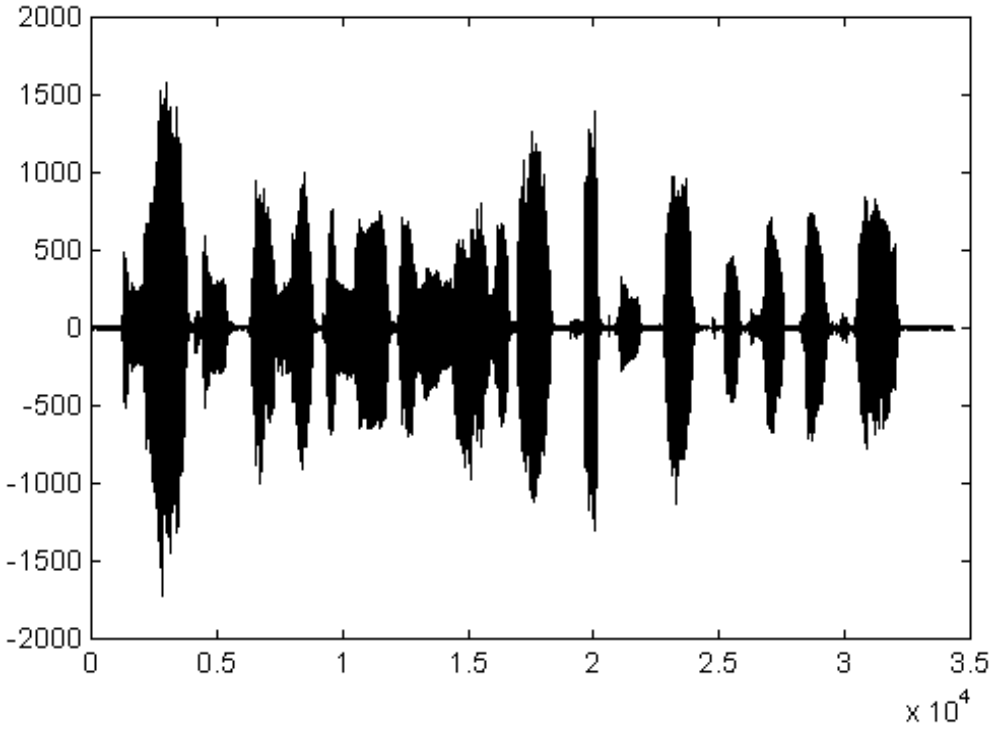


Fig 5.19 Comparaison de forme d'onde d'un segment de parole de la phrase F4.
a) signal original. **b)** signal synthétique.



-a-



-b-

Fig 5.20 Codage de la phrase F4 **a)** signal original **b)** signal synthétique.

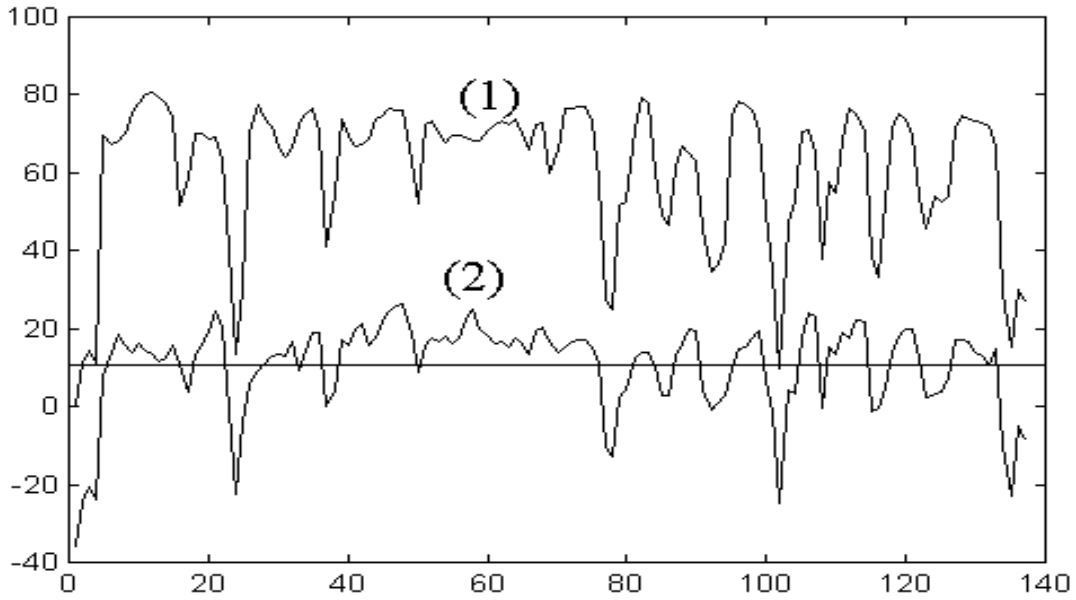


Fig 5.21 (1) Puissance du signal de parole. (2) Evolution du RSB. (en dB) (trames de 240 échantillons).

5.4 Conclusion

Après avoir donné une étude détaillée du LD-CELP 16 kb/s conçu et réalisé au CDTA, nous avons entamé dans ce chapitre l'étude du LD-CELP 8 kb/s. La première étape a consisté à doubler la taille du vecteur d'excitation de 5 à 10 échantillons en gardant la même structure algorithmique. Le codeur LD-CELP 8 kb/s obtenu a fourni un signal parole synthétique bruité et cela était prévisible du moment qu 'à 8 kb/s on est à 1 bit/dimension au lieu de 2 bits/dimension comme c'est le cas pour le 16 kb/s.. Nous avons trouvé dans le signal résiduel de prédiction du filtre de synthèse des corrélations long-terme et donc pour améliorer les performances de notre codeur 8 kb/s nous avons rajouté un prédicteur long-terme (LTP) adaptatif. Cela a été effectué en trois étapes :

1- Augmenter la taille de la sous trame de 10 à 20 échantillons.
2- Diminuer l'ordre du filtre LPC de 50 à 20.
3- Insérer un prédicteur pitch adaptatif.

Le prédicteur pitch utilisé est un dictionnaire adaptatif qui remplace le filtre prédicteur du pitch à un (01) état. Deux types d'adaptation du pitch ont été effectués. La première l'a été en backward et la seconde en forward. Les deux types d'adaptation ont donné sensiblement les mêmes performances mais l'adaptation du pitch en forward est préférée à celle effectuée en backward en ce sens que la taille du dictionnaire fixe est réduite de moitié et le temps de traitement (ou de codage) est réduit aussi.

Conclusion Générale

Un Codeur/Décodeur de parole LD-CELP 8 kb/s à bande étroite a été conçu et réalisé donnant ainsi une bonne qualité de parole synthétique. Ce codeur utilise un prédicteur pitch adaptatif en backward.

Le problème du codage à moyen débit consiste à calculer, pour des segments de parole, les coefficients d'un filtre de synthèse puis à rechercher le signal d'excitation qui minimise l'énergie de la différence entre le signal à coder et le signal de synthèse.

L'introduction dans l'expression de l'erreur de codage d'une pondération tenant compte du processus de perception du bruit de quantification améliore considérablement la qualité de la parole codée.

La complexité de recherche du meilleur code vecteur a été accéléré grâce à la transformation des opérations de filtrage en des produits matriciels (soustraction du '' zéro input response''). On n'omettra pas de signaler la mise en œuvre du filtre de perception pour le masquage du bruit. Ce filtre améliore considérablement l'effet subjectif.

Afin d'accélérer davantage le temps de calcul mis pour la recherche du meilleur vecteur d'excitation, nous avons implémenté la technique dite '' Backward Filtering '' qui permet de réduire la complexité de recherche du vecteur d'excitation d'un facteur d'ordre p (ordre du modèle Autoregressif) dans le cas des dictionnaires pauvres en échantillons (Sparse codebook).

La recherche des paramètres du pitch $-\beta$ (gain) et D (délai) – s'est faite en boucle fermée. Il est connu que la recherche en boucle fermée offre de meilleure performance que celle en boucle ouverte. Pour l'adaptation en forward, Le codage du délai M a nécessité 7bits pour sa quantification alors que le gain β en a nécessité 3.

Pour la conception du dictionnaire d'excitation, deux types d'excitations ont été utilisés et combinés :

- Excitation issue d'un dictionnaire stochastique.
- Excitation issue d'un dictionnaire algébrique.

Les mesures objectives et subjectives montrent que la qualité de la parole synthétisée est de bonne qualité.

Un codeur LD-CELP travaillant à 8 kb/s et à bande étroite a été conçu et réalisé. Il peut être utilisé comme programme de compression de signaux de parole dans les supports de stockage tels les CD ROMs ou les supports de stockage miniaturisés.

Références bibliographiques

[1] N. Moreau, " Techniques de compression des signaux'', Collection Technique et Scientifique des Télécommunications, 1995.

[2] J. G. Proakis. Digital communications. McGraw-Hill, 1989, 2e edition.

[3] R. Boite et M. Kunt, "Traitement de la parole'', Presses Polytechniques Romandes, 1987.

[4] Calliope, "La parole et son traitement automatique'', Collection Technique et Scientifique des télécommunications, MASSON 1989.

[5] J. Stachurski, "A Pitch Pulse Evolution Model for Linear Predictive Coding of Speech", PhD's Thesis. McGill University Montreal, Canada. May 1997.

[6] M. Mauc, ''Réduction de la complexité des algorithmes de codage de la parole de type CELP'', Thèse de Doctorat en sciences, Université Paris XII Novembre 1993.

[7] N. Jayant and P. Noll, "Digital Coding of Waveforms : Principles and applications to speech and video", Englewood Cliffs, New Jersey : Prentice-Hall, 1984.

[8] M. R. Zad-Issa, ''Smoothing the Evolution of the Spectral Parameters in Speech Coders", Masters thesis. McGill University Montreal, Canada. January 1998.

[9] N. Jayant, "Signal Compression : Technology Targets and Research Directions", IEEE Journal on Selected Areas in Communications. Vol. 10, No. 5, June 92.

[10] J. H. Chen, R. V. Cox, Y. C. Lin, N. Jayant and M. J. Melchner, "A Low-Delay CELP Coder for the CCITT 16 kb/s Speech Coding Standard", IEEE Journal on selected areas in communications, Vol. 10, No.5, June 1992.

[11] N. Batri, ''Robust Spectral Parameter Coding in Speech Processing", Masters thesis. McGill University Montreal, Canada. May 1998.

[12] J. MacQeen, "Some Methods for Classification and Analysis of multivariate observations", In Proc.Of Fifth Berkeley Symposium on Math. Stat. And Prob., Vol. 1,pp.281-296,1967.

[13] G. S. Kang and D. C. Coulter, "600 bps voice digitizer", ICASSP, pp 91-94, Philadelphia 1976.

[14] M. B. Pucci, ''Compression numérique de la parole au Moyen de codeurs de type CELP (Code Excited Linear Prédiction)'', Thèse de Doctorat, Université de Nice, Juin 1992.

[15] R. M. Gray, ''Vector Quantization", IEEE Trans. Acoustics, Speech, Signal Processing, Vol.ASSP-34, April 1984.

[16] J. Makhoul, S. Roucos and H. Gish, "Vector Quantization in Speech Coding", Proc. IEEE, Vol. 73, pp. 1551-1588, November 1985.

[17] V. Recordel, C. Labit, “Quantification Vectorielle par emboîtage d’une hiérarchie de réseaux réguliers de points’’, IRISA, Publication Interne N° 945. July 1995.

[18] A. Gersho, R. M. Gray, “Vector Quantization and Signal Compression”, Kluwer Academic Publishers, Boston, 1992.

[19] Y. Linde, A. Buzo, and R. M. Gray, “An algorithm for vector quantizer design”, IEEE Transactions on Communications,1980.

[20] I. Katsavounidis, C.-C. Jay Kuo, and Zhen Zhang, “A New Initialisation Technique for Generalized Liyod Iteration”, IEEE Signal Processing Letters. Vol. 1 No. 10 october 1994.

[21] J.Makhoul, “Linear prediction: A tutorial review”, Proc. IEE, vol.63, pp.561-580, April 1975.

[22] J.R.Deller Jr.,J.G.Proakis,and J.H.L.Hansen, ‘’Discrete-Time Processing of Speech Signal”, Macmillan, 1993.

[23] B.S.Atal and S.L.Hanauer, “Speech analysis and synthesis by linear prediction of the speech wave”, J. Acoustical Society of America, vol.50, pp.637-655, Aug.1971.

[24] G. Golub and C. Loan, “Matrix Computations”, Johns Hopkins University Press, Baltimore,1996.

[25] J. Leroux and C. Gueguen, “A fixed point computation of partial correlation coefficients”, IEEE Trans. Acoust., Speech, Signal Processing, Vol.ASSP-25, no. 3, pp. 257-259, 1979.

[26] S. Haykin, “Adaptative Filter Theory”, Upper Saddle River: Prentice Hall, third ed., 1996.

[27] C.Papacostantinou, “Improved pitch Modelling for Low Bit-Rate Speech Coders”, McGill University, Montreal, august 1997.

[28] R.V.Cox and P.Kroon, “Low Bit-Rate Speech Coders for Multimedia Communication”, IEEE Communications Magazine. December 1996.

[29] B.S.Atal and J.R.Remde, “A New Model of LPC Excitation for Producting Natural Souding Speech at Low Bit Rates”, Proc.IEEE ICASSP, Apr.1982, pp.614-617.

[30] M.R.Schroeder and B.S.Atal, “Code-Excited Linear Prediction (CELP) : High Quality Speech at Very Low Bit Rates”, Proc.IEEE ICASSP, 1985, pp.937-940.

[31] J. P. Adoul *et al* ., ‘’Fast CELP Based on Algebraic Codes’’, Proc.IEEE ICASSP, Apr.1987, pp. 1957-1960.

[32] M. R. Schroeder and B. S. Atal, ”Stochastic Coding of Speech Signals at Very Low Bit Rate: The Importance of Speech Perception”, Speech Communications Vol. 4, Aug 1985, pp. 155-162.

[33] M. R. Schroeder and B. S. Atal, ”High Quality Speech at Very Low Bit Rates”, in. Proc. Intern. Conf. On Acoust. Speech and Signal Process., ICASSP 85, pp. 937-940.

[34] B. S. Atal, ”Predictive Coding of Speech at Low Bit Rates”, IEEE Trans. Comm. Vol. COM-30, pp. 600-614, Apr. 1982.

[35] L. Watts and V. Cuperman, “A Vector ADPCM Analysis by Synthesis Configuration for 16 kb/s Speech Coder”, in Proc. IEEE Global Comm. Conf. Dec. 1988, pp. 275-279.

[36] S. Singhal and B. S. Atal, “Improving Performance of Multi-Pulse LPC Coders at Low Bit Rates”, in Proc. Intern. Conf. Acoust. Speech, Signal Process., vol. 1, paper No. 1.3, March 1988.

[37] R. P. Ramachandran and P. Kabal, ”Pitch Prediction Filters in Speech Coding”, Vol. ASSP-37 No. 5, May 1989, pp. 642-650.

[38] R. P. Ramachandran and P. Kabal, ”Stability and Performance Analysis of Pitch Filters in Speech Coders”, IEEE Trans. Acoustics, Speech, Signal Process., Vol. ASSP-35, July 1987, pp. 937-946.

[39] James H. Y. Loo, ”Intraframe and Interframe Coding of Speech Spectral Parameters”, Master’s Thesis, McGill University, Montreal, Canada, September 1996.

[40] J. H. Chen, “High-Quality 16 kb/s Speech Coding with A One-Way Delay Less than 2ms”, in Proc. Int. Conf. Acoust. Speech, Signal Process. Apr. 1990, pp. 453-456.

[41] M.Djedou, “Conception et réalisation d’un codeur-décodeur de la parole à bande étroite à 16kbits/s et possédant un faible retard, ‘’ Thèse de Magister, Ecole Nationale Polytechnique, El Harrach Alger 1998.

[42] L. Hanzo and J. P. Woodard, “Voice Compression and Communications: Principles and Applications for Fixed and Wireless Channels,” Voice-Book-Sample, Department of Electronics and Computer Science, University of Southampton, UK 1999.

[43] T. P. Barnwell III, “Recursive Windowing for Generating Autocorrelation Coefficients for LPC Analysis”, IEEE Trans. Acoust. Speech and Signal Process, pp. 1062-1066, Oct 81.

[44] J. H. Chen and A. Gersho, “Gain-Adaptative Vector Quantization with Application to Speech Coding”, IEEE Trans. Comm, Sep. 1987, pp. 918-930.

[45] “Detailed Description of AT & T’s LD-CELP Algorithm” Technical Report.

[46] W. B. Kleijn, D. J. Krasinski, and R. H. Ketchum, “Improved speech quality and efficient vector quantization in SELP”, Proc. IEEE Int. Conf. On Acoustics, Speech, Signal Processing (New York), pp. 155-158, Apr. 1988.

[47] K. Amir, M. Bengherbi and M. Halimi, “Low Delay 8KB/S Speech Coder Using A Bacckward Adaptative Long Term Predictor”, Boumerdès, Algeria, November, 4^{th}- 6^{th}, 2000.

[48] C. Hong, ‘’Low Delay Switched Hybrid Vector Excited Linear Predictive Coding of Speech”, Phd Thesis, National University of Singapore 1994.