REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et
de la Recherche Scientifique

Ecole nationale Polytechnique

وزارة التعليـــــــم العالـــــــي
و البحث العلمـــــــي
المدرسة الوطنية المتعددة التقنيات

Département d'Electronique

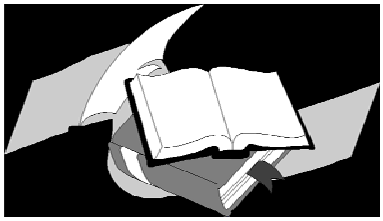
Projet de fin d'études

Thème :

Extraction des formes d'ondes caractéristiques dans dans le codeur de la parole par interpolation de formes d'ondes

Proposé et dirigé par :
M^elle^: F.MERAZKA

Étudié par:
- *LADJ Mohamed*
- *LARBI Mounir*



Promotion 2006

Laboratoire signal et communication
E.N.P. 10, Avenue Hassen-Badi, El Harrach, ALGER

REMERCIEMENTS

Ce travail a été effectué au sein du laboratoire de signal et communication du département d'électronique de l'Ecole Nationale Polytechnique, sous la direction de Dr F.MERAZKA

Nous tenons à lui exprimer nos plus sincères remerciements pour ses précieux conseils, son aide et sa patience tout au long de ce travail.

Nous exprimons notre plus sincère gratitude au Professeur D.BERKANI, pour son aide et sa disponibilité et qui a rendu possible l'entreprise de ce travail.

Nous tenons à remercier tous nos amis et camarades pour toute leur sincère amitié le long de cinq années d'études.

Dédicaces

Je dédie ce modeste travail
Aux deux êtres les plus chers au monde qui sont :

Ma très chère mère qui a toujours été la pour moi et qui s'est donnée de la peine afin que j'obtienne mon diplôme.

Mon père qui a été toujours derrière mes exploits, avec son aide matérielle et morale durant toutes mes études.

A mon unique frère Rafik que j'estime beaucoup.

A mes adorables sœurs Isama et Amel que j'apprécie beaucoup.

A mon beau frère Mounir et ma petite nièce Ines

A tous ceux qui me sont chers et à toute la famille.

A tous mes amis (es) qui m'ont aidé et
soutenu : Hassen, Rym, Nassima, Nadib,Farid, Réda, Rochdi, Mehdi, Nordine, Mouloud, Bilal, Djamil…

A toi mon ami et frère Mohamed pour les moments précieux que nous avons passé ensemble et que je n'oublierai jamais.

…Mounir

Dédicaces

Je dédie ce modeste travail
Aux deux êtres les plus chers au monde qui sont :

Mon père qui a été toujours derrière mes exploits, avec son aide matérielle et morale durant toutes mes études.

Ma très chère mère qui a toujours été la pour moi et qui s'est donnée de la peine afin que j'obtienne mon diplôme.

A mes frères que j'estime beaucoup.

A mes adorables sœurs.

A tous ceux qui me sont chers

A tous mes amis (es) qui m'ont beaucoup aidé et soutenu : Mehdi, Hassen, Nadjib, Farid, Hakim, Reda, Rochdi, Bilal, Djamil, Nordine, Mouloud …

A toi mon ami et frère Mounir pour les moments Précieux que nous avons passé ensemble et que je n'oublierai jamais.

…Mohamed

Résumé

Le codage de la parole à bas débit promet d'être largement employé dans des applications telles que la téléphonie visuelle et les communications mobiles et personnelles. Pendant les dernières décennies, on a proposé une variété de techniques de codage de la parole, analysée, et développée. Le présent travail a pour but de développer une partie du codeur basé sur le schéma d'interpolation de la forme d'onde (WI : Waveform Interpolation), avec comme objectif l'extraction des formes d'ondes caractéristiques
(CW : Caractéristique Waveform), pour cela l'implémentation a été réalisé a l'aide du langage de programmation C (Builder C++ 6.0) et la logiciel Matlab pour les représentations.

Mots clés : codage de la parole, prédiction linéaire, forme d'onde caractéristique, interpolation de la forme d'onde.

Abstract

Speech coding at low bit rates is expected to be widely deployed in applications such as visual telephony, mobile and personal communications. . During the last decades, one proposed a variety of techniques of speech coding, analyzed, and developed. This research focuses on developing a part of speech coder based on the waveform interpolation (WI) scheme, with like objective the extraction of characteristic waveform (CW), for that the implementation was carried out using the C programming language (Builder C++ 6.0) and the Matlab software for the representations.

Key words: speech coding, linear prediction, characteristic waveform, waveform interpolation.

الملخص

تشفير الكلام بتدفق منخفض يعد باستعمال واسع في مجالات مثل الهاتف المرئي و الاتصالات المتنقلة و الشخصية. في العشرية الأخيرة, تم اقتراح عدد مختلف من الطرق لتشفير الكلام.,هذا العمل يهدف إلى تحقيق جزء من نظام تشفير الكلام, يعتمد علي مبدأ استكمال شكل الموجة مع التركيز علي استخراج شكل الموجة المميزة. لهذا تم تجريبه باستخدام لغة البرمجةC و MATLAB لتمثيل المنحنيات.

كلمات مفتاحية : تشفير الكلام التنبؤ الخطي تشكيل الموجة المميزة استكمال شكل الموجة.

Sommaire

LISTE DES FIGURES

LISTE DES TABLES

LISTE DES ABREVIATIONS

ADPCM : Adaptive Differential Pulse Code Modulation

CDMA : Code Division Multiple Access

CELP : Code-Excited Linear Prediction

CODEC: Coder and Decoder

CW : Characteristic Waveform

DCVQ: Dimension Conversion Vector Quantization

DOD : Department of Defense (U.S.)

DSP : Digital Signal Processor

DTFS : Discrete-Time Fourier Series

EVRC: Enhanced Variable Rate Codec

FBR : Fixed Bit-Rate

FS : Federal Standard (U.S.)

GLA : Generalized Lloyd Algorithm

IMBE : Improved Multi-Band Excitation

ITU : International Telecommunication Union

ITU-T : ITU - Telecommunication standardization sector

LD-CELP: Low-Delay Code Excited Linear Prediction

LP : Linear Prediction

LPC : Linear Prediction Coding

LSF : Line Spectral Frequency

LSP : Line Spectral Pair

MBE : Multi-Band Excitation

MELP : Mixed Excitation Linear Prediction

MIPS : Million Instructions Per Second

MOS : Mean Opinion Score

MSE : Mean Square Error

PCM : Pulse Code Modulation

PWI : Prototype Waveform Interpolation

REW : Rapidly Evolving Waveform

SEW : Slowly Evolving Waveform

SNR : Signal-to-Noise Ratio

V/UV : Voiced /UnVoiced

VBR : Variable Bit-Rate

WI : Waveform Interpolation

Introduction

Dans les systèmes digitaux modernes, un son articulé est représenté dans un format digital, c'est-à-dire dans un ordre de bits binaires. Pour des applications de mémoire et de transmission, moins de bits signifie moins de mémoire et moins de largeur de bande, de puissance et/ou de mémoire respectivement. Le codage de la parole est la technologie qui offre de tels algorithmes de compactage des signaux de voix.

Avec la tendance croissante des transmissions et des applications multimédia tel que le répondeur digital, la demande sur l'économie de mémoire augmente. En même temps, il y a toujours un besoin croissant d'économie de largeur de bande, en particulier dans les communications par satellite et sans fil. Ces conditions duelles continueront certainement à coder la parole dans une zone animée de recherches et de développement à l'avenir.

L'objectif dans le codage de la parole est de représenter le signal vocal avec un nombre restreint de bits tout en mettant à jour sa qualité perceptuelle.

Plusieurs codeurs ont été développés offrant une qualité acceptable de la parole reproduite [18]. On peut citer les codeurs de forme d'ondes, les codeurs par synthèse et les codeurs hybrides.

Dans notre implémentation, nous nous somme intéressés au codage de la parole par interpolation de formes d'ondes, qui est une technique efficace de compactage et produit la parole de qualité aux bas débits binaires.

Cette thèse présente le codeur de la parole WI, dont notre objectif est d'extraire les formes d'ondes caractéristiques, pour améliorer la qualité du discours reproduit.

Nous avons donc organisé notre travail en trois chapitres :

Le premier chapitre est consacré au codage de la parole : la prédiction linéaire, le modèle de production de la parole humaine et sa distorsion.

Le deuxième chapitre donne une description générale des codeurs de la parole qui existent et détail le codeur WI jusqu'à l'extraction des formes d'ondes caractéristiques.

Le troisième chapitre regroupe l'étude des méthodes de l'extraction des CW que nous avons implémentées, les simulations réalisées et l'interprétation des résultats obtenus. Enfin une conclusion générale.

Chapitre 1

Codage de la parole

Introduction

Le traitement de la parole est aujourd'hui une composante fondamentale des sciences de l'ingénieur. Située au croisement du traitement du signal numérique et du traitement du langage (c'est-à-dire du traitement de données symboliques), cette discipline scientifique a connu depuis les années 60 une expansion fulgurante, liée au développement des moyens et des techniques de télécommunications.
Ce chapitre regroupe des généralités sur les notions fondamentales de la production du signal parole, ses propriétés ainsi que sa perception. Cet aspect est utile à la bonne compréhension de l'évolution des techniques de codage de la parole.

1.1 Signal vocal

La parole peut être décrite comme étant le résultat de l'action volontaire et coordonnée d'un certain nombre d'organes. Cette action se déroule sous le contrôle du système nerveux central qui reçoit en permanence des informations par rétroaction auditive et par les sensations kinesthésiques [4].

1.2 Mécanisme de phonation

Les principaux organes composant l'appareil phonatoire sont [1]: les poumons, la trachée artère, le pharynx, les cavités buccales et nasales qui sont schématisés par la Figure 1.1.
L'appareil respiratoire fournit l'énergie nécessaire à la production de sons, en poussant de l'air à travers 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. Le larynx est un ensemble de muscles et de cartilages mobiles qui entoure une cavité située à la partie supérieure de la trachée.

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 progressivement, déterminer une ouverture triangulaire appelée ***glotte***. L'air y passe librement pendant la respiration et la voix chuchotée ainsi que pendant la phonation des sons non voisés.

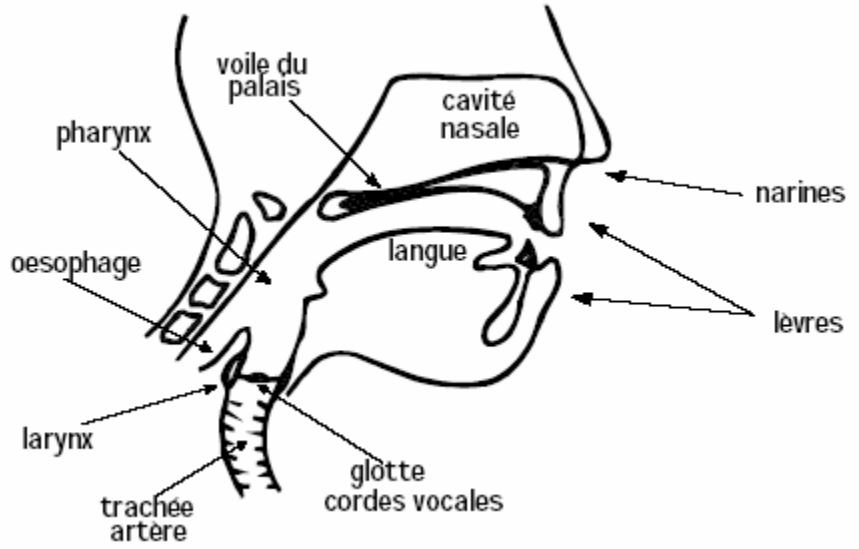


Fig. 1.1 Appareil phonatoire

Les sons voisés résultent, au contraire, d'une vibration périodique des cordes vocales. Le larynx est d'abord complètement fermé, ce qui accroît la pression en amont des cordes vocales et force ces dernières à s'ouvrir, ce qui fait tomber la pression en permettant aux cordes vocales de se refermer. Des impulsions périodiques de pression sont ainsi appliquées au conduit vocal composés des cavités pharyngienne et buccale pour la plupart des sons. Lorsque la *luette* est en position basse, la cavité nasale vient s'y ajouter en dérivation. Notons pour terminer le rôle prépondérant de la langue dans le processus phonatoire. Sa hauteur détermine la hauteur du pharynx : plus la langue est basse, plus le pharynx est court. Elle détermine aussi le *lieu d'articulation*, région de rétrécissement maximal du canal buccal, ainsi que l'aperture qui représente l'écartement des organes au point d'articulation. L'intensité du son émis est liée à la pression de l'air en amont du larynx. Sa hauteur est fixée par la fréquence de vibration des cordes

vocales, appelée fréquence du fondamental ou pitch. La fréquence du fondamental peut varier [2] [3] :

- 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.

Un *son voisé* est un signal quasi périodique dont le spectre est tracé à la Figure 1.2. On y observe les raies qui correspondent aux harmoniques du fondamentale F0 (pitch).
L'enveloppe de ces raies présente des maximums appelés *formants* et qui correspondent aux fréquences propres *Fi* du conduit vocal (structure formantique). Les trois premiers formants sont essentiels pour caractériser le spectre vocal; les formants d'ordre supérieur ont une influence plus limitée.

Un son *non voisé* ne présente pas de structure périodique. Il peut être considéré comme un bruit blanc filtré par la transmittance de la partie du conduit vocal situé entre la constriction et les lèvres comme le montre la Figure 1.3; son spectre ne présente donc pas de structure de pitch.
La classification ainsi exposée est forcément un peu sommaire et concerne surtout la production normale de la parole. Ainsi, une voyelle peut être chuchotée, c'est-à-dire produite avec la glotte largement ouverte; dans ce cas, le spectre du signal résulte de l'excitation du conduit vocal par une source aléatoire : c'est un spectre continu qui présente une structure formantique semblable à celle d'une voyelle voisée mais ne possède pas de structure de pitch (raies dues aux harmoniques du fondamental).

De nos jours, il reste très difficile de dire comment l'information auditive est traitée par le cerveau. On a pu, par contre, étudier comment elle était finalement perçue dans le cadre d'une science spécifique appelée *psychoacoustique*. Sans vouloir entrer dans trop de détails sur la contribution majeure des *psychoacousticiens* dans l'étude de la parole, il est intéressant d'en connaître les résultats les plus marquants. Ainsi, l'oreille ne répond pas également à toutes les fréquences. Le seuil d'audition de l'oreille est non linéaire par rapport aux fréquences. L'oreille atteint sa sensibilité maximale entre 3 et 4 kHz.

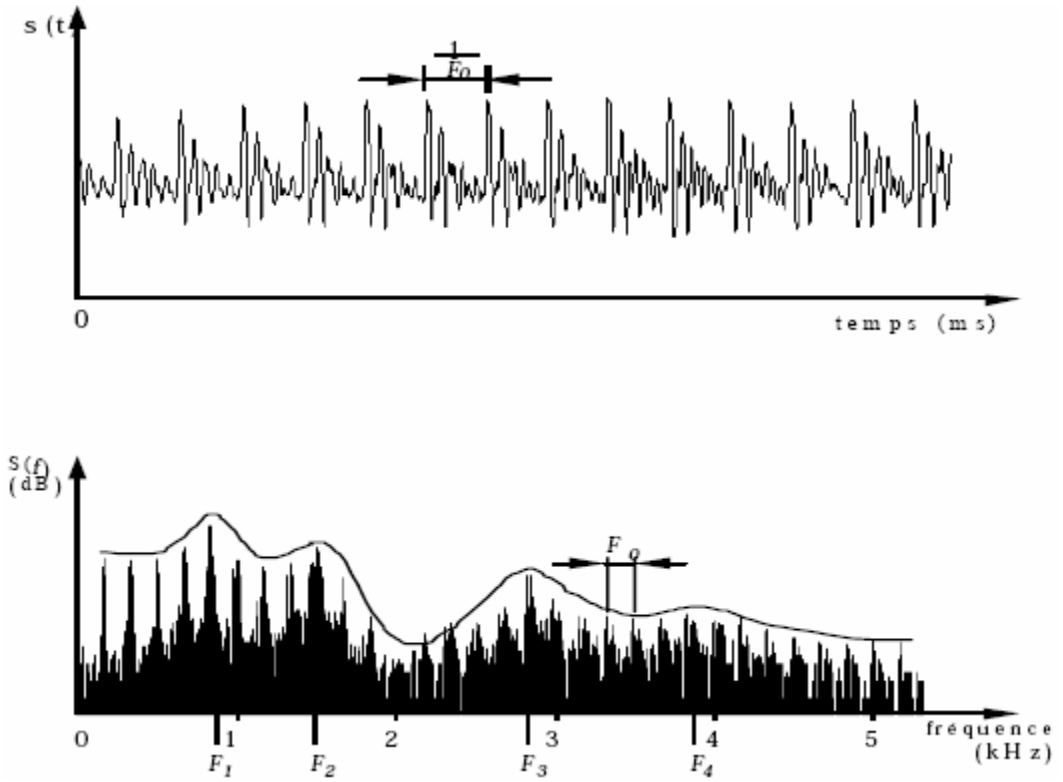


Fig. 1.2 Un signal vocal voisé et son spectre [3][4]

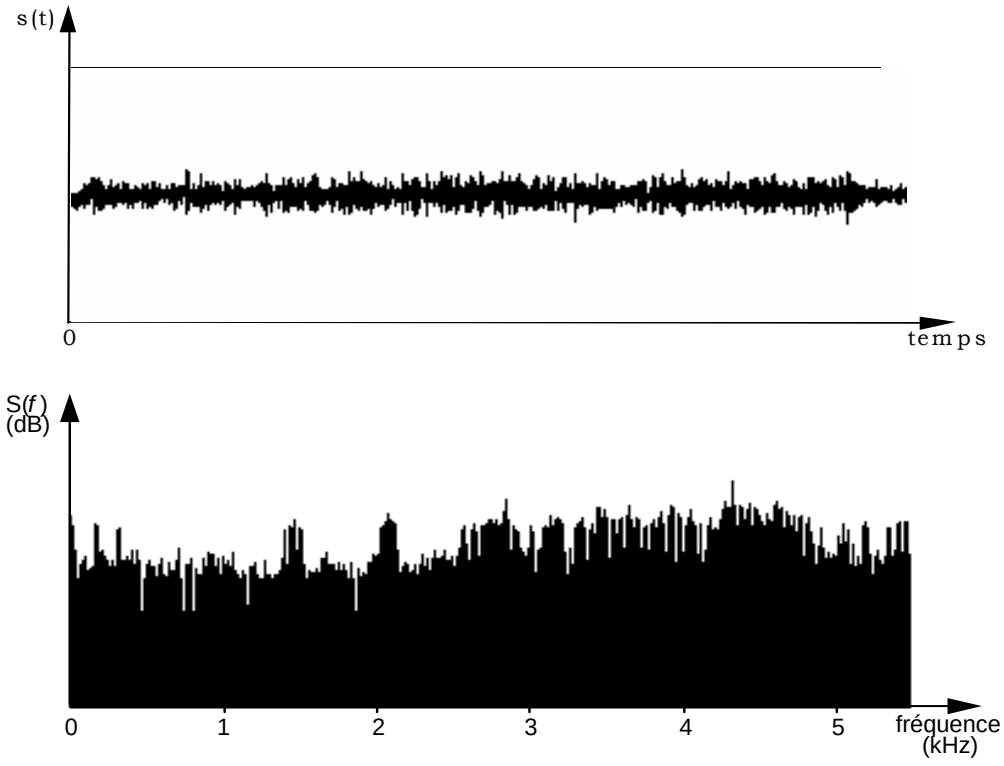


Fig. 1.3 Un signal vocal non voisé et son spectre [3][4]

1.3 La redondance du signal vocal

Telle que définie par Shannon, la redondance est la partie du signal parole qui, si elle est éliminée, n'affecte pas le contenu du message ou du signal information.
Le signal vocal est caractérisé par une très grande redondance, condition nécessaire pour résister aux perturbations du milieu ambiant, cette redondance sera mise à profit par les techniques de codage de la parole, pour réduire le débit binaire nécessaire au stockage ou à la transmission de la parole, sans, pour autant nuire à son intelligibilité.

On définie l'information associée à un message constitué par des éléments discrets x_i appartenant à un ensemble donné X, et si $p(x_i)$ est la probabilité a priori d'occurrence du symbole x_i , on a donc l'information moyenne associée à l'occurrence du message $X=[x_1,x_2,\ldots,x_l]$ qui vaut :

$$H(X) = -\sum_i p(x_i) \log_2 p(x_i) \tag{1.1}$$

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

Dans la conversation courante, environ dix phonèmes [(1)] sont prononcés chaque seconde; l'information moyenne est donc inférieur à 50 bits/s [2].Or, on sait que pour un canal continu sans erreurs, le débit maximum d'information est donné par l'équation (1.2) :

$$C=B \log_2[1+S/N] \tag{1.2}$$

Avec B est la longueur de la bande passante en Hz, et *S/N* est le rapport signal sur bruit en dB.
Par exemple, pour un canal téléphonique, supposé continu et sans erreurs, de bande passante B=3000 Hz et avec un rapport signal sur bruit *S/N*=30 dB, on trouve C=30000 bits/s, il y a apparemment une redondance énorme dans ce canal. La suppression partielle des redondances permet une représentation plus efficace des données.

[(1)]Phonème : c'est la plus petite unité présente dans la parole et susceptible par sa présence de changer la signification d'un mot [2].

La compression des données peut se faire sans pertes d'information ou avec pertes en exploitant dans ce cas la tolérance de l'organe récepteur (l'oreille). La compression du signal consistera à réduire les redondances du signal parole.

1.4 Modèle de production de la parole

L'analyse de la parole est une étape indispensable à toute application de synthèse, de codage ou de reconnaissance.

Le modèle électrique linéaire a été proposé par Fant [3] en 1960, qui spécifie qu'un signal voisé peut être modélisé par le passage d'un train d'impulsions *u(n)* à travers un filtre numérique récursif de type tous-pôles (*Auto Régressif*). On montre que cette modélisation reste valable dans le cas des sons non voisés, à condition que *u(n)* soit cette fois un bruit blanc. Le modèle final est illustré à la Figure I.4. Il est souvent appelé modèle auto régressif (*AR*), parce qu'il correspond dans le domaine temporel à une régression linéaire de la forme :

$$s(n)=G.u(n)+\sum_{i=1}^{p}-a_i s(n-1) \tag{1.3}$$

Où *u(n)* est le signal d'excitation et *p* l'ordre du système.

Chaque échantillon est obtenu en ajoutant un terme d'excitation à une prédiction obtenue par combinaison linéaire des *p* échantillons précédents.

Les coefficients du filtre $\{a_i\}$ sont appelés coefficients de prédiction et le modèle AR est souvent appelé modèle de prédiction linéaire.

les paramètres du modèle *AR* sont : la période du train d'impulsions (sons voisés uniquement), la décision Voisé/Non Voisé (V/NV), le gain *G* et les coefficients du filtre *1/A(z)*, appelé ***filtre de synthèse***.

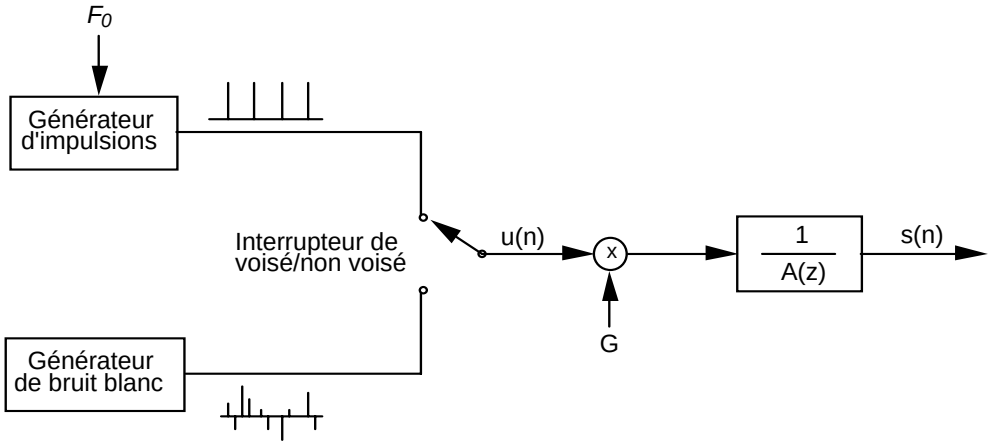


Fig. 1.4 Modèle simplifié de production de la parole [3][4]

Les relations d'équivalences entre le modèle physique et le modèle mathématique peuvent être données comme suit :

Conduit vocal	⟺	*H(z)*, le filtre LPC
Le flux d'air	⟺	Le signal d'excitation *u(n)*
Vibration des cordes vocales	⟺	Voisé
Fricatives et plosives	⟺	Non Voisé/Voisé
Volume d'air	⟺	*G,* le gain

Le problème de l'estimation d'un modèle AR, souvent appelée analyse *LPC* revient à déterminer les coefficients d'un filtre tous-pôles dont on connaît le signal de sortie, mais pas celui de l'entrée. Il est par conséquent nécessaire d'adopter un critère, afin de faire un choix parmi l'ensemble infini de solutions possibles. Le critère généralement utilisé est celui de la minimisation de l'énergie de l'erreur de prédiction.

1.5 Prédiction Linéaire

La prédiction linéaire est assez bien utilisée dans les systèmes de codage et de compression [6][7][8]. Cette méthode est considérée comme une technique prédominante pour l'estimation des paramètres de la parole. Son succès est dû au fait qu'elle représente une solution linéaire au problème de l'estimation des paramètres du modèle de la production de la parole.
Le principe fondamental de la prédiction linéaire est qu'un échantillon donné peut être prédit à partir d'une combinaison linéaire des échantillons finis qui le précèdent. Un seul jeu de coefficients du prédicteur sont déterminés en minimisant les différences entre les échantillons actuels et ceux prédits. La technique de prédiction linéaire est basée sur le modèle de la production de la parole représenté à la figure 1.4.

Le signal parole *s(n)* peut être modélisé comme la sortie d'un système *auto régressif à moyenne ajustée* (ARMA) avec une entrée *u(n)* [3][5][9]. Son expression est alors :

$$s(n) = \sum_{k=1}^{p} a_k s(n-k) + G\sum_{i=0}^{q} b_i u(n-i), \quad b_0=1, \tag{1.4}$$

Où le gain *G*, les coefficients $\{a_k\}$ et $\{b_i\}$sont les paramètres du système, et p et q sont les ordres des polynômes. L'équation (1.4) prédit la sortie courante en utilisant une combinaison linéaire des sorties précédentes et les entrées courantes et précédentes.

Dans le domaine fréquentiel, la fonction de transfert du modèle de prédiction linéaire de la parole est de la forme :

$$H(z) = \frac{B(z)}{A(z)} = \frac{G[1 + \sum_{i=1}^{q} b_i z^{-i}]}{1 - \sum_{k=1}^{p} a_k z^{-k}} \tag{1.5}$$

H(z) est le modèle pôle-zéro dans lequel les racines du dénominateur et de numérateur sont, respectivement, les pôles et les zéros du système.

Si a_k=0 pour *1≤k≤p*, *H(z)* devient un modèle tous-zéros ou modèle à *moyenne ajustée* (*MA*).

Si pour b_i=0, pour 1≤i≤q, *H(z)* devient un modèle tous-pôles ou modèle *auto régressive* (*AR*), exprimé par :

$$H(z) = \frac{1}{A(z)} \tag{1.6}$$

L'analyse spectrale montre que les pôles correspondent aux résonances du conduit vocal, c'est-à-dire aux *pics* du spectre, les *formants* ; tandis que les zéros correspondent aux antirésonances, c'est-à-dire aux *vallées*.

Dans l'analyse de la parole, les classes de phonèmes comme les fricatives et les nasales contiennent des vallées spectrales qui correspondent aux zéros dans *H(z)*.
Par contre, les voyelles contiennent des résonances qui peuvent être modélisées par le modèle tous-pôles; pour des raisons de simplicité, ce modèle est préféré pour l'analyse par prédiction linéaire de la parole. Ainsi, le signal prédit est égal à :

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{1.7}$$

La différence entre l'échantillon original $s(n)$ et l'échantillon prédit $\tilde{s}(n)$ est appelée *erreur de prédiction* (ou *résidu*) et elle est définie par:

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{p} a_k s(n-k) \tag{1.8}$$

Le problème de l'analyse par prédiction linéaire se réduit donc à trouver un ensemble de coefficients a_k de façon à minimiser l'erreur de prédiction $e(n)$ dans un certain intervalle. Les méthodes d'estimation des coefficients a_k sont nombreuses [10].
Deux grandes approches sont utilisées pour l'analyse par prédiction linéaire LPC court-terme : La méthode d'autocorrélation et la méthode de covariance.

1.5.1 Méthode d'Autocorrélation

La méthode d'autcorrélation garantit la stabilité du filtre LP. Les hypothèses de cette méthode sont les suivantes :

Le signal est défini pour toutes les valeurs du temps ; il est identiquement nul en dehors d'une séquence de *N* échantillons, où *N* est un entier; ceci est équivaut à multiplier le signal de parole *s(n)* par une fenêtre *w(n)* de longueur finie correspondant à *N* échantillons pour obtenir un segment du signal de parole fenêtré $s_w(n)$ [11].

$$s_w(n) = \begin{cases} w(n).s(n) & pour \quad 0 \le n \le N-1 \\ 0 & ailleurs \end{cases} \tag{1.9}$$

La fonction de pondération la plus courante est la fenêtre de *Hamming* :

$$w(n) = \begin{cases} 0.54 - 0.46\cos\dfrac{2n\pi}{N-1} & pour \quad 0 \le n \le N-1 \\ 0 & ailleurs \end{cases} \tag{1.10}$$

Chaque échantillon peut être prédit approximativement à partir des échantillons précédents. Ceci est valable pour toutes les valeurs du temps; ($-\infty<n<+\infty$).

L'erreur quadratique totale entre le signal fenêtré $s_w(n)$ et le modèle (signal prédit) est minimisée sur l'ensemble des échantillons.

La fonction d'autocorrélation du signal fenêtré $s_w(n)$ est :

$$R(i) = \sum_{n=1}^{N-1} s_w(n).s_w(n-i) \qquad 1 \le i \le p \tag{1.11}$$

La fonction d'autocorrélation est une fonction paire: *R(i)= R(-i)*.

Pour trouver les coefficients du filtre LPC, l'énergie du résiduel de prédiction doit être minimisée sur l'intervalle fini : $0 \le n \le N-1$

$$E = \sum_{n=-\infty}^{\infty} e^2(n) = \sum_{n=-\infty}^{\infty} [s_w(n) - \sum_{k=1}^{p} a_k s_w(n-k)]^2 \tag{1.12}$$

Cette erreur peut être minimisée en annulant les dérivées partielles par rapport aux coefficients du filtre :

$$\frac{\partial E}{\partial a_k} = 0 \qquad 1 \le k \le p \tag{1.13}$$

On obtient p équation linéaire avec p coefficient inconnus a_k :

$$\sum_{k=1}^{p} a_k \sum_{n=-\infty}^{\infty} s_w(n-i)s_w(n-k) = \sum_{n=-\infty}^{\infty} s_w(n-i)s_w(n). \qquad tq: 1 \le i \le p \tag{1.14}$$

Alors, les équations linéaires peuvent être écrites sous la forme :

$$\sum_{k=1}^{p} R(|i-k|)a_k = R(i) \qquad 1 \le i \le p \tag{1.15}$$

La forme matricielle de l'ensemble des équations linéaires (1.14) est représenté par ***R.a=v*** et peut être réécrite comme suit :

$$\begin{bmatrix} R(0) & R(1) & \dots & R(p-1) \\ R(1) & R(0) & \dots & R(p-2) \\ & & \dots & . \\ . & . & \dots & . \\ . & . & \dots & . \\ R(p-1) & R(p-2) & \dots & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ . \\ . \\ . \\ R(p) \end{bmatrix} \tag{1.16}$$

La matrice d'autocorrélation *pxp* obtenue est symétrique dont tous les éléments de la diagonale sont égaux, c'est une matrice de *Toeplitz*. Ce qui nous permet de trouver les coefficients de prédiction minimisant la moyenne quadratique de l'erreur de prédiction par l'algorithme de *Levinson-Durbin* (Annex A).

1.5.2 Méthode de Covariance

Les méthodes d'autocorrélation et de covariance diffèrent dans l'emplacement de la fenêtre d'analyse.

Dans cette méthode c'est le signal erreur qui est fenêtré au lieu du signal parole, de façon à ce que l'énergie à minimiser soit :

$$E = \sum_{n=-\infty}^{\infty} {e_w}^2(n) = \sum_{n=-\infty}^{\infty} e^2(n) w^2(n) \tag{1.17}$$

En annulant les dérivées partielles en utilisant l'équation (I.13) on obtient p équations linéaires :

$$\sum_{k=1}^{p} \Phi(i,k) = \Phi(i,0) \qquad 1 \le i \le p \tag{1.18}$$

Où la fonction de covariance :

$$\Phi(i,k) = \sum_{n=-\infty}^{\infty} w(n) s(n-1) s(n-k) \tag{1.19}$$

On peut exprimer les p équations, sous la forme : $\Phi.a=\Psi$

$$\begin{bmatrix} \Phi(1,1) & \Phi(1,2) & \dots & \Phi(1,p) \\ \Phi(2,1) & \Phi(2,2) & \dots & \Phi(2,p) \\ \Phi(3,1) & \Phi(3,2) & \dots & \Phi(3,p) \\ & & \dots & \\ & & \dots & \\ \Phi(p,1) & \Phi(p,2) & \dots & \Phi(p,p) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \\ \\ a_p \end{bmatrix} = \begin{bmatrix} \Psi(1) \\ \Psi(2) \\ \Psi(3) \\ . \\ . \\ \Psi(p) \end{bmatrix} \tag{1.20}$$

Tel que; $\Psi(i)=\Phi(i,0)$ pour $1 \le i \le p$

La matrice Φ n'est pas une matrice Toeplitz, et ne garantie pas la stabilité du filtre LPC, elle est symétrique et définie positive. Donc, la matrice de covariance peut être décomposée en deux matrices, l'une triangulaire inférieure L et l'autre triangulaire supérieure U.

$$\Phi=L.U \tag{1.21}$$

La décomposition de Cholesky peut être utilisée pour convertir la matrice de covariance sous la forme :

$$\Phi=C.C^T \qquad \text{tq; } C=L \text{ et } C^T=U$$

Le vecteur a est obtenu en résolvant d'abord l'équation (I.22) :

$$L.y=\Psi \tag{1.22}$$

Puis :

$$U.a=y \tag{1.23}$$

1.5.3 Considération Pratiques

Pour bien mener l'analyse LPC, il faut choisir :

- La fréquence d'échantillonnage f_e.
- La méthode d'analyse et l'algorithme correspondant.
- L'ordre p de l'analyse LPC.
- Le nombre d'échantillons par tranche N et le décalage entre tranches successives L.

Le choix de la fréquence d'échantillonnage est fonction de l'application visée et de la qualité du signal à analyser :

- 8 kHz pour les signaux téléphoniques.
- 10 kHz pour les applications de reconnaissance.
- 16 kHz pour les applications de synthèse.

L'ordre de prédiction p est choisi de façon à ce qu'il permette de bien représenter toute la séquence du signal parole; l'ordre p est fonction de la fréquence d'échantillonnage, on estime en général qu'une paire de pôles est nécessaire par 1Khz de bande passante.

Lorsque la fréquence d'échantillonnage est f_e (exprimée en échantillons/sec), une période de 1ms correspond à $f_e/1000$ échantillons.

A la fréquence d'échantillonnage de 8 kHz, la valeur correspondante de p doit être au moins égale à 8. Elle trouve d'ailleurs une justification expérimentale dans le fait que l'énergie de l'erreur de prédiction diminue rapidement lorsqu'on augmente p à partir de 1, pour tendre vers une asymptote au voisinage de ces valeurs : il devient inutile d'augmenter encore l'ordre, puisqu'on ne prédit rien de plus.

De plus la durée des trames d'analyse et leur décalage sont souvent fixés inférieur à 30ms.Les valeurs choisies sont liées au caractère quasi-stationnaire du signal parole.

Enfin, comme vu précédemment dans la méthode d'autocorrélation, pour compenser les effets de bord, on multiplie en général préalablement chaque tranche d'analyse par une fenêtre de pondération *w(n)*, la plus souvent utilisées est celle de *Hamming* (équation (1.10)).

1.5.4 Représentation des paramètres de prédiction

Les coefficients de prédiction linéaire (*LP*) sont calculés à base de "bloc par bloc", généralement sur des trames de 5-40ms [12]. Pour une transmission efficace de la parole, les coefficients *LP* sont sujets à une **quantification** et une **interpolation**. L'interpolation rend possible la transmission de l'information sur les coefficients *LP* moins souvent, ainsi réduisant le débit binaire. Cependant, une simple quantification ou une interpolation des coefficients *LP* est problématique parce que de petits changements dans les coefficients peuvent induire un grand changement dans le spectre de puissance et causer l'instabilité du filtre de synthèse *LP*. Par conséquent, un nombre de représentations des coefficients *LP* été considéré pour essayer de trouver la représentation qui minimise ses limitations.

Les représentations les plus utilisées sont les coefficients de réflexion, les LAR (log-area ratios) [12] et les LSPs (Line Spectrum Pairs) [13].

Cependant la représentation la plus répandue et la plus prisée pour ses performances reste la représentation en paires de rais spectrales LSP.
Elles seront détaillées dans ce qui va suivre.

1.5.4.1 Paires de raies spectrales

Connus aussi sous le nom de fréquences de raies spectrales.

La représentation LSP a été introduite par ***Itakura*** [13].

Les LSPs sont les solutions des deux équations suivantes :

$$\begin{cases} P(z) = A(z) + z^{-(p+1)} A(z) \\ Q(z) = A(z) - z^{-(p+1)} A(z) \end{cases} \tag{1.24}$$

Ce qui nous donne :

$$A(z) = \frac{1}{2}[P(z) + Q(z)] \tag{1.25}$$

Soong et ***Juang*** [14] ont montrés que si *H(z)* est stable, où *A(z)* est à phase minimale, alors les zéros des polynômes *P(z)* et *Q(z)* sont appels les LSP. Ces polynômes ont les propriétés suivantes [4]:

- Tous les zéros de *P(z)* et *Q(z)* se trouvent sur le cercle unité.
- Les zéros de *P(z)* et *Q(z)* sont entrelacés les uns aux autres, les LSP sont dans un ordre croissant.

Il a été montré [15] que le filtre LPC *A(z)* est à phase minimale si et seulement si les LSP satisfont les deux propriétés citées plus haut, donc la stabilité du filtre de synthèse est facilement vérifiable. De plus, les caractéristiques suivantes ont été relevées

1. comme illustré dans la figure il y a une relation évidente entre les LSP et le spectre du filtre LPC. Une concentration des LSP dans une certaine bande de fréquences correspond approximativement à une résonance dans cette bande.
2. sensibilité spectrale; un changement d'une LSP cause seulement un changement dans la forme du filtre d'analyse dans une petite gamme de fréquence autour de cette LSP.

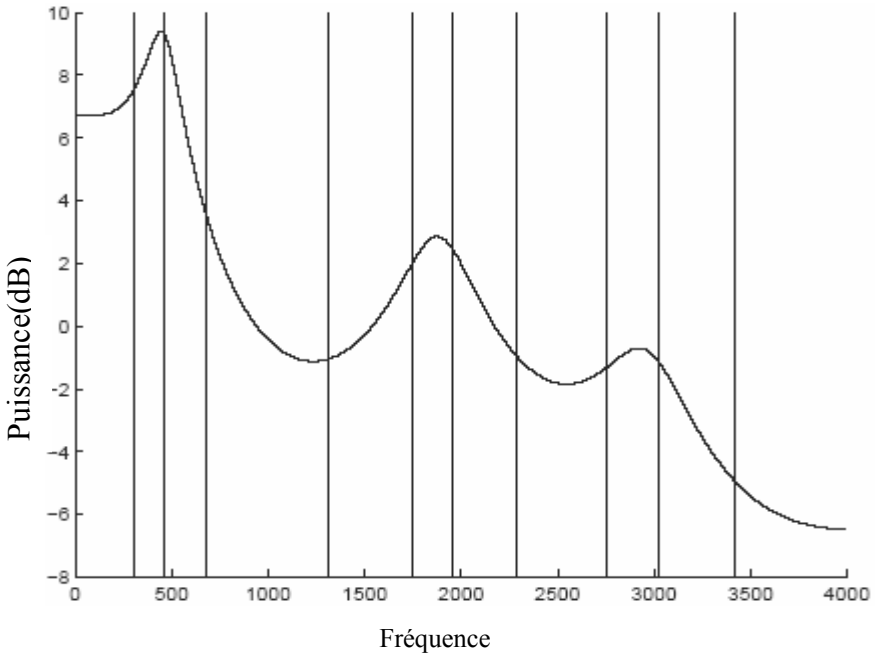


Fig. 1.5 Spectre LPC avec LSF superposé

1.6 Principe de la quantification

La quantification est le processus de substitution des échantillons d'un signal analogique par des valeurs arrondies prises parmi un nombre fini de valeurs possibles [4].

La quantification peut être *scalaire* ou *vectorielle* selon que les signaux sont à une ou plusieurs dimensions. La quantification vectorielle peut être de deux types soit statistique ou algébrique.

1.6.1 Quantification scalaire

Dans la quantification scalaire (*QS),* chaque échantillon du signal d'entrée est quantifié séparément des autres échantillons. Comme l'illustre la figure 1.6, un échantillon x du signal d'entrée est spécifié par l'indice k s'il se trouve dans l'intervalle suivant :

$$I_k : \{x_k \ \pi \ x \le x_{k+1}\} \qquad k = 1,2,....., N \tag{1.26}$$

Les valeurs x_k et x_{k+1} sont appelées niveaux de décision ou seuils.

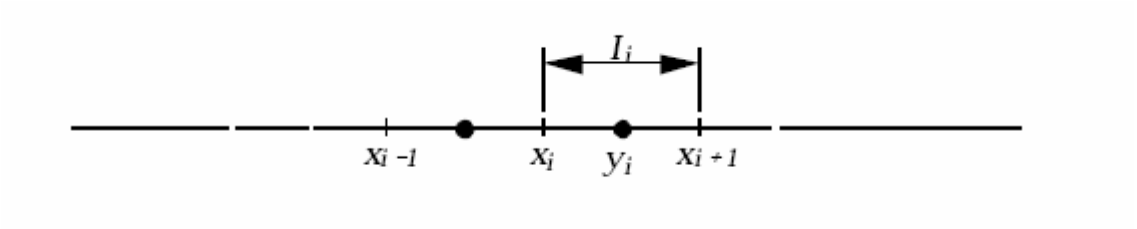


Fig. 1.6 Quantification scalaire

Tous les échantillons situés dans l'intervalle I_i seront remplacés par une valeur y_i appelée *niveau de reconstruction* ou *représentant*.

1.6.2 Quantification vectorielle

La quantification vectorielle (*VQ*) est l'extension de la quantification scalaire à un espace multidimensionnel.

Nous appellerons quantificateur vectoriel de dimension *m* à *N* niveaux une application *Q* qui, à un vecteur d'entrée $x = \{x_1, x_2, ..., x_m\}$, fait correspondre une valeur approchée *y* choisie dans un ensemble fini de *N* éléments $y = \{y_i, i = 0,1,....., N-1\}$.

L'ensemble *y* est un dictionnaire de *N* représentants. En posant $R = \log_2(N)$, nous dirons que les vecteurs d'entés sont quantifiés sur *N* niveaux et codés avec *R* bits.

Contrairement à la quantification scalaire, un quantificateur vectoriel peut fonctionner avec un débit fractionnaire (R < 1) [5].

1.7 Qualité des codeurs

L'estimation de la qualité d'un codeur est un problème complexe. Une première approche consiste à utiliser une mesure objective de la ressemblance qui existe entre le signal original et le signal reconstitué. Cette méthodologie se situe dans le domaine des tests dits "objectifs". Ils s'appliquent très bien aux codeurs de bonne qualité et font plutôt appel à la théorie du signal qu'aux connaissances sur la parole.

Lorsque l'on cherche une évaluation plus fine des codeurs, il faut faire appel à la dimension subjective de la qualité de la parole. Étant donné la part de subjectivité qui est présente dans l'appréciation d'un individu, il faut utiliser des procédures de test très élaborées. L'évaluation d'un

codeur à l'aide de tests subjectifs est une opération délicate qui est généralement confiée à des laboratoires spécialisés.

1.7.1 Mesure de distorsion subjective

L'évaluation subjective est obtenue par des tests d'écoutes; dans ces tests, la qualité de la parole est mesurée par l'intelligibilité spécifiquement définie par le pourcentage de mots ou phonèmes correctement écoutés et avec une sonorité naturelle (naturalness).

Il existe trois types de mesures subjectives [4] de la qualité généralement utilisées.

- Le test DRT (Diagnostic Rhyme Test)
- Le test DAM (Diagnostic Acceptability Measure)
- Le test MOS (Mean Opinion Score)

MOS	Qualité
1	Mauvais
2	Médiocre
3	Passable
4	Bon
5	Excellant

Tableau1.1: Qualité avec la mesure MOS.

1.7.2 Mesure de distorsion objective

Le système auditif de l'être humain est l'estimateur le plus adéquat de la qualité et des performances d'un codeur de la parole. Il permet de préciser l'intelligibilité et la sonorité naturelle des sons. Bien que, Les tests d'écoute subjectifs donnent une bonne évaluation pour les codeurs de la parole, ils peuvent exiger beaucoup de temps et sont non conformé. Les mesures objectives peuvent donner une estimation immédiate de la qualité perceptuelle de la parole [16].
Les mesures objectives de distorsions peuvent être calculées aussi bien dans le domaine temporel que fréquentiel [4].

Les performances d'une mesure objective résident dans sa corrélation avec la mesure subjective correspondante (qualité ou intelligibilité).

Les mesures de distorsions sont classifiées en trois domaines [2] [4] :

- Domaine temporel (RSB et RSBseg)
- Domaine fréquentiel (distorsion spectrale)
- Domaine perceptuel (EMBSD)

1.7.2.1 Domaine temporel

- Rapport Signal sur Bruit :

Si $\{S(n)\}_{n=0.Nt}$ sont les N_t échantillons du signal parole original et $\{\check{S}(n)\}_{n=0.Nt}$ sont les N_t échantillons du signal parole codé dans le *RSB* à la forme suivante :

$$RSB = 10\log\left(\frac{\sum_{n=0}^{Nt-1} S(n)^2}{\sum_{n=0}^{Nt-1}\left[S(n)-\widetilde{S}(n)\right]^2}\right) \quad (dB) \tag{1.27}$$

Le *RSB* donne une valeur après avoir traité tout le fichier, donc il n'y a pas moyen de retrouver les instants ou les divergences ont été enregistrées. De plus le *RSB* est dominé par la portion de forte énergie (tranches voisées), alors que le bruit a un effet perceptuel plus important sur les portions de faibles énergies.

- Rapport Signal sur Bruit segmenté :

Le RSB_{seg} mesuré en dB, est la moyenne du *RSB* calculé sur de courts intervalles de temps du signal parole. Le RSB_{seg} calculé sur N_F trames de longueur Ns est donné par :

$$RSB_{seg} = \frac{1}{N_F}\sum_{i=0}^{N_F-1} 10\log\left(\frac{\sum_{j=0}^{Ns-1} S(N_s i+j)^2}{\sum_{j=0}^{Ns-1}\left[S(N_s i+j)-\widetilde{S}(N_s i+j)\right]^2}\right) \quad (dB) \tag{I.28}$$

Le RSB_{seg} est meilleur que le *RSB*. Cependant, les tranches de silences renvoient de grandes négatives, biaisant de la sorte le résultat final. Ce problème peut être résolu en éliminant dans le calcul de la distorsion les trames de silence.

1.7.2.2 Domaine fréquentiel

La distorsion spectrale est définie comme étant la racine carrée de la moyenne au carrée des différence entre le logarithmique décimale du spectre LPC original et le logarithme décimale du spectre LPC quantifier. La définition mathématique est comme suit :

$$DS_i = \sqrt{\frac{1}{F_e}\int_0^{F_e}\left[10Log_{10}\frac{S_i(f)}{\widetilde{S}_i(f)}\right]^2 df} \qquad (dB) \tag{1.29}$$

Où F_e est la fréquence d'échantillonnage, $S_i(f)$ et $\widetilde{S}_i(f)$ sont les spectres de trame i donnés par :

$$S_i(f) = \frac{1}{A_i(e^{j2\pi f/F_e})} \tag{1.30}$$

$$\widetilde{S}_i(f) = \frac{1}{\widetilde{A}_i(e^{j2\pi f/F_e})} \tag{1.31}$$

Ou, $A_i(z)$ *et* $\widetilde{A}_i(z)$ sont respectivement, les polynômes PL original et quantifié vus plus haut, pour la trame i, au lieu de l'intégration, une sommation des cœfficients obtenus après application de la TFD (transformée de Fourier Discret) aux cœfficients LPC, peut utilisée pour calculer DS_i. La distorsion devient donc :

$$DS_i = \sqrt{\frac{1}{n_1 - n_0}\sum_{k=n_0}^{n_1-1}\left[10\log\frac{S_i(e^{j2\pi k/N})}{\widetilde{S}_i(e^{j2\pi k/N})}\right]^2} \qquad (dB) \tag{1.31}$$

Dans notre travail, les signaux d'entrés sont échantillonnés à F_e=8 KHz et nous avons calculé la distorsion sur une bande allant de 0 KHz à 3 KHz avec une TFD sur N+256 points. Ce qui donne $n_0 = 0$ et $n_1 = 95$. La distorsion fréquentielle est de 31.25 Hz (8000/256).

Une distorsion spectrale moyenne (la moyenne des distorsions spectrales calculées pour toutes les trames) de 1 dB est habituellement acceptée. Cependant, selon ***Atal*** et ***Paliwal*** les conditions de transparence spectrale (pas de distorsion audible) établies expérimentalement sont les suivantes :

- La moyenne DS inférieur à 1dB
- Le nombre de trames ayant DSi dans l'intervalle 2-4 dB est inférieur a 2%
- Pas de trames ayant DSi supérieur a 4 dB

1.7.2.3 Mesure de distance euclidienne LSP pondérée

Cette distance a été développée le but d'optimiser le quantification des paramètres LP, elle a la forme suivante :

$$d_{LSF} = \sum_{i=1}^{p} \left[c_i w_i (\omega_i - \tilde{\omega}_i) \right]^2 \tag{1.32}$$

Ou c_i et w_i sont les poids du i^{eme} coefficients LSP ω_i, et p est l'ordre du filtre LP. Pour un filtre d'ordre 10, les poids fixes c_i sont donnés par :

$$c_i = \begin{cases} 1.0 & pour \quad 1 \le i \le 8 \\ 0.8 & pour \quad i = 9 \\ 0.4 & pour \quad i = 10 \end{cases} \tag{1.33}$$

Ces poids sont utilisés pour donner plus d'importance aux basses fréquences par rapport aux hautes fréquences. Ceci est justifié par le fait que l'oreille humaine est plus sensible aux basses fréquences qu'aux hautes fréquences. Les poids adaptatifs ω_i sont utilises pour accentuer les régions de l'enveloppe spectrale $S(e^{j\omega})$ à forte énergie (formants). Ces poids sont données par :

$$w_i = \left[S(e^{j\omega}) \right]^r \tag{1.34}$$

Ou, *r* est une constante empirique qui contrôle le degré de la pondération, empiriquement *r*=0.15. Une pondération plus simple a été proposée par [17], elle a la forme suivante :

$$w_i = \frac{1}{\omega_i - \omega_{i-1}} + \frac{1}{\omega_{i+1} - \omega_i} \qquad ou \quad \omega_0 = 0 \, et \quad \omega_{p+1} = \pi \tag{1.35}$$

Les mesures dans le domaine perceptuel sont basées sur les modèles d'audition humaine. Le signal est transformé vers un domaine adéquat de telle manière qu'on puisse exploiter effets de masquage psycho-acoustique. Parmi les mesures perceptuelles les plus utilises nous pouvons citer : Perceptuel Evaluation of Speech Quality (*PESQ*) et Enhanced Modified Bark Spectrum Distorsion (*EMBSD*).

L'EMBSD estime la distorsion perceptuel d'un signal en le comparant au signal original dans le domaine des sons forts (loudness domain) tout en tenant compte du seuil de masquage de bruit modifié et du modèle cognitif basé su le post-masquage.

Conclusion

La prédiction linéaire exploite la redondance dans le signal parole et extrait des coefficients (paramètres LPC) qui caractérisent le comportement du signal. La simplicité de son concept, la linéarité dans la résolution des systèmes et ses performances dans le codage de la parole, la rendent la plus admise et la plus largement utilisée dans le codage du signal de parole.

Chapitre 2

Codage de formes d'ondes

Introduction

Un système de codage de la parole comprend deux parties: le codeur et le décodeur (codec). Le codeur analyse le signal pour en extraire un nombre réduit de paramètres pertinents qui sont représentés par un nombre restreint de bits pour archivage ou transmission. Le décodeur utilise ces paramètres pour reconstruire un signal de parole synthétique.

Les algorithmes de codage de la parole peuvent être divisés en trois catégories [18]

- Codage de forme d'onde (waveform coding).
- Codage paramétrique (parametric coding).
- Codage hybride (hybrid coding).

La figure 2.1 montre la différence de qualité de parole qui existe entre les codecs.

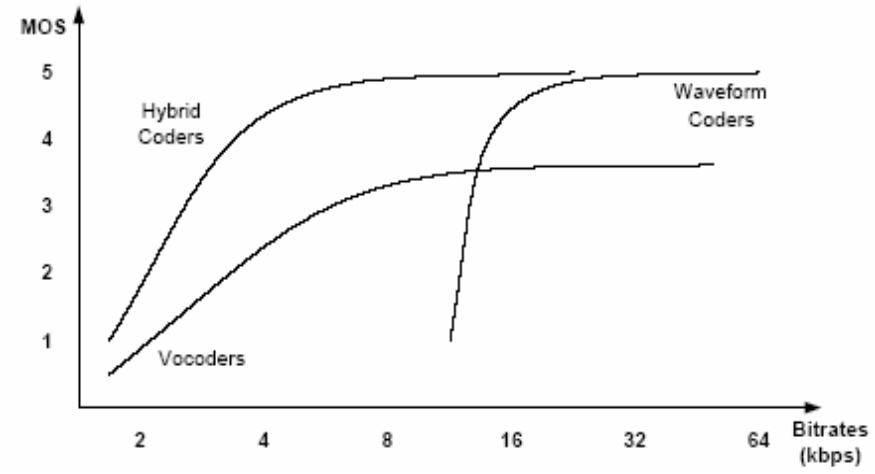


Fig. 2.1 Comparaison de la qualité de codage de parole [18]

2.1 Le codage de forme d'onde

Les codeurs de formes d'ondes sont relativement simples à mettre en œuvre, ils produisent une qualité acceptable jusqu'à des débits de 16 Kbits/s. En deçà, la qualité du signal reconstruit se dégrade rapidement.
L'algorithme de codage le plus simple est celui qui revient seulement à échantillonner un signal analogique et à quantifier les échantillons (c'est à dire à les convertir des valeurs réelles en valeurs de précision finie) ; ce codage est appelé PCM (*Pulse Code Modulation*).

Codeurs PCM

La PCM (Pulse Code Modulation) est une méthode de codage de forme d'onde définie dans le cahier des charges d'ITU standard G.711. C'est le type le plus simple de codage de forme d'onde. C'est un processus de quantification échantillon par échantillon. N'importe quelle formule de la quantification scalaire peut être utilisée avec cet arrangement, mais la formule la plus commune de la quantification utilisée est la quantification logarithmique.

Le codage PCM est à la base d'une famille de codages différentiels qui est basé sur l'observation que des échantillons successifs d'une source audio sont fortement corrélés. Il semble donc judicieux d'encoder non pas les échantillons eux même mais la différence entre des échantillons successifs. On peut citer le codage DPCM (*Differential PCM*), ADPCM (*Adaptive Differential PCM*) et ADM (*Adaptive Delta Modulation*).

Codeurs DPCM et ADPCM

La PCM ne fait aucune prétention au sujet de la nature de la forme d'onde à coder, par conséquent cela fonctionne bien pour des signaux parole non voisés. Cependant, dans le codage de la parole il y a une corrélation très élevée entre les échantillons adjacents. Cette corrélation doit être employée pour réduire le débit binaire.
Une méthode simple de faire ceci est de transmettre seulement les différences entre chaque échantillon.

Ce signal de différence aura un intervalle dynamique beaucoup plus inférieur que le discours initial. Dans cette méthode, l'échantillon précédent est employé pour prévoir la valeur de l'échantillon actuel. La prévision serait améliorée si une plus grande case du discours est employée pour faire la prévision.

Cette technique est connue en tant que modulation d'indicatif d'impulsion différentielle (DPCM). Sa structure est montrée dans fig. 2.2.

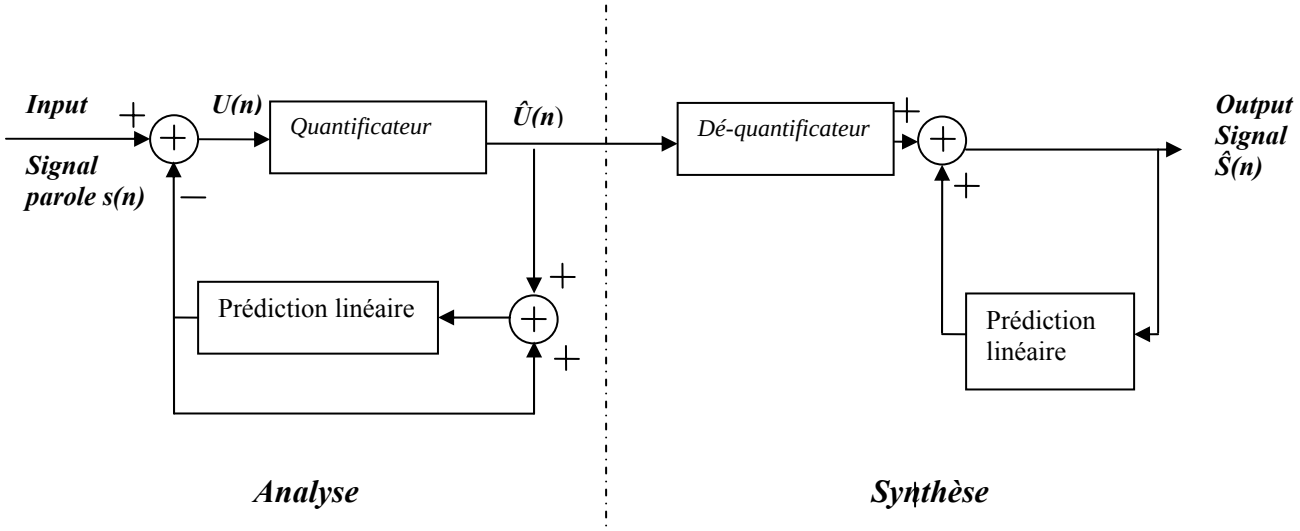


Fig.2.2. codeur PCM

Une version améliorée de la DPCM est la DPCM adaptatif dans laquelle le prédicteur et le quantificateur sont adaptés aux caractéristiques locales du signal d'entrée. Il y a un certain nombre de recommandations d'ITU basé sur des algorithmes d'ADPCM pour le discours à bande étroite (de qualité de prélèvement de 8 KHz) codage sonore par exemple, G.726 fonctionnant à 40, 32, 24 et 16 kbps. La complexité des codeurs ADPCM est assez basse.

2.2 Le codage par synthèse

Connu aussi sous le nom de codage de source ou vocodeurs (voice coders), ces codeurs sont destinés à fonctionner pour des bas débits et sont destinés à maintenir l'intelligibilité de la parole. La plupart de ces codeurs sont basés sur le codage linéaire prédictif LP. La performance de ce type de codage dépend du modèle de production de la parole.

Le codage LP consiste à synthétiser des échantillons à partir d'un modèle d'un système de production vocal et d'une excitation. Pour la voix humaine, le système de production vocal est l'ensemble poumons-cordes vocales -trachée -gorge -bouche -lèvres. En pratique, on modélise ce système par un ensemble de cylindres de diamètres différents, 10 dans le cas de LP-10, excités par un signal qui est soit une sinusoïde, soit un bruit blanc. Le choix de la fonction d'excitation (sinusoïde ou bruit blanc) dépend des caractéristiques, voisée ou non voisée, du signal.
Ce modèle suppose que la parole est produite en excitant un filtre linéaire variable par un bruit blanc pour les segments de la parole non voisée, ou un train d'impulsions pour les sons voisés comme le montre la figure 2.3 .

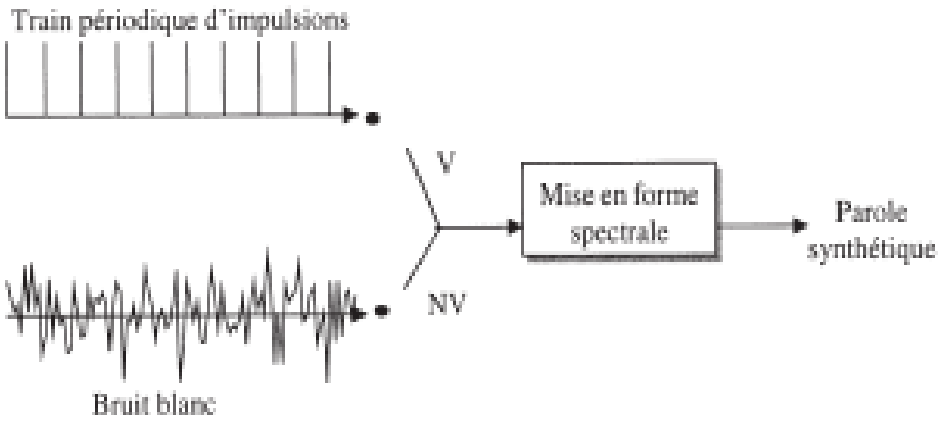


Fig.2.3 Synthèse dans un vocodeur à 2 états d'excitation.

2.3 Le Codage Hybride

La qualité des codeurs de formes d'ondes chute rapidement pour des débits inférieurs à 16 kbits/s, et comme les vocodeurs apportent une amélioration négligeable dans la qualité à des débits supérieurs à 4 kbits/s, Les codeurs hybrides sont alors utilisés pour combler ce vide, donnant ainsi une qualité de la parole à des débits moyens. Cependant, ces codeurs ont tendance à nécessiter un nombre d'opérations plus élevé. Virtuellement, tous les codeurs hybrides reposent sur l'analyse LPC pour l'obtention des paramètres du modèle de synthèse. Les techniques de formes d'ondes utilisées pour coder le signal d'excitation et les modèles de production du pitch peuvent être incorporés pour améliorer les performances.

A partir des années 80, l'intérêt pour les codeurs CELP (Code Exited Linear Prediction) ne cesse d'augmenter. Ces codeurs sont basés sur les algorithmes de codage de la parole les plus actuellement utilisés dans la téléphonie sans fil. Dans les codeurs CELP, l'analyse LP est utilisée pour obtenir le signal d'excitation. La modélisation du pitch est utilisée pour coder efficacement le signal d'excitation.

Beaucoup de codeurs de CELP ont été normalisées, incluant G.723.1 [68, 36] fonctionnant à 6.3/5.3 kbps, le standard G.729 de l'ITU est un codeur CELP qui produit une qualité téléphonique (toll quality) de la parole à 8 kbits/s [68], G.728 fonctionnant à 16 kbps. Le codeur d'interpolation de forme d'onde est également un codeur hybride.

Nous nous somme intéressés au codage de la parole par interpolation de formes d'ondes dont notre objectif est l'extraction des formes d'ondes. Nous détaillerons dans ce qui suit toutes les étapes pour arriver à notre but.

2.4 Codeur de la parole par interpolation de formes d'ondes

2.4.1 Origine et principes du codage WI

L'importance de la perception de la périodicité dans la parole voisée est à l'origine du développement de la technique de codage par interpolation de la forme d'onde. Cette technique a été introduite, en premier lieu, par W. B. Kleijn [20] et la première version était appelée Prototype Waveform Interpolation (PWI). La PWI codait les segments voisés seulement et, par conséquent, elle était utilisée en combinaison avec d'autres codeurs tels que le CELP pour coder les segments non voisés.

La PWI exploite le fait que les formes d'ondes de longueur égale à la période du pitch (période fondamentale) évoluent lentement dans le temps. Cette évolution lente des formes d'ondes suggère qu'on n'a pas besoin de transmettre toutes les périodes de la trame au décodeur ; au lieu de cela, on peut les transmettre à des intervalles réguliers. Au décodeur, les formes d'ondes non transmises sont retrouvées au moyen d'une interpolation. De cette manière, le degré de

périodicité de la parole voisée sera mieux contrôlé et, par conséquent, on obtient une parole voisée reconstituée de haute qualité [21]. Dans la PWI, les périodes du signal sélectionnées pour être transmises sont dites formes d'ondes prototypes (Prototype Waveforms).

Bien que la PWI travaille remarquablement bien avec les segments voisés, elle a le défaut de ne pas pouvoir être appliquée aux segments non voisés. En d'autres termes, elle doit toujours être utilisée avec une autre méthode de codage de la parole pour manipuler les segments non voisés. Ainsi, la commutation entre les codeurs devient inévitable et réduit considérablement la robustesse du codeur. En 1994, la PWI a été raffinée pour devenir la WI qui est capable de prendre en charge les sons voisés et non voisés [23, 28]. Similaire à la PWI, la WI représente un signal parole avec une séquence de forme d'onde. Pour la parole voisée, ces formes d'ondes sont simplement de longueurs égales à la période du pitch (pitch cycles).

Pour la parole non voisée et le bruit de fond, les formes d'ondes sont de différentes longueurs et contiennent des signaux assimilables à du bruit. Puisque les formes d'ondes ne sont plus limitées à la période du pitch, il n'est plus approprié d'utiliser le terme forme d'onde prototype ou pitch-cycle. A la place, on adopte le terme forme d'onde caractéristique (Charactéristic Waveform) qui sera abrégé par CW par la suite.

Une différence clé entre la WI et la PWI est que les formes d'ondes dans la WI sont prélevées à une fréquence plus grande. Cependant, une augmentation de la fréquence de prélèvement des formes d'ondes entraînera une augmentation du débit. Pour contrer ce problème, la WI décompose la CW en une forme d'onde à évolution lente (SEW) et une forme d'onde a évolution rapide (REW). La SEW représente la composante quasi-périodique du signal parole tandis que la REW représente la composante non périodique et le bruit restants dans le signal. Puisque les deux formes d'ondes ont des propriétés différentes du point de vue perception, elles sont quantifiées séparément pour améliorer l'efficacité du codage.

2.4.2 Vue d'ensemble du codeur WI

La figure 2.4 présente un schéma bloc du codeur WI. On peut le diviser en deux couches : la couche d'analyse-synthèse et la couche de quantification. Dans la première couche, le bloc d'analyse (processeur 100) exécute, d'abord, l'analyse LPC sur le signal parole entrant et fournit le signal résiduel. Puis, le pitch est estimé et le signal résiduel est, alors, décomposé en une suite de CW. Ces CW sont, alors, alignées et normalisées en puissance pour donner une surface (signal à deux dimension) qui illustre l'évolution des formes d'ondes à travers la trame. L'étage de synthèse (processeur 200) effectue l'opération inverse de celle de l'analyse. Le signal résiduel est reconstruit à partir des CW et envoyé au filtre de synthèse LP où le signal parole est, finalement, reconstitué.

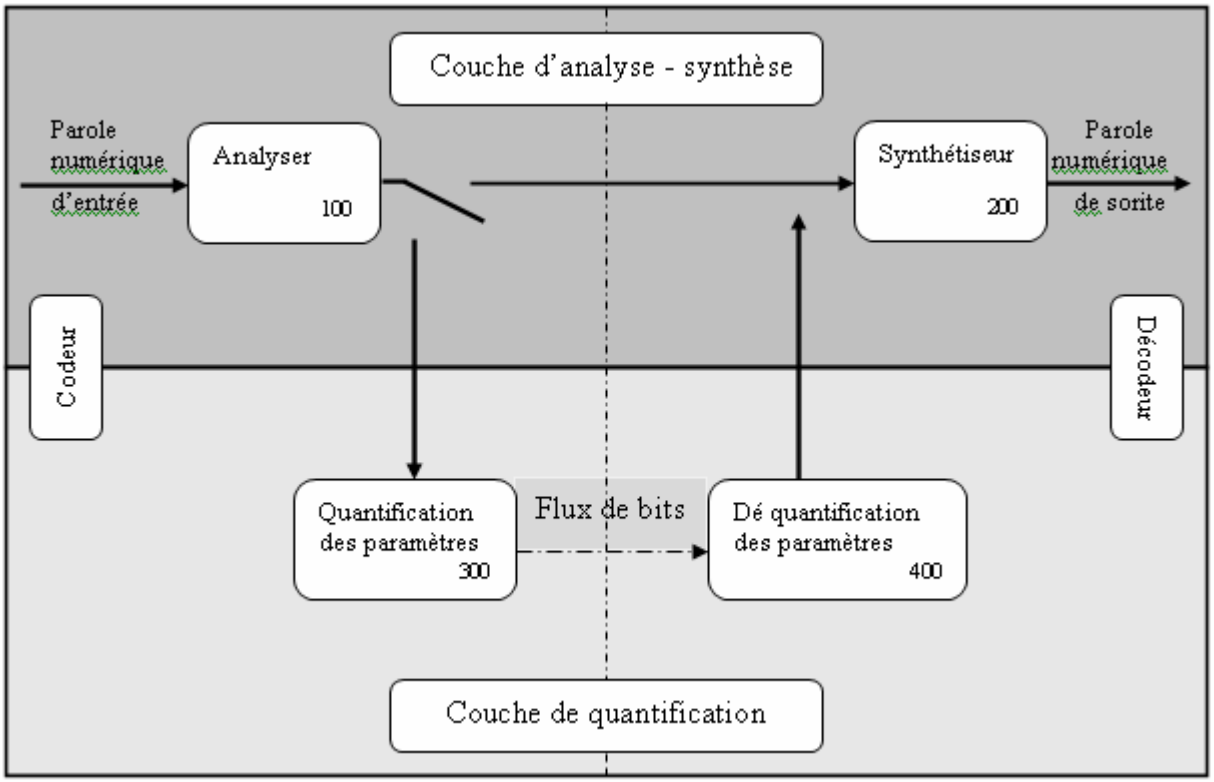


Fig.2.4 Schéma bloc d'un système de codage WI. Le commutateur permet au codeur d'éviter la couche de quantification et nous permet de mesurer la performance de la couche d'analyse-synthése.

Le processeur **300** dans la couche quantification exécute la décomposition en SEW/REW et la quantification des paramètres. Le processeur **400** au récepteur dé-quantifie et reconstitue les CW à partir des SEW et REW transmises.

Dans ce chapitre, on va discuter de la couche d'analyse-synthèse qui comprend les éléments clés de la technique WI comme l'extraction du pitch, l'extraction des CW, leur alignement et leur interpolation, et de la couche de quantification. Cette étude est basée largement sur le travail de W.B.Kleijn sur la WI.

Pour chaque processeur dans la couche, on donnera les détails d'implémentation avec les calculs mathématiques appropriés. Pour faciliter la discussion, on donnera les schémas détaillés des processeurs sélectionnés.

2.4.3 Représentation des formes d'ondes caractéristiques

Avant de rentrer dans les détails de chaque processeur, on commence, d'abord, par choisir une représentation mathématique appropriée pour les CW. Comme on va le voir au fur et à mesure, la majorité des calculs dans la WI sont associés aux CW, il est donc crucial d'avoir la meilleure représentation des CW qui permet de réduire la complexité du codeur.

Les CW sont, finalement, utilisées pour construire une surface bidimensionnelle décrivant l'évolution des formes d'ondes du signal résiduel. Ainsi, la représentation des CW recherchée doit permettre d'avoir un signal bidimensionnel.

Pour commencer, on considère une seule CW unidimensionnelle. La CW est une séquence de valeurs réelles à temps discret de longueur égale à la période du pitch. Donnons la notation $s(m)$ à la CW de longueur P (Pitch period) :

$$s(m) \in \Re \qquad m= 0, 1, \ldots, P-1 \qquad (2.1)$$

Une partie du traitement dans la WI est faite dans le domaine fréquentiel. Ceci implique qu'une représentation temps- fréquence serait très favorable. Nous avons, donc, choisi la représentation en série de Fourier à temps discret (DTFS : Discrete Time Fourier Series) où $s(m)$ peut être exprimée par :

$$s(m) = \sum_{k=0}^{P/2}\left[A_k \cos\left(\frac{2\pi km}{P}\right) + B_k \sin\left(\frac{2\pi km}{P}\right)\right] \qquad 0 \le m < P \qquad (2.2)$$

Où $\{A_k\}$ et $\{B_k\}$ sont les coefficients de Fourier à temps discret (DTFS) calculés à l'aide d'un ensemble d'équations de transformation.

Plus précisément, si P est pair :

$$\left.\begin{aligned} A_k &= \frac{2}{P}\sum_{m=0}^{P-1}\left[s(m)\cos\left(\frac{2\pi km}{P}\right)\right] \\ B_k &= \frac{2}{P}\sum_{m=0}^{P-1}\left[s(m)\sin\left(\frac{2\pi km}{P}\right)\right] \end{aligned}\right\} \quad pour \quad k = 1,2,\ldots,P/2-1$$

$$\left.\begin{aligned} A_k &= \frac{1}{P}\sum_{m=0}^{P-1}\left[s(m)\cos\left(\frac{2\pi km}{P}\right)\right] \\ B_k &= \frac{1}{P}\sum_{m=0}^{P-1}\left[s(m)\sin\left(\frac{2\pi km}{P}\right)\right] \end{aligned}\right\} \quad pour \quad k = 0 \quad et \quad P/2 \qquad (2.3)$$

Quand P est impair :

$$\left.\begin{aligned} A_k &= \frac{2}{P}\sum_{m=0}^{P-1}\left[s(m)\cos\left(\frac{2\pi km}{P}\right)\right] \\ B_k &= \frac{2}{P}\sum_{m=0}^{P-1}\left[s(m)\sin\left(\frac{2\pi km}{P}\right)\right] \end{aligned}\right\} \quad pour \quad k = 1,2,\ldots,(P-1)/2$$

$$\left.\begin{aligned} A_k &= \frac{1}{P}\sum_{m=0}^{P-1}\left[s(m)\cos\left(\frac{2\pi km}{P}\right)\right] \\ B_k &= \frac{1}{P}\sum_{m=0}^{P-1}\left[s(m)\sin\left(\frac{2\pi km}{P}\right)\right] \end{aligned}\right\} \quad pour \quad k = 0 \qquad (2.4)$$

La forme d'une CW peut, maintenant, être décrite par un ensemble de coefficients DTFS $\{A_k, B_k\}$. Notons que l'indice m dans (2.2) n'est pas nécessairement entier; il peut prendre n'importe quelle valeur réelle dans l'intervalle $0 \leq m < P$. En d'autres termes, les valeurs situées entre deux instants discrets ($s(1.3)$, par exemple) peuvent être calculées aisément par (2.2).

Après avoir obtenu la représentation pour une CW, nous somme, maintenant, prêts à construire une représentation bidimensionnelle pour une séquence de CW. En fait, cette représentation est simplement obtenue en ajoutant une modification à (2.2). Ainsi, on attache un indice de temps discret n à tous les paramètres dans (2.2) qui varient dans le temps. Ces paramètres sont $\{A_k\}$, $\{B_k\}$ et P.

L'équation (2.2) peut donc être écrite comme suit :

$$s(n,m) = \sum_{k=0}^{P(n)/2} \left[A_k(n) \cos\left(\frac{2\pi km}{P(n)}\right) + B_k(n) \sin\left(\frac{2\pi km}{P(n)}\right) \right] \qquad 0 \leq m < P(n) \quad (2.5)$$

où les coefficients $\{A_k(n)\}$ et $\{B_k(n)\}$ sont, maintenant, variants dans le temps de même que la valeur du pitch P(n). Il faut noter que nous avons ignoré les coefficients A_0 et B_0 dans l'équation (l'indice k commence à partir de k = 1 au lieu de k = 0). Ceci est du au fait que B_0 dans (2.3) et (2.4) est un coefficient redondant ($\sin(0) = 0$). D'un autre côté, A_0 représente la composante DC du signal et n'a aucune importance vis à vis de la perception. Par conséquent, ces deux coefficients peuvent être ignorés.
L'équation 2.5 est, à présent, la représentation d'un signal bidimensionnel où m et n sont les variables courantes. Chaque CW évolue le long de l'axe m et la forme des CW évolue à travers le temps le long de l'axe n.

Cependant, la longueur de la CW dans (2.5) dépend du pitch P(n) variant dans le temps ; les CW à des instants différents peuvent avoir des longueurs différentes. Il est, généralement, plus convenable de normaliser toutes les CW à une longueur commune.

Cette normalisation peut être accomplie en substituant :

$$\phi = \phi(m) = \frac{2\pi m}{P(n)} \qquad (2.6)$$

dans (2.5) et on peut obtenir :

$$S(n,\phi) = \sum_{k=1}^{P/2} \left[A_k(n)\cos(k\phi) + B_k(n)\sin(k\phi) \right] \qquad 0 \le \phi(.) < 2\pi \qquad (2.7)$$

De cette manière, toutes les CW ont la même longueur 2л. La figure 2.5 donne une illustration de cette normalisation et un exemple d'une surface bidimensionnelle représentée par (2.7).

Remarques sur la représentation en DTFS :

A première vue, $B_{P/2}$ dans (2.3) semble être un coefficient redondant puisque sin(mл)=0 pour tout entier m. En fait, ce n'est pas entièrement vrai. Comme nous le verrons au paragraphe 2.4.4.5, ce coefficient particulier ne sera plus égal à zéro quand le signal subit un décalage dans le temps dans le processeur d'alignement.

Généralement, représenter un signal par ses coefficients DTFS implique que le signal est répété de façon périodique. De même, représenter une CW par les DTFS signifie qu'elle est extraite d'un signal périodique.

Les représentations dans le domaine temps peuvent réduire la complexité du codeur dans une certaine mesure en évitant les transformations en DTFS directe et inverse. Néanmoins, elles peuvent être problématiques dans les traitements liés à la fréquence [27].

2.4.4 Étage d'analyse

Pour commencer, on va se concentrer sur le processeur d'analyse **100**. Comme il est déjà mentionné, le but fondamental de ce processeur est de décomposer le signal parole en une série de CW (une surface bidimensionnelle) et d'extraire d'autres paramètres orthogonaux tels que les LSF, l'énergie et le pitch. La figure 2.3 montre tous les processeurs que comprend la couche d'analyse.

2.4.4.1 Analyse LP

Chaque trame de parole entrante est, tout d'abord, envoyée au processeur **130** où elle subit une analyse LP d'ordre 10 pour en extraire l'ensemble $\{a_k\}$ des coefficients LP. Avant cela, le signal parole subit une pré-accentuation en utilisant $\alpha = 0.1$ dans (2.15). Cette opération a pour but de compenser la perte de l'énergie des composantes haute fréquence due au filtrage passe – bas pendant la conversion A/D. La parole pré-accentuée est, alors, fenêtrée en utilisant la fenêtre de Hamming définie dans (2.3) avec $L_w = 240$. Le centre de la fenêtre coïncide avec l'extrémité droite de la trame courante. En d'autres termes, la fenêtre couvre 120 échantillons de la trame courante et 120 de la trame future. Ces 120 échantillons futurs provoquent un retard algorithmique de 15ms. La méthode d'auto-corrélation est appliquée à cette fenêtre de parole pour générer les coefficients du filtre $\{a_k\}$. Ces $\{a_k\}$ sont modérés en utilisant $\gamma = 098829$ ce qui est équivalent à une extension de la largeur de bande égale à 30 Hz. Les coefficients résultants sont convertis en coefficients LSF et envoyés au processeur **120**.

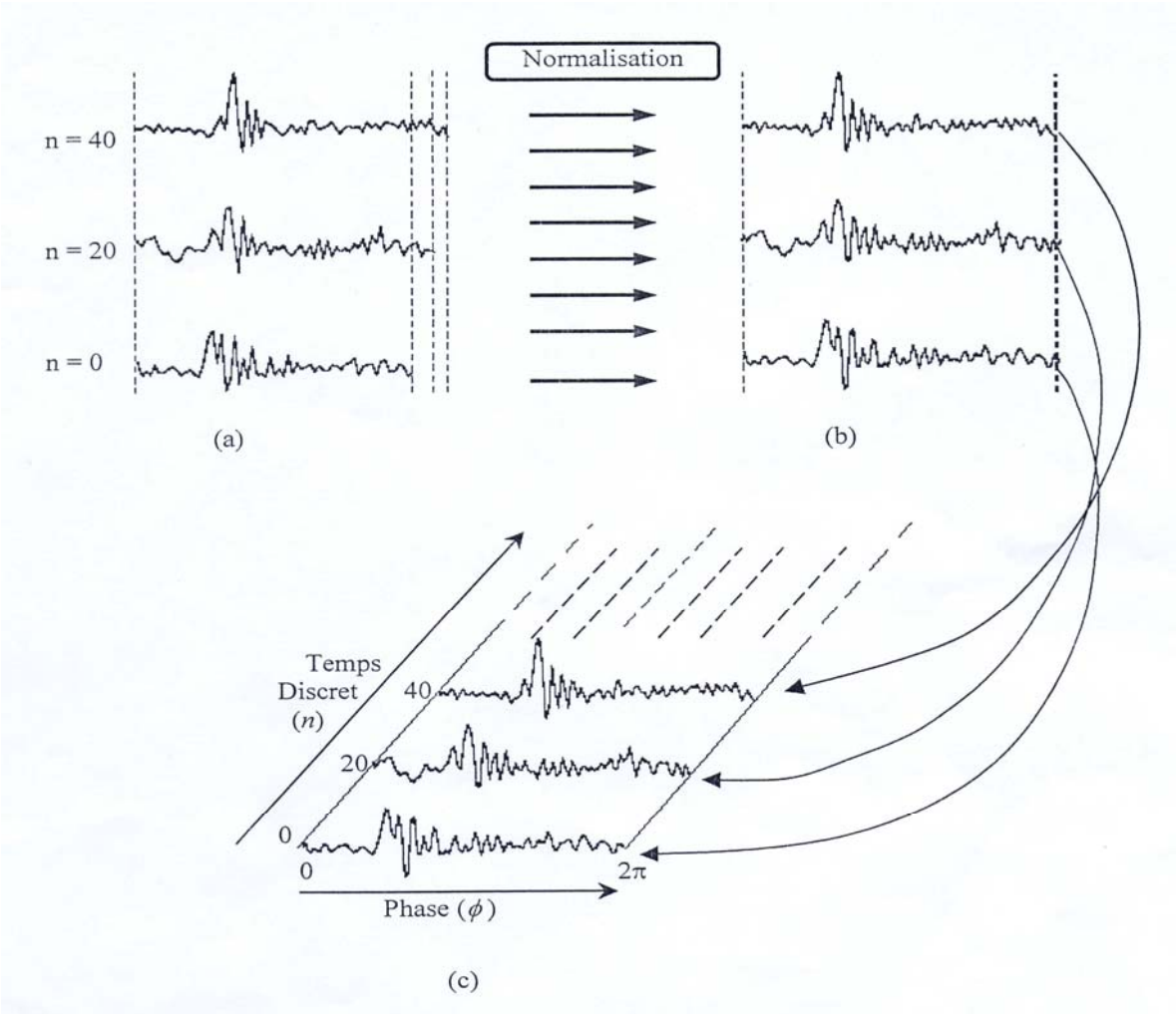


Fig. 2.5 Exemple d'une surface de formes d'ondes caractéristiques. **(a)** Les CW (pré alignées) sont prélevées aux instants n= 1, 9, 17. On remarque qu'elles ont des longueurs différentes. **(b)** Les CW après normalisation. **(c)** Formation de la surface d'évolution des CW. Chaque CW évolue le long de l'axe de fi et l'évolution des CW dans le temps se fait sur l'axe n.

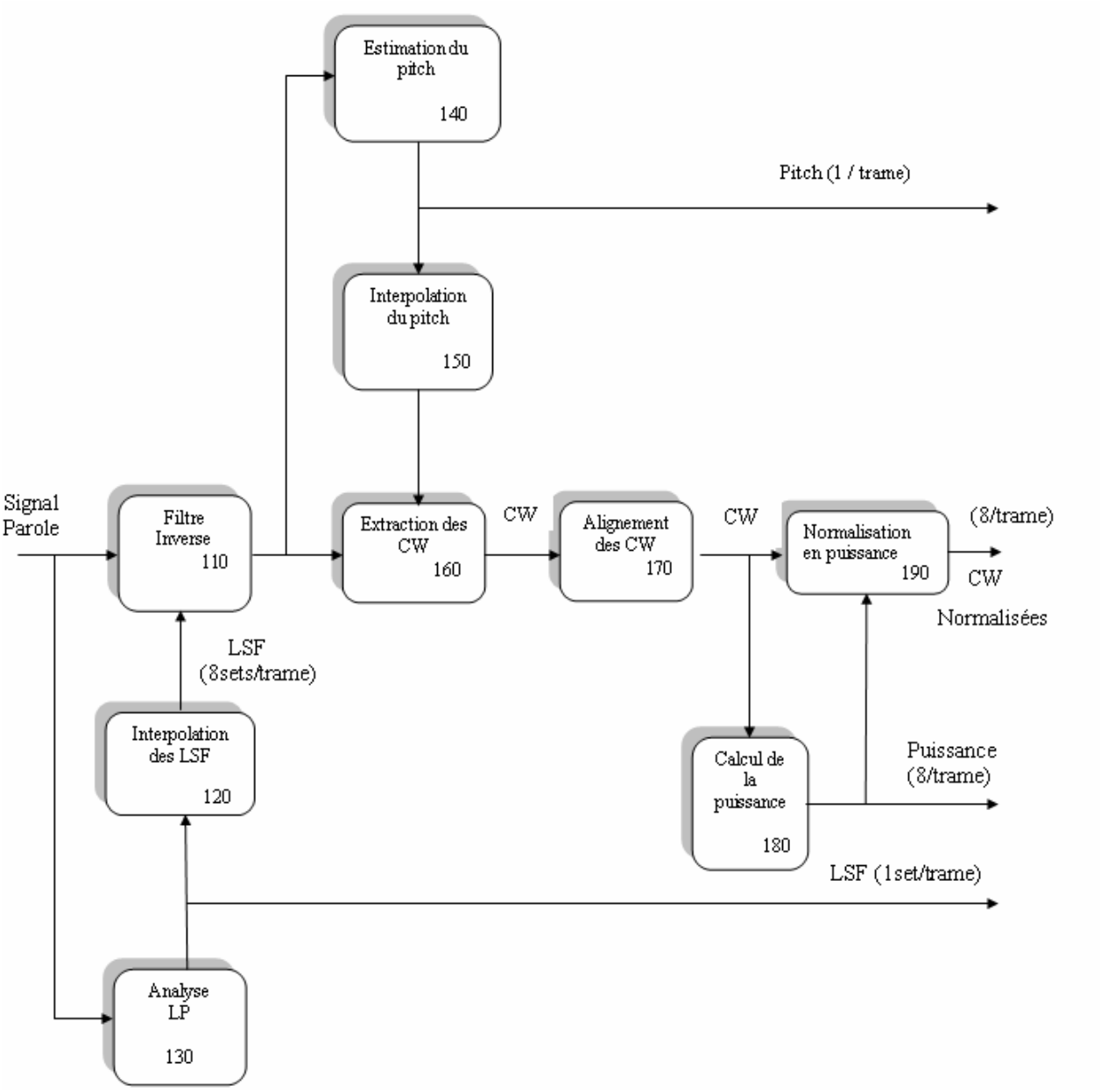


Fig. 2.6 Schéma bloc de la couche d'analyse de la WI (processeur 100). Les processeurs colorés travaillent à la fréquence des sous-trames tandis que les autres travaillent à celle des trames.

2 .4.4.2 Estimation du pitch

Les échantillons du signal résiduel (y compris les 120 échantillons de la trame future) sont envoyés au processeur **140** qui effectue l'estimation du pitch. Dans la technique WI, la précision de l'estimateur du pitch est très cruciale pour la performance du codeur. En particulier, l'opération d'extraction au codeur (processeur **160**) et l'interpolation (processeurs **230** et **250**) au décodeur reposent lourdement sur la valeur estimée du pitch.

Il existe plusieurs procédures d'estimation du pitch. Quelques unes sont basées sur la localisation des « marqueurs de pitch » (le pic dominant dans chaque période pitch du signal résiduel) tandis que d'autres sont basées sur la recherche de la position du maximum d'auto-corrélation ou du gain de prédiction pour une trame d'échantillons. Dans cette implémentation de la WI, on adopte l'algorithme tiré du EVRC (Enhanced Variable Rate Codec) [29] qui appartient à la seconde catégorie.

L'estimation du pitch est effectuée une fois par trame. Pour chaque trame de données, l'estimateur fait deux calculs indépendants sur deux fenêtres qui se recouvrent. La première comprend la trame courante entière et la deuxième fenêtre comprend la seconde moitié de la trame courante et la première moitié de la trame future. Ces échantillons futurs ont déjà été calculés dans le processeur **110**. Donc, l'estimation du pitch n'introduit aucun autre retard au codeur.

Puis, les calculs des gains de prédiction pour toutes les valeurs possibles du retard sont faits séparément pour chaque fenêtre. Ce gain de prédiction, noté β, est défini par :

$$\beta = \max\left\{0, \min\left\{\frac{\sum_{i=0}^{Lf-i-d} r(i)r(i+d)}{\sqrt{\sum_{j=0}^{Lf-i-d} \mathrm{r}^2(j) \sum_{K=0}^{Lf-i-d} \mathrm{r}^2(\mathrm{k}+\mathrm{d})}}, 1.0\right\}\right\} \quad , \ \mathrm{Pmin} \leq \mathrm{d} \leq \mathrm{Pmax} \qquad (2.8)$$

où d est un entier qui représente le retard et r(.) est le signal résiduel. Le dénominateur sert comme facteur de normalisation et les fonction max et min permettent de garder β dans l'intervalle [0, 1]. Si le retard d correspond à la vraie valeur du pitch du signal ou à son multiple entier, le β correspondant sera proche de 1. Par contre, β tend à être considérablement inférieur à l'unité pour toutes les valeurs du retard si le signal ne présente aucun caractère périodique (parole non voisée). Ainsi, dans le but de retrouver le meilleur pitch, on cherche le retard d qui fournit un β maximum. Ce retard sera appelé retard optimal.

Après avoir trouvé le retard optimal pour chaque fenêtre, on utilise quelques seuils pour combiner les retards optimaux des deux fenêtres afin d'obtenir le retard le plus fiable dans la trame

courante. Soit (d_0, β_0) le retard optimal et le gain correspondant de la première fenêtre et (d_1, β_1) ceux de la deuxième fenêtre, le retard final estimé d_{opt} est obtenu par :

Si ($\beta_o > \beta_1 + 0.4$)
{
si ($|d_0 - d_1| > 15$)
$d_{opt} = d_o$
sinon
$d_{opt} = [(d_0 + d_1)/2.0]$
}
sinon
$d_{opt} = d_1$

β_0 et β_1 sont des fonctions de confiance qui indiquent le degré de fiabilité des pitchs estimés (do et d1). Par exemple, si β_0 est plus grand que β_1 cela indique que d_0 est plus fiable que d_1. Il faut noter que les valeurs de d dans (2.8) sont entières. Donc, l'estimateur de pitch décrit par cette équation donne des valeurs entières du pitch..

Remarques sur l'estimation du pitch

• Ce processeur donne toujours une période du pitch même si le signal n'est pas périodique. Dans le cas de parole non voisée où β est faible, la période du pitch varie. Dans ce cas, le pitch est fixé à la valeur minimale Pmin afin de réduire la charge de calcul du codeur. Comme on va le voir dans le paragraphe 2.5.4.4, cette valeur du pitch sera utilisée pour fixer la longueur des CW extraites dans le processeur **160**. Les plus courtes CW permettent de réduire la complexité (les calculs), spécialement dans la transformation en DTFS et dans le processus d'alignement.

• Le calcul du gain de prédiction sur tout l'intervalle des retards (de Pmin à Pmax.) est très onéreux.

2.4.4.3 Interpolation du pitch

Comme déjà mentionné dans 2.4.4.2, le pitch est estimé une seule fois par trame. Cependant, la WI exige une valeur de la période du pitch à chaque point d'extraction dans le processeur **160** pour exécuter l'extraction. Pour résoudre ce problème tout en gardant le même degré de complexité, on utilise un interpolateur de pitch (processeur **150**) pour calculer les pitchs intermédiaires. Bien qu'il existe plusieurs algorithmes d'interpolation du pitch, la technique d'interpolation linéaire classique est suffisante pour la WI.

Si on définit P(n1) et P(n2) comme étant les valeurs des pitchs aux extrémités de la trame courante telles que n1 < n2, alors, le pitch peut être linéairement interpolé par :

$$P(n) = \frac{(\mathrm{n2 - n})P(n1) + (n - \mathrm{n1})P(\mathrm{n2})}{\mathrm{n2 - n}}, \qquad n1 \le n \le n2 \qquad (2.9)$$

où $n_1 - n_2 = L_f = 160$ échantillons dans notre implémentation.

Néanmoins, dans la parole naturelle, plus spécialement, au début et à la fin d'un segment voisé, la valeur du pitch peut doubler, tripler ou diminuer de la moitié [21]. En plus, les estimateurs de pitch souffrent souvent des erreurs fréquentes où le pitch estimé est un multiple entier du vrai pitch. Si on ne fait pas attention et qu'on effectue l'interpolation linéaire à travers ces déviations de la vraie valeur du pitch, le signal parole reconstitué contiendra des pépiements audibles.

Pour corriger ce problème, on interpole les valeurs du pitch comme suit.

Pour le cas où $P(n_1) < P(n_2)$:

$$P(n) = \begin{cases} \dfrac{C(\mathrm{n}_2 - \mathrm{n})P(n_1) + (n - \mathrm{n}_1)P(\mathrm{n}_2)}{\mathrm{C(n2 - n)}}, & pour \quad n_1 \le n < \dfrac{n_1 + n_2}{2}. \\ \dfrac{C(\mathrm{n}_2 - \mathrm{n})P(n_1) + (n - \mathrm{n}_1)P(\mathrm{n}_2)}{\mathrm{C(n2 - n)}}, & pour \quad \dfrac{n_1 + n_2}{2} \le n < n_2. \end{cases} \qquad (2.10)$$

où la constante C est définie comme étant le rapport P(n2) sur P(n1) arrondi au plus proche entier.

Pour $P(n_1) > P(n_2)$:

$$P(n) = \begin{cases} \dfrac{(n_2 - n)P(n_1) + C(n - n_1)P(n_2)}{C(n2 - n)}, & pour \quad n_1 \le n < \dfrac{n_1 + n_2}{2}. \\ \dfrac{(n_2 - n)P(n_1) + C(n - n_1)P(n_2)}{C(n2 - n)}, & pour \quad \dfrac{n_1 + n_2}{2} \le n < n_2. \end{cases} \qquad (2.10)$$

où C est le plus proche entier rapport de $P(n_1)$ sur $P(n_2)$.

Le facteur C peut être considéré comme un indicateur qui nous informe si le pitch est multiple ou sous–multiple du précédent. Quand C est égal à 1, ceci indique qu'il n'y a aucun doublement ou triplement du pitch et les formules précédentes effectueront une simple interpolation linéaire (2.9). D'autre part, quand C est supérieur à 1, ça implique que le pitch est un multiple ou sous-multiple du précédent et l'interpolation décrite par (2.10, 2.11) est réalisée de manière à ce que le pitch change de façon discontinue au point milieu par le facteur C. La figure 2.7 illustre un exemple d'une telle interpolation dans le cas d'un doublement du pitch et dans celui d'une diminution de moitié du pitch.

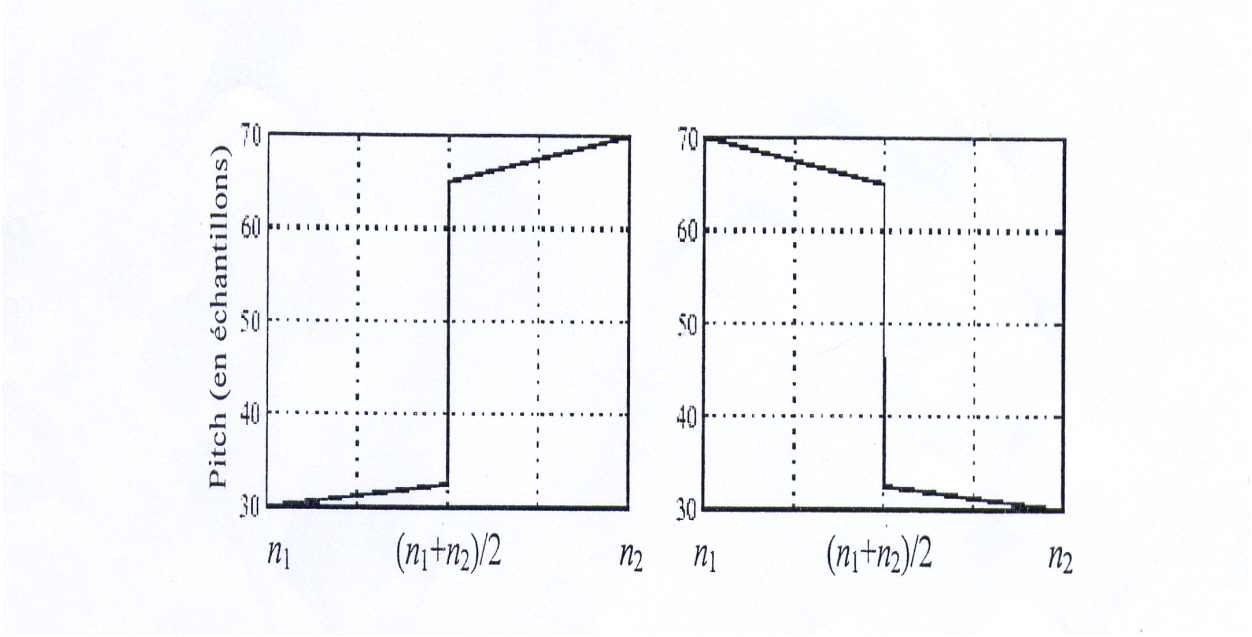


Fig. 2.7 Interpolation du pitch dans le cas d'un doublement de sa valeur. A gauche : interpolation entre 30 et 70 en utilisant (2.10). A droite : vice-versa en utilisant (2.11).

Les valeurs fractionnelles du pitch n'étant pas importantes dans la WI, toutes les valeurs résultantes de (2.9) et (2.10) ou (2.11) sont arrondies aux valeurs entières les plus proches.

2.4.4.4 Extraction des CW

Après avoir estimé et interpolé le pitch, on passe à l'extraction des CW dans le processeur **160**. L'opération d'extraction est effectuée une fois par sous-trame à une fréquence déterminée par le débit d'extraction R_{extr}. En fait, ce débit est lié aux limites de la fréquence fondamentale (donc de la période du pitch). Comme la limite inférieure de la longueur du pitch est égale à 20 échantillons, le nombre de CW à extraire dans une trame de 160 échantillons ne doit pas être inférieur à 160/20 = 8 CW.

Dans le processus d'extraction; on commence par diviser la trame courante en huit intervalles de même longueur. Le point situé sur l'extrémité droite de chaque intervalle sera un point

d'extraction comme illustré dans la figure 2.8a. Donc, deux points d'extraction adjacents seront séparés de 20 échantillons. Cet intervalle définit la longueur L_{sf} de notre sous-trame.

A chaque point d'extraction, on prend le pitch interpolé dans le processeur et on forme une fenêtre d'extraction de cette longueur. La fenêtre d'extraction est centrée au point d'extraction et le signal résiduel contenu dans cette fenêtre formera notre CW extraite. Par conséquent, la CW extraite a toujours la longueur de la période du pitch.

Les CW sont étendues périodiquement pendant la conversion au domaine DTFS. Par conséquent, si aucune attention n'est observée vis à vis des extrémités de la CW pendant l'extraction, cela peut mener à des discontinuités importantes dans la CW périodique (à l'endroit où l'extrémité droite rencontre l'extrémité gauche). De telles discontinuités peuvent causer des distorsions audibles dans la parole reconstituée. Pour éviter cela, le point d'extraction de chaque CW est laissé libre de balayer une certaine plage ε de positions à droite et à gauche de sa position initiale. La position qui donne la plus petite énergie du signal autour des deux extrémités de la fenêtre d'extraction est choisie. La figure 2.8 montre un exemple de l'opération d'extraction. Dans notre implémentation, ε peut prendre des valeurs entre $-\varepsilon_{\min}$ et $+\varepsilon_{\max} = 15$. Des expériences ont montré que $\varepsilon_{\max}$, peut aller jusqu'à 16 échantillons sans affecter la qualité de la parole reconstituée.

Pour calculer efficacement l'énergie des extrémités, on crée d'autres fenêtres appelées fenêtres d'énergie des extrémités centrées sur les deux points extrémités de la fenêtre d'extraction, comme montré sur la figure 2.9. L'énergie des extrémités pour une fenêtre d'extraction est la somme des énergies des échantillons qui entourent les deux extrémités de cette fenêtre. La longueur de la fenêtre d'énergie de chaque extrémité est notée δ qu'il est suffisant de mettre égale à 10 échantillons.

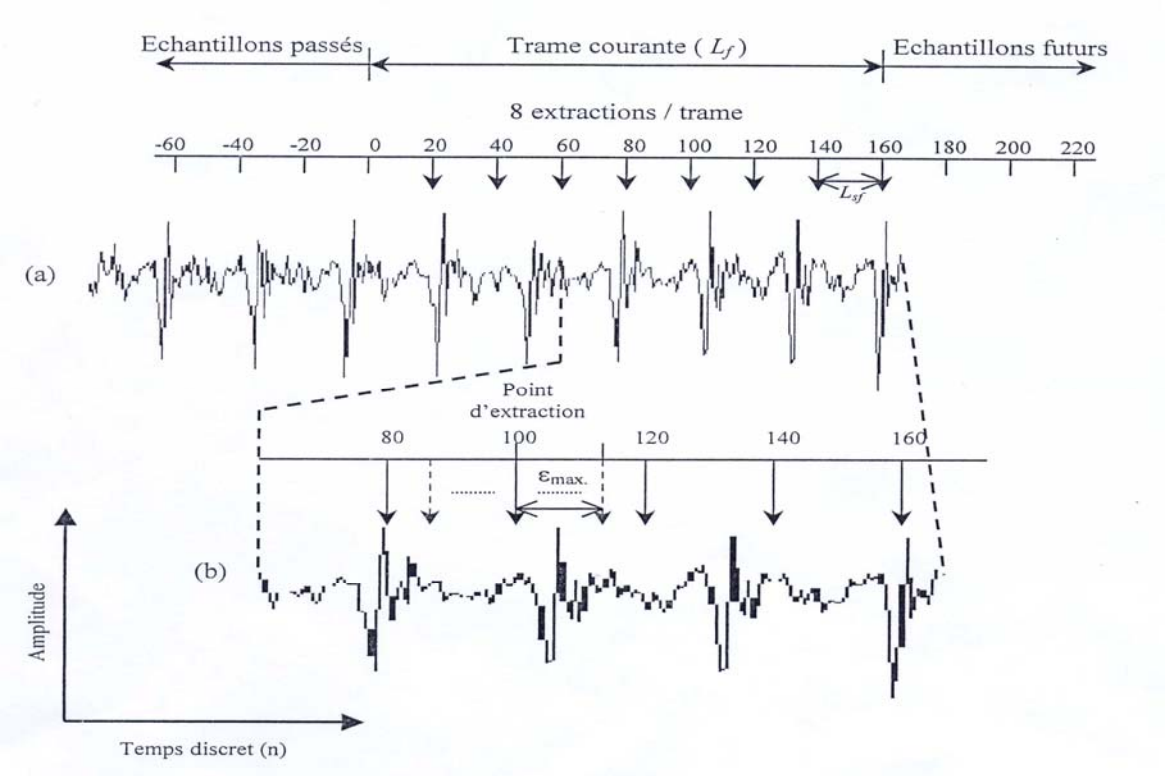


Fig. 2. 8 Exemple d'un point d'extraction libre. **(a)** Les positions originales des points d'extractions des 8 CW. Chaque point d'extraction peut être déplacé légèrement jusqu'à ce que les extrémités de la fenêtre d'extraction soient dans des régions de faible énergie. **(b)** Illustration détaillée pour le point d'extraction à n=100.

En plus de l'extraction, le processeur 160 effectue la transformation des CW au domaine DTFS en utilisant les équations (2.3) et (2.4). Il est à rappeler que les coefficients Ao et Bo peuvent être ignorés.

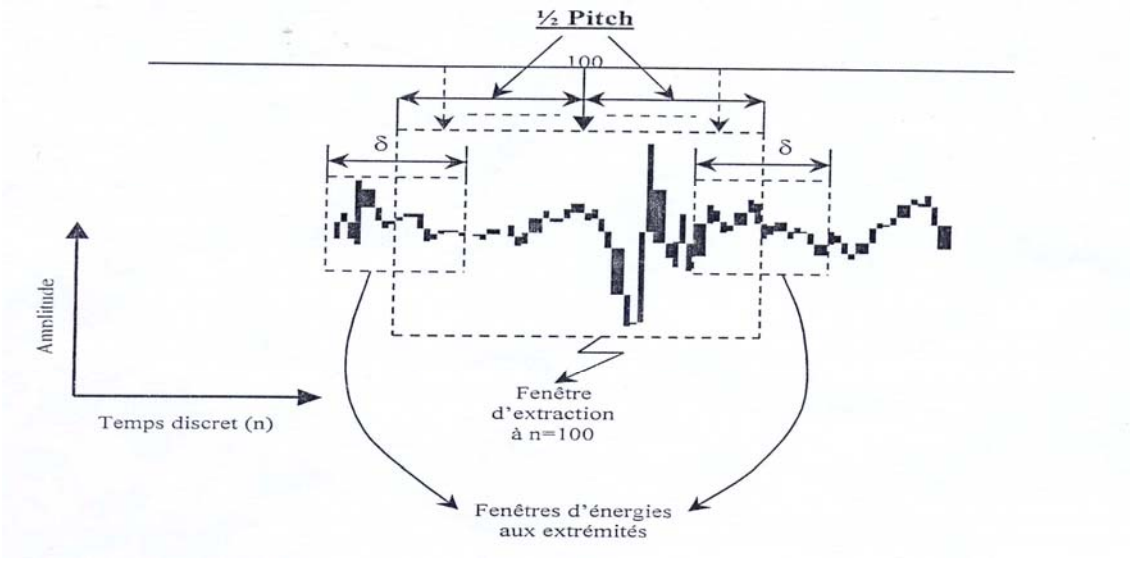


Fig. 2.9 La fenêtre d'extraction au point n=100. Ses deux fenêtres d'énergie aux extrémités sont illustrées clairement et sont de longueur S. La fenêtre d'extraction est de longueur égale à la période du pitch.

2.4.4.5 Alignement des CW

La procédure d'extraction dans le processeur **170** donne une description en DTFS pour chaque CW. En général, ces CW ne sont pas en phase, ceci dit, les caractéristiques principales dans les formes d'ondes ne sont pas alignées. Afin d'avoir une description précise des CW et de leur évolution dans la trame (comme celle illustrée dans la figure 2.5c), on doit établir un alignement de ces CW.

Dans le codeur WI, cet alignement est réalisé dans le processeur **170** à la fréquence des sous-trames. Plus précisément, cela se fait pour chaque deux CW successives (la CW courante et la CW précédente). Le processeur aligne la CW courante avec celle précédente en introduisant un décalage temporel circulaire à la trame courante. Puisque la représentation en DTFS nous permet de considérer la CW comme une seule période d'un signal périodique, ce décalage temporel circulaire est, en réalité, équivalent à l'adition d'une phase linéaire aux coefficients DTFS.

La figure 2.10 montre un schéma bloc du processeur d'alignement **170**. Pour faciliter la compréhension de ce schéma, on va séparer la discussion du processus d'alignement en trois scénarios différents. Dans le premier scénario, on va supposer que les deux CW sont de même longueur. On va, donc, discuter le critère d'alignement (processeur **173**) et l'opération de décalage dans le temps (processeur **174**). Le premier processeur détermine la longueur du décalage temporel nécessaire à la CW courante pour être alignée avec la précédente. Le deuxième décale la CW courante en introduisant le décalage circulaire calculé par le processeur **173** aux coefficients DTFS. Les processeurs **171** et **172** ne sont pas nécessaires dans ce scénario car les CW ont la même longueur.

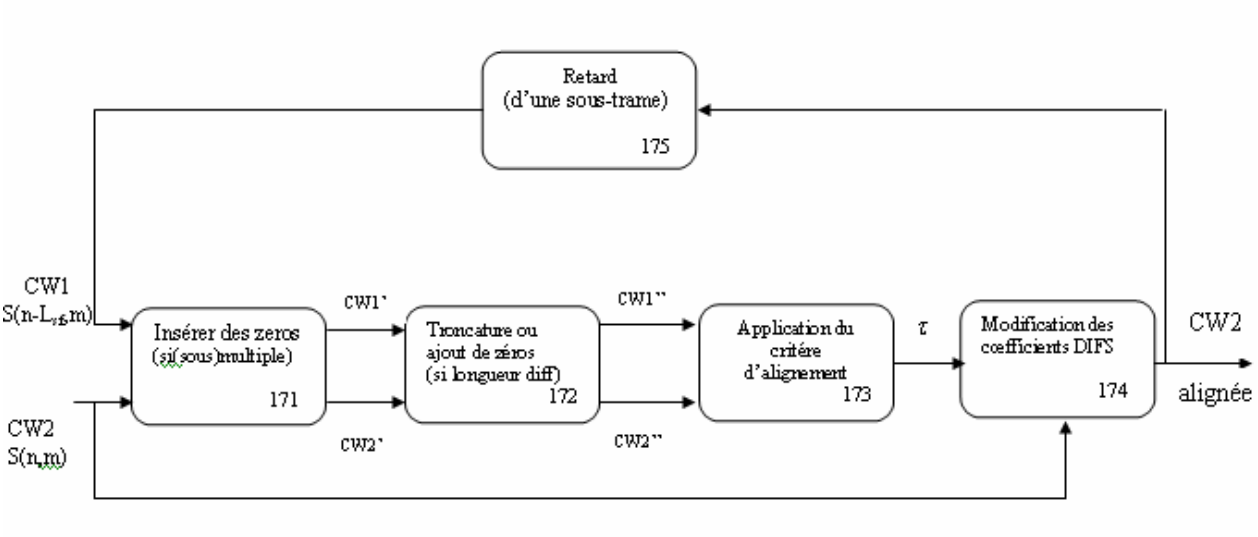


Fig. 2.10 Schéma bloc du processeur d'alignement 170

Dans le deuxième scénario, les deux CW sont supposées de longueurs différentes (sans que l'une soit multiple de l'autre). Au lieu de chercher une nouvelle version du processeur **173**. On ajoute le processeur **172** pour résoudre le problème de la différence en longueur.
Dans le dernier scénario, on étudiera un autre processeur qui traite le cas où la longueur de l'une est multiple de celle de l'autre.

Scénario 1 : Alignement avec même dimension

Commençons avec le premier scénario où les CW courante et précédente ont la même longueur. En utilisant (2.5), les représentations en DTFS de deux CW successives sont :

$$S(n_0, m) = \sum_{K=1}^{M}\left[Ak(n0)\cos\left(\frac{2\pi km}{P}\right) + Bk(n0)\sin\left(\frac{2\pi km}{P}\right)\right]$$

$$S(n_1, m) = \sum_{K=1}^{M}\left[Ak(n1)\cos\left(\frac{2\pi km}{P}\right) + Bk(n1)\sin\left(\frac{2\pi km}{P}\right)\right] \qquad (2.12)$$

où n_0 et n_1 sont les positions dans le temps des CW précédente et présente respectivement.
En plus, pour une meilleure commodité de notation,

$$\begin{aligned} &P = P(n) = P(n - 1) \\ &M = [\ P(n)/2] = [\ P(n - 1)/2] \end{aligned} \qquad (2.13)$$

P représente la longueur (pitch) des CW et M est le nombre d'harmoniques du spectre. Dans notre implémentation, puisque le processeur **170** travaille à la fréquence des sous-trames, n1 - no = Lsf = 20.
Supposons, maintenant, qu'un décalage circulaire de T échantillons est appliqué à la CW courante, s(n1, m) devient

$$S(n_1, m-T) = \sum_{K=1}^{M}\left[Ak(n1)\cos\left(\frac{2\pi k(m-T)}{P}\right) + Bk(n1)\sin\left(\frac{2\pi k(m-T)}{P}\right)\right] \qquad (2.14)$$

Il est clair que le décalage circulaire T dans le temps est équivalent à l'adition d'une phase linéaire $\frac{2\pi T}{P}$ dans le domaine DTFS. Pour trouver la valeur du décalage temporel T nécessaire à l'alignement de la CW1 avec la CW0, on utilise leur inter-corrélation comme suit :

$$T = \underset{0\le\tau'<2\pi}{\operatorname{argmax}} \sum_{k=1}^{M}\left\{ \left[Ak(n0)Ak(n1) + Ak(n0)Bk(n1)\right]\cos\left(\frac{2\pi kT}{P}\right) + \left[Bk(n0)Ak(n1) + Ak(n0)Bk(n1)\right]\sin\left(\frac{2\pi kT}{P}\right)\right\} \qquad (2.15)$$

Le terme de droite de (2.15) est l'inter-corrélation entre les deux CW exprimée en terme de coefficients DTFS. Cette équation peut être exprimée en terme du décalage temporel normalisé τ. En substituant

$$\tau = \frac{2\pi T}{P} \qquad (2.16)$$

Dans (2.15), on obtient :

$$T = \underset{0\le\tau'<2\pi}{\arg\max} \sum_{k=1}^{M} \{[Ak(n0)Ak(n1) + Ak(n0)Bk(n1)]\cos(k\tau') + [Bk(n0)Ak(n1) + Ak(n0)Bk(n1)]\sin(k\tau')\} \qquad (2.17)$$

Cette équation représente le critère d'alignement et forme la base du processeur **173**.

Un avantage immédiat de l'exécution de l'alignement dans le domaine DTFS est que cela permet un alignement fractionnel sans calcul additionnel tout en évitant les sur-échantillonnage et sous-échantillonnage conventionnels. Cet alignement fractionnel se fait à n'importe quelle résolution désirée (τ peut prendre toutes les valeurs réelles entre 0 et 2л). Une résolution de 1/4 d'un échantillon pour τ (pour une fréquence d'échantillonnage de 8000 Hz) donne de bons résultats.

La prochaine étape dans l'alignement est d'incorporer le décalage temporel r dans les coefficients DTFS de la CW courante $S(n_1, m)$. Cela se fait en développant les sinus et cosinus de (2.14) en utilisant les identités trigonométriques fondamentales. En regroupant les termes significatifs, on obtient un nouvel ensemble de DTFS :

$$\left.\begin{aligned} A'_k(n_1) &= A(n1)\cos\left(\frac{2\pi kT}{P}\right) - Bk(n1)\sin\left(\frac{2\pi kT}{P}\right) \\ B'_k(n_1) &= Ak(n1)\cos\left(\frac{2\pi kT}{P}\right) + Bk(n1)\sin\left(\frac{2\pi kT}{P}\right) \end{aligned}\right\} \quad \text{pour k=1, 2, …, M} \qquad (2.18)$$

D'où

$$S(n_1, m-T)\sum_{K=1}^{M}\left[\ \mathrm{A'k}\cos\left(\frac{2\pi kT}{P}\right) + B'k\sin\left(\frac{2\pi kT}{P}\right)\right] \qquad (2.19)$$

$\{A'k(n1)\}$ et $\{B'k(n1)\}$ sont les nouveaux coefficients DTFS de la CW décalée de T échantillons à droite. L'équation (2.18) peut être exprimée en terme du décalage temporel normalisé τ en utilisant (2.16) :

$$\left.\begin{aligned} A'k(n_1) &= Ak(n1)\cos(k\tau) - Bk(n1)\sin(k\tau) \\ B'k(n_1) &= Ak(n1)\cos(k\tau) + Bk(n1)\sin(k\tau) \end{aligned}\right\} \quad \text{pour k= 1, 2, …, M} \qquad (2.20)$$

En résumé, le processeur 173 utilise (2.17) pour trouver le τ optimal et le processeur **174** utilise, alors, (2.20) pour incorporer τ dans les coefficients DTFS.

Scénario 2 : Alignement avec dimensions différentes

Dans le premier scénario, on a supposé que les deux CW ont la même longueur, ce qui, en général, n'est pas le cas. En d'autres termes, le critère d'alignement (2.17), qui est basé sur cette supposition d'égalité de dimension, n'est plus applicable directement. Pour éviter de calculer un nouveau critère d'alignement, on dédie le processeur **172** pour un pré -traitement des CW en appliquant une des deux opérations suivantes afin d'égaliser leur dimensions avant de passer au critère d'alignement :

- dans le domaine fréquentiel, on tronque la CW la plus longue jusqu'à ce qu'elle ait la même longueur que l'autre.
- dans le domaine fréquentiel, on remplit de zéros la plus courte CW jusqu'à ce qu'elle ait la même longueur que l'autre.

Dans la première approche, abandonner les harmoniques de haute fréquence aura pour effet de rétrécir la CW dans le temps. Bien que la CW peut perdre quelques détails temporels dans ce processus, les harmoniques à l'extrémité haute fréquence du spectre tendent à avoir relativement une faible énergie. Par conséquent, la forme de la CW tronquée se rapproche, généralement, très bien la forme originale.

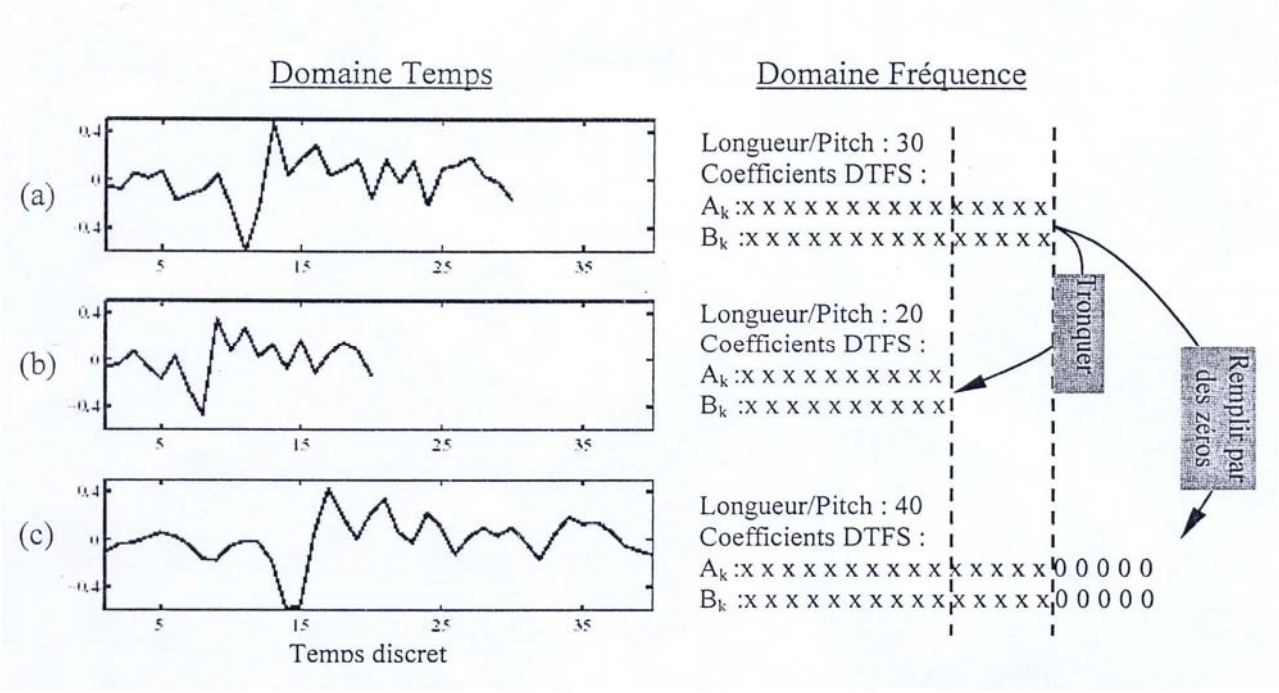


Fig. 2.11 Échelonnage temporel des CW. **(a)** Une CW de longueur 30 échantillons et ses coefficients DTFS correspondants 15 {Ak} + 15 {Bk}. **(b)** La version tronquée de longueur 20 échantillons. La forme temporelle globale est préservée mais les détails sont plus ou moins perdus. **(c)** La version étirée après avoir ajouté des zéros aux coefficients DTFS. Cet étirement n'introduit aucune information nouvelle à la séquence temporelle, mais il offre une meilleure résolution.

Dans la seconde option, le remplissage par des zéros dans le domaine spectral provoque un allongement temporel de la CW pour qu'elle ait la même longueur que la CW précédente. Cette opération, équivalente à une interpolation à bande limitée dans le domaine temporel, n'introduit aucune information temporelle nouvelle à la séquence, mais elle offre une résolution plus élevée. La figure 2.11 montre l'exemple d'une CW contractée et étirée dans le temps.

Scénario 3 : Alignement avec longueur (sous-)multiple du pitch

Comme on l'a déjà mentionné au paragraphe 2.4.4.3, le pitch peut, occasionnellement, doubler, tripler ou diminuer de moitié dans la parole naturelle. Donc, des périodes de pitch multiples ou sous-multiples peuvent apparaître dans une CW extraite. Afin d'éviter les complications dans

l'alignement, la plus courte CW est dupliquée un nombre entier de fois dans le processeur **171** de manière à ce que sa longueur atteigne celle de la plus longue CW. Dans le domaine fréquentiel, ceci est équivalent à l'insertion d'harmoniques d'amplitude nulle entre les harmoniques de la plus courte CW. La figure 2.12 montre comment les zéros sont insérés entre les coefficients DTFS $\{A_k, B_k\}$ et le résultat correspondant dans le domaine temporel.

Pour détecter l'apparition de (sous-) multiple du pitch, on opère de la même manière que celle du paragraphe 2.4.4.3 en utilisant l'indicateur C. Si cet indicateur est différent de l'unité, alors, il y a eu division ou multiplication du pitch. C = 2, signifie que la valeur du pitch a doublé et on insère un zéro entre chaque deux coefficients DTFS adjacents pour que la CW soit dupliquée une fois (Fig. 2.12b). C = 3, signifie que le pitch a triplé et on insère, alors, deux zéros entre chaque deux coefficients DTFS adjacents de la plus courte CW pour qu'elle soit dupliquée deux fois (Fig. 2.12c). On procède de la même manière pour les autres multiples.

Remarques sur le processus d'alignement

L'alignement est réalisé entre deux CW successives. Pour aligner entre deux trames successives, on applique la même règle entre la première CW de la trame courante et la dernière CW de la trame passée.

Dans la figure 2.11b, il est clair que la puissance du signal (énergie par échantillon) diminue après l'avoir tronqué dans le domaine spectral. Par contre, le remplissage par zéros et l'insertion de zéros (Figures 2.11c et 2.12) préserve la puissance du signal.

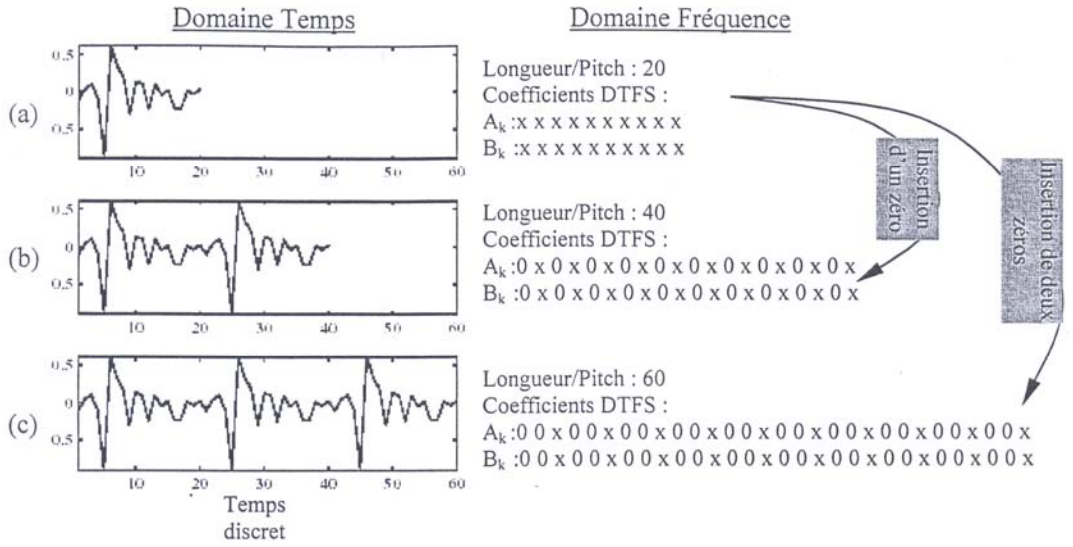


Fig. 2.12 Illustration de l'insertion de zéros entre les composantes spectrales. **(a)** Une CW de longueur 20 échantillons : 10 $\{A_k\}$ +10 $\{B_k\}$. **(b)** La forme d'onde de (a) est dupliquée une fois après insertion d'un zéro entre deux harmoniques adjacentes. (c) La forme d'onde de (a) est dupliquée deux fois après insertion de deux zéros entre chaque deux harmoniques adjacentes.

Une simple évaluation du critère d'alignement (2.17) peut être très coûteuse du point de vue calcul, particulièrement pour les CW longues. Par exemple, si $S(n_0,.)$ et $S(n_1,.)$ sont de longueur 90 échantillons, on aura 90 x 4 = 360 inter - corrélations à calculer (en supposant que la résolution de l'alignement est de 1/4 échantillon). Chacune de ces inter - corrélations nécessite au moins 90 x 2 = 180 multiplications selon (2.17). Ainsi, le coût total du calcul nécessaire au critère d'alignement tout seul est d'environ 360 x 180 = 28800 multiplications.

Les CW (comme on l'a déjà mentionné au § 2.4.4.4) sont extraites de manière à éviter une grande énergie aux extrémités ; cependant, vu la nature du décalage circulaire, le processus d'alignement peut engendrer des CW à énergie élevée aux extrémités. Toutefois, cela ne causera aucune discontinuité dans la parole reconstituée puisque les CW ont été étendues périodiquement avant l'alignement.

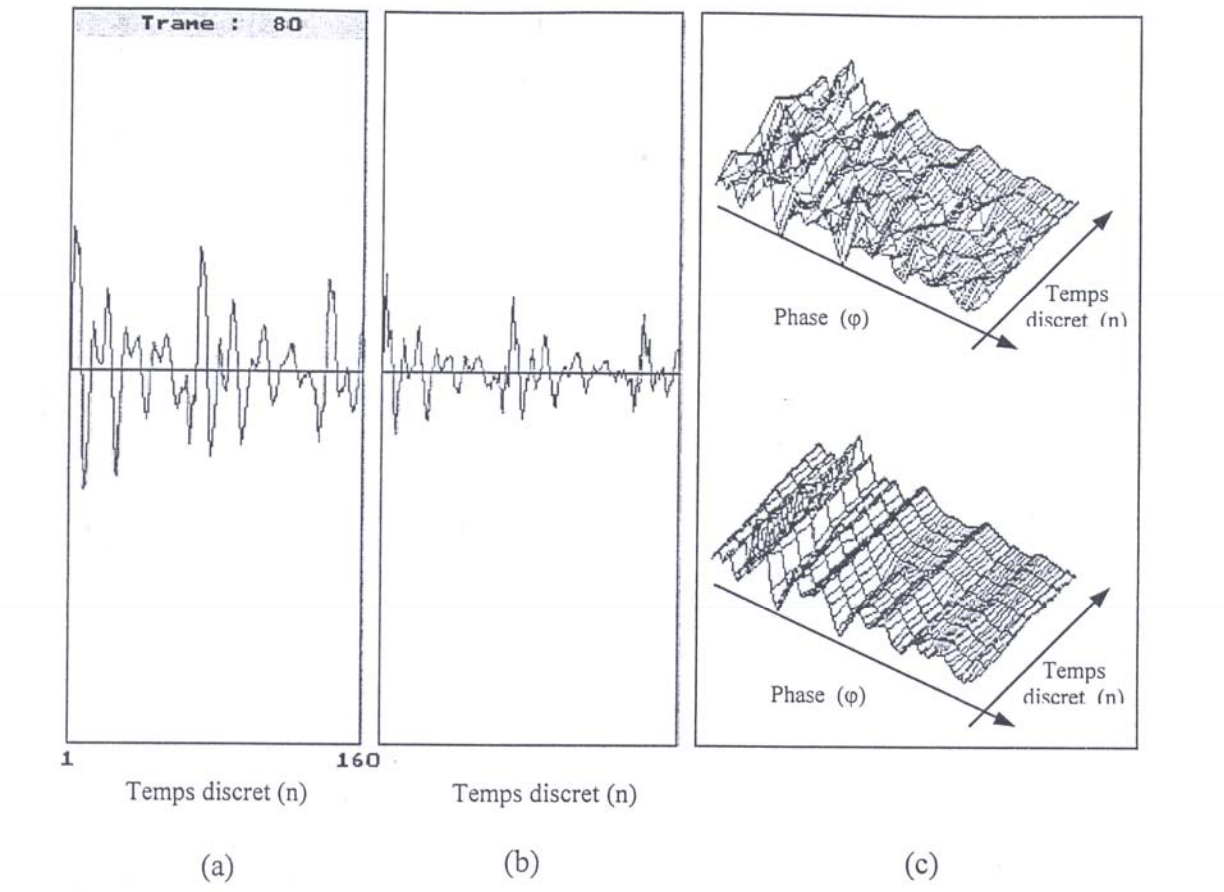


Fig. 2.13 Illustration de l'opération d'alignement.
Une trame (20 ms) de parole d'une voix masculine (fichier m28b.wav).
La trame du signal résiduel correspondant.
En haut : Extraction et formation de la surface
d'évolution de 8 CW. En bas : La surface d'évolution
des CW après alignement. Le pitch vaut 68 échantillons.

Conclusion :

En ce chapitre nous avons révisé le codeur existant de WI, initialement mis en application par Choy [13] basé sur les travaux de Kleijn [22]. Notre souci primaire est l'extraction des formes d'ondes caractéristiques. Le prochain chapitre fournit toutes les étapes que nous avons implémenté pour arriver à notre objectif principal ainsi que tous les résultats intermédiaires.

Chapitre III

Chapitre 3

Résultats et Interprétations

Introduction

On a simulé le codeur WI (jusqu'à l'extraction des CW et leurs alignement) en utilisant le langage C pour la partie programmation et Matlab pour les représentations.

Dans cette simulation, on a tenu compte de l'importante de l'extraction des formes d'ondes caractéristiques dans le codage de la parole par interpolation de formes d'ondes. C'est dans cette optique qu'on c'est intéressé essentiellement à approfondir nos tests et simulations sur ce bloc du codeur.

Notre travail consiste à effectuer les opérations suivantes :

- Réaliser l'analyse LP : qui comprend l'extraction des coefficients du filtre inverse $\{a_i\}$, ces coefficients sont convertis en coefficients LSP (utilisés dans la quantification). A partir des coefficients $\{a_i\}$ trouvés auparavant et du signal parole on construit le signal résiduel.

- Estimation du pitch une fois par trame à partir du signal résiduel, ces valeurs du pitch sont interpolées à l'aide des équations (2.9), (2.10) et (2.11) pour avoir un pitch à chaque instant de la trame.

- Finalement, l'extraction des formes d'ondes caractéristiques s'effectue en utilisant le signal résiduel et le pitch interpolé. Dans notre implémentation, les formes d'onde caractéristiques sont extraites chaque 2.5ms, c'est-à-dire (8 CW / trame).

Toutes les étapes citées ci-dessus sont représentées dans le schéma bloc 3.1

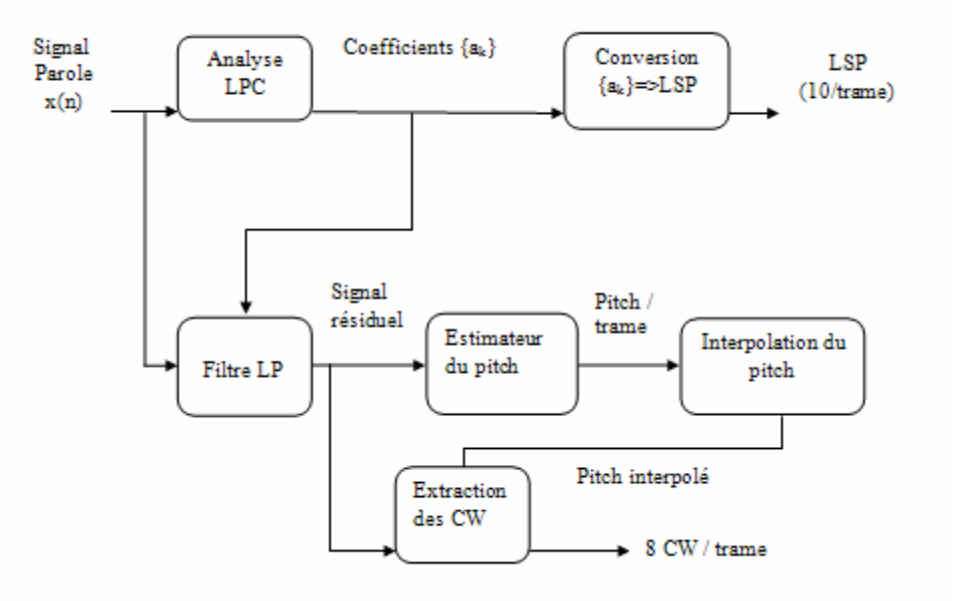


Fig.3.1 Schéma bloc de notre simulation.

Remarques :

- Pour la partie programmation on a utilisé le langage C (Builder C++ 6.0), Matlab et Cool-Edit pour les représentations.

- On suppose que la parole à l'entrée est dans un format numérique (16 bits par échantillon) avec une fréquence d'échantillonnage de 8 kHz. La taille de la trame L_f est de 160 échantillons (20 ms) et la longueur de la sous-trame L_{sf} est de 20 échantillons.

- On travaille avec un fichier wave (original1.wav) de base de donnée TIMIT [].

- Toutes les opérations précédentes sont effectuées sur chaque trame, d'où la fréquence de mise à jour des coefficients LP vaut 50 Hz dans notre implémentation.

3.1 L'analyse LP

L'analyse LPC est le procédé qui consiste à partir d'un signal parole non codé ni compressé; simplement échantillonné à une fréquence donnée (8 khz); d'obtenir les paramètres du modèle LPC. Il faut donc déterminer, au cours de l'analyse LPC, les coefficients du filtre d'analyse LP, les LSP et le signal résiduel.

3.1.1 Détermination des coefficients du filtre inverse

Le signal parole est tout d'abord fenêtré par la fenêtre de Hamming puis corrélé et enfin les coefficients du filtre LP sont trouvé à l'aide de l'algorithme de Levinson-Durbin (Annexe A).

Notre signal parole contient 1430 trames, et chaque trame est sur 160 échantillons.

On représente toutes les opérations précédentes dans une trame de 240 échantillons pour mieux voir et interpréter les résultats obtenus.

Le signal parole original avant et après fenêtrage ainsi que la fenêtre de Hamming sont représentés dans les figures 3.2, 3.3 et 3.4 respectivement.

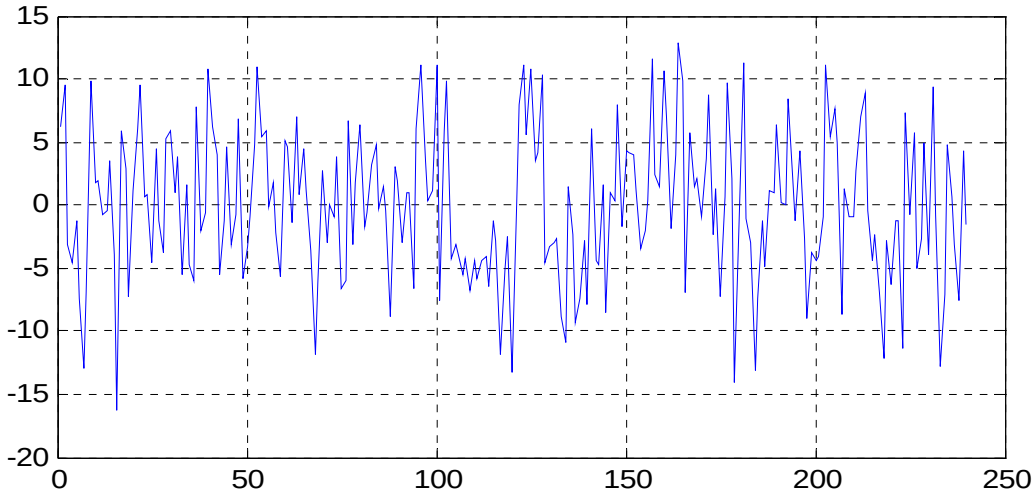


Fig.3.2 Signal parole original avant fenêtrage (240 échantillons)**.**

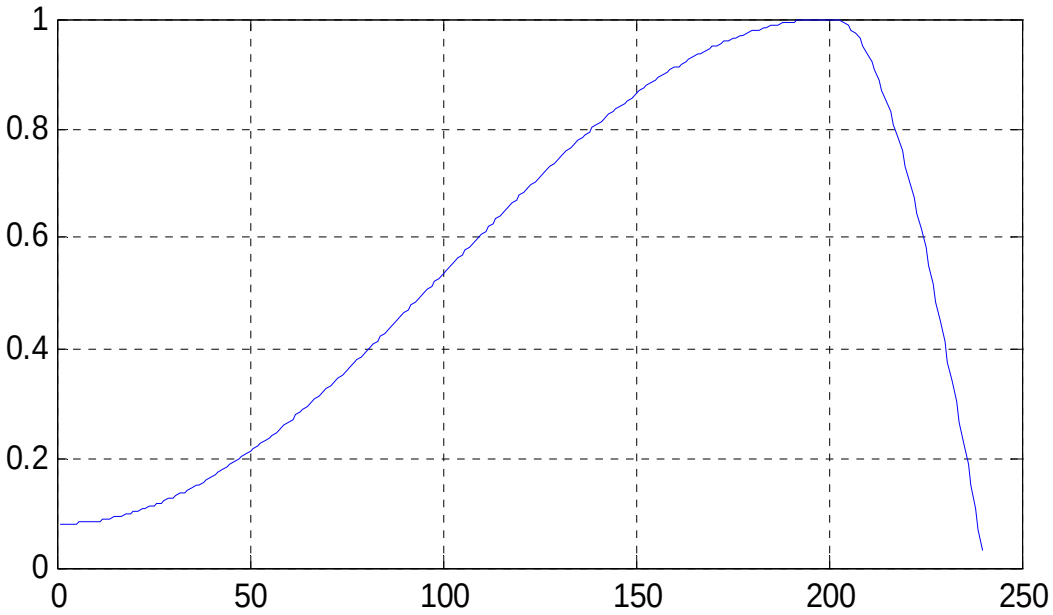


Fig.3.3 Fenêtre de Hamming

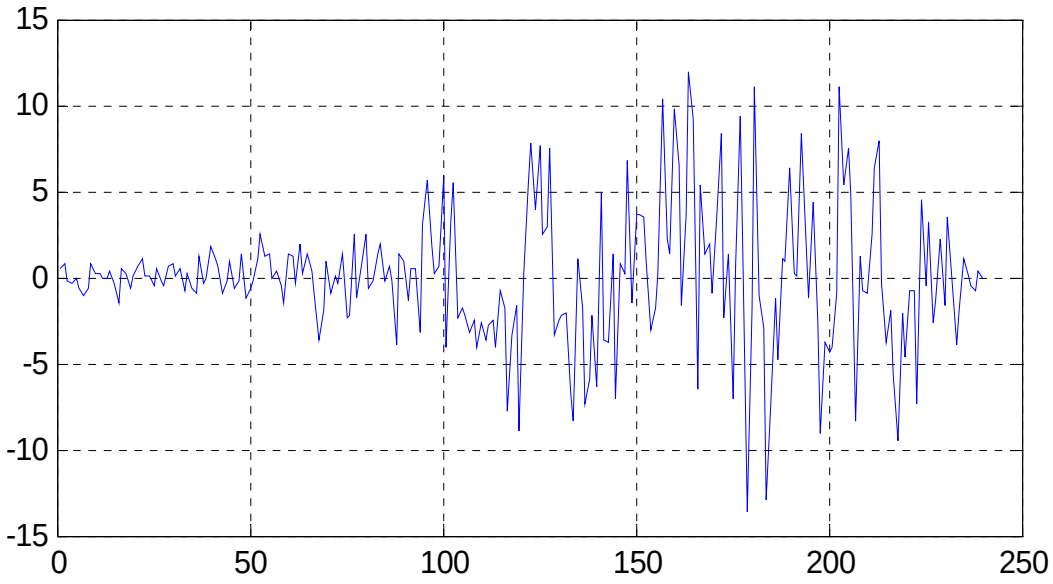


Fig.3.4 Signal parole après fenêtrage de Hamming.

On a utilisé la fonction de Hamming pour éliminer les effets de bord, c'est-à-dire éviter les chevauchements, ce choix sera justifié par la suite.

La corrélation entre un signal et un autre mesure en quelques sortes l'énergie mutuelle contenue dans ces deux signaux selon le décalage de l'un par rapport à l'autre. Plus cette énergie est grande plus les signaux se ressemblent, plus l'énergie du produit est grande. Dans notre cas, nous pratiquons une corrélation du signal avec lui-même, c'est-à-dire l'autocorrélation, voir paragraphe 3.6.1.

Dans le calcul de la fonction d'autocorrélation il faut néanmoins prendre en compte le fait que le signal traité n'est pas infini et qu'en plus il n'est traité que par tranches de 240 échantillons environ. Une technique que nous avons essayée consiste à prendre une fenêtre de 240 échantillons et à la corréler avec le signal lui-même.

Le signal est considéré seulement sur une plage temporelle limitée. Pour diminuer l'influence du fenêtrage lors du calcul du spectre du signal, en particulier en très basse fréquence, nous avons choisi une fenêtre de Hamming qui a pour effet de diminuer l'erreur introduite de cette manière car les échantillons au bord de la fenêtre d'analyse sont très atténués par la fenêtre. Cela justifie encore à posteriori la fenêtre choisie.

Le calcul de l'autocorrélation du signal se fait à partir de l'équation 1.11 et le calcul lié à la dérivation pour trouver les coefficients se réduit à 1.15.

Ce calcule peut s'exprimer sous la forme d'un calcul matriciel (matrice de Toeplitz 3.16) c'est-à-dire que la matrice est symétrique et tous les éléments de la diagonale sont égaux. Cette matrice s'inverse efficacement à l'aide de l'algorithme de Levinson-Durbin. On obtient ainsi les coefficients $\{a_i\}$.

La figure 3.5 montre les coefficients de corrélation de la trame numéro 7 du signal parole original et la figure 3.6 donne les coefficients $\{a_i\}$ de cette trame.

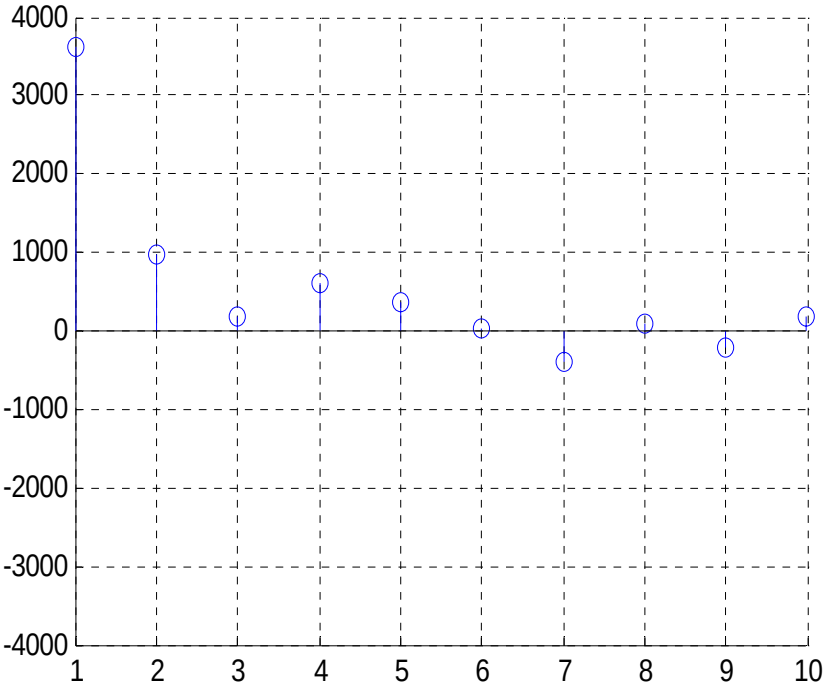


Fig.3.5 Les coefficients de corrélation (10/trame) de la trame numéro 7.

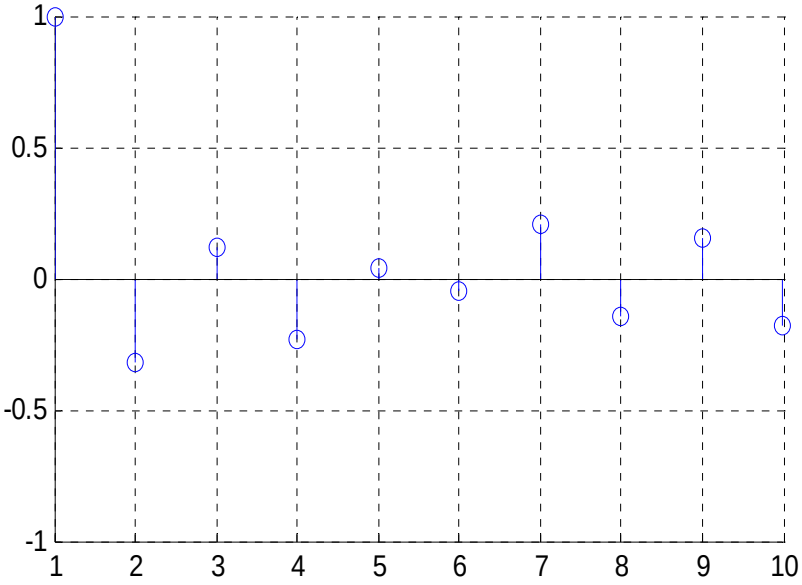


Fig.3.6 Les coefficients $\{a_i\}$ de la trame numéro 7.

Cet histogramme nous donne le nombre d'occurrence des coefficients $\{a_i\}$ pour chaque colonne, c'est-à-dire le nombre de valeurs des $\{a_i\}$ qui se répètent pour les 10 colonnes.
On remarque que la plupart des valeurs sont entre -1 et 1 ce qui ne donne pas une grande stabilité du filtre LP.

3.1.2 Conversion des coefficients $\{a_i\}$ en LSP

Afin d'avoir une stabilisation optimale du filtre LP, on utilise les coefficients LSP (paires de raies spectrales) qui offrent de meilleurs propriétés de quantification et d'interpolation.

Après calcul et représentation des coefficients LSP (figures 3.8, 3.9 et 3.10 respectivement), on a constaté qu'ils se trouvent bien sur le cercle unité et sont entrelacés (module égale a 1 et phase variable -figures 3.8 -) ce qui prouve bien la stabilité du filtre LP et limite le codage de la phase seulement.

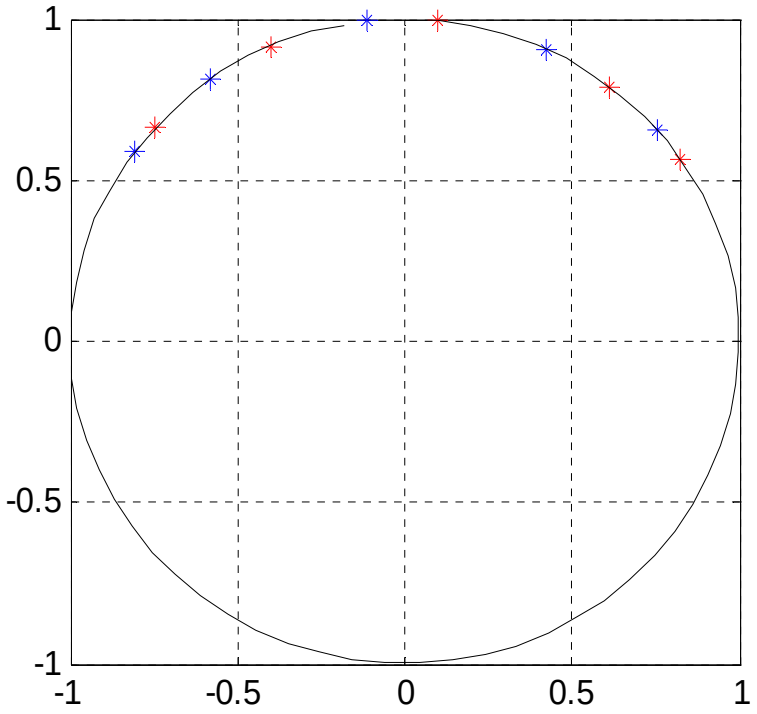


Fig.3.8 Représentation des LSP sur le cercle unité (10 / trame).

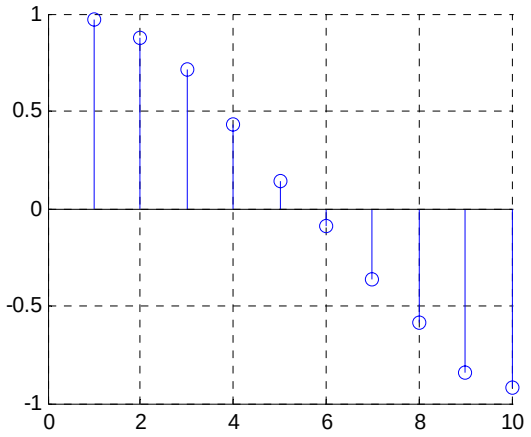


Fig.3.9 Les coefficients LSP (10 /trame) de la trame numéro 7.

L'histogramme des LSP représenté dans la figure 3.10.

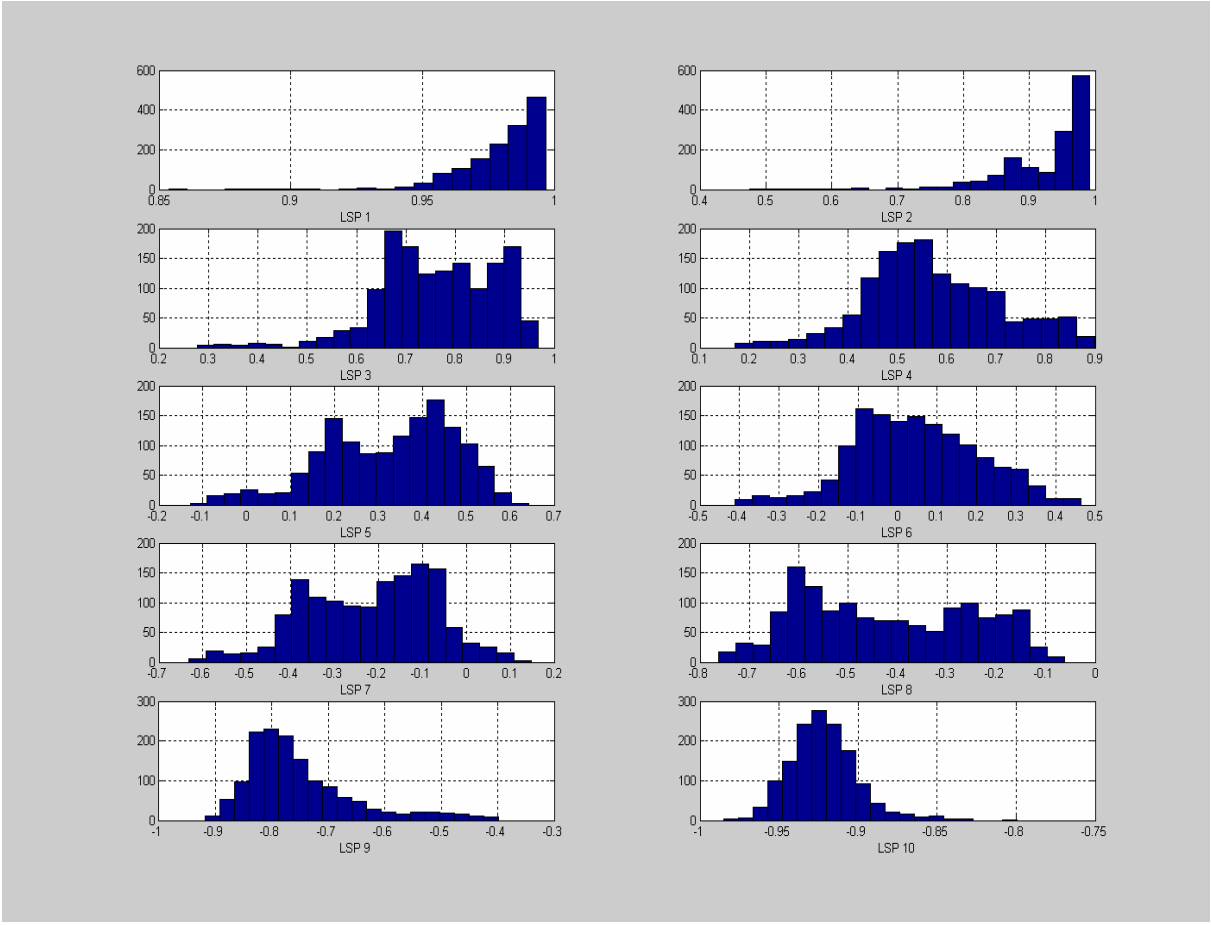


Fig.3.10 Histogramme des coefficients LSP

3.1.3 Extraction du signal résiduel

Comme on l'a déjà vu au paragraphe 1.6, la fonction principale du filtre d'analyse LP est d'extraire le signal résiduel du signal parole comme le montre la figure 3.11.

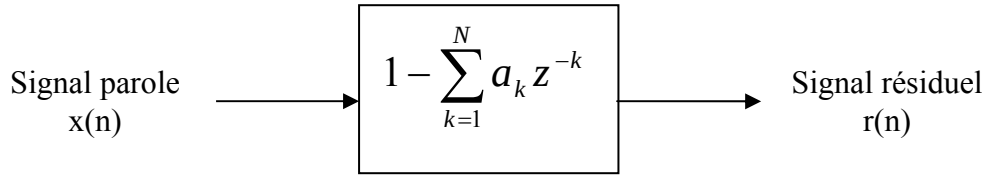


Fig.3.11 Filtre inverse

Le signal parole original et le signal résiduel sont donnés par la figure 3.12 et 3.13 respectivement.

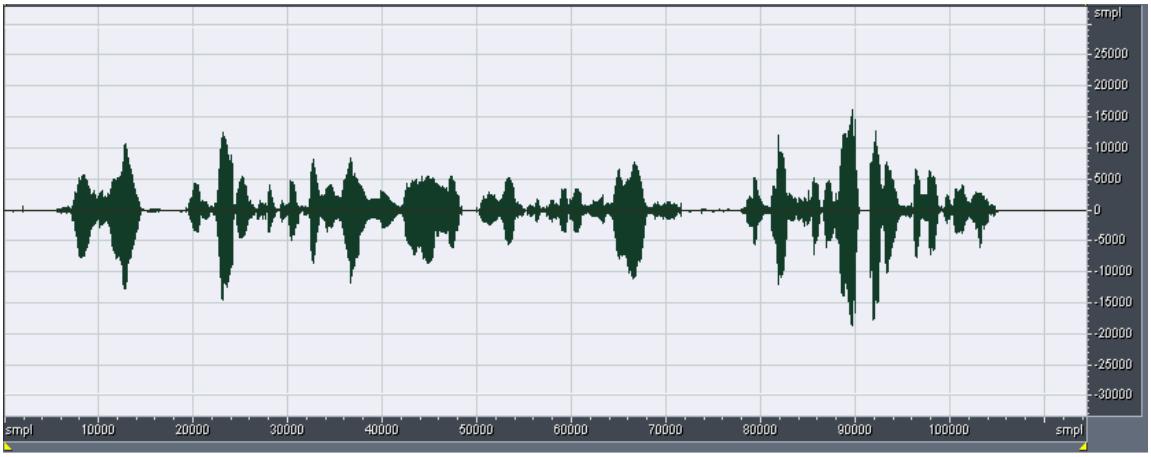


Fig.3.12 Signal parole (original1.wav)

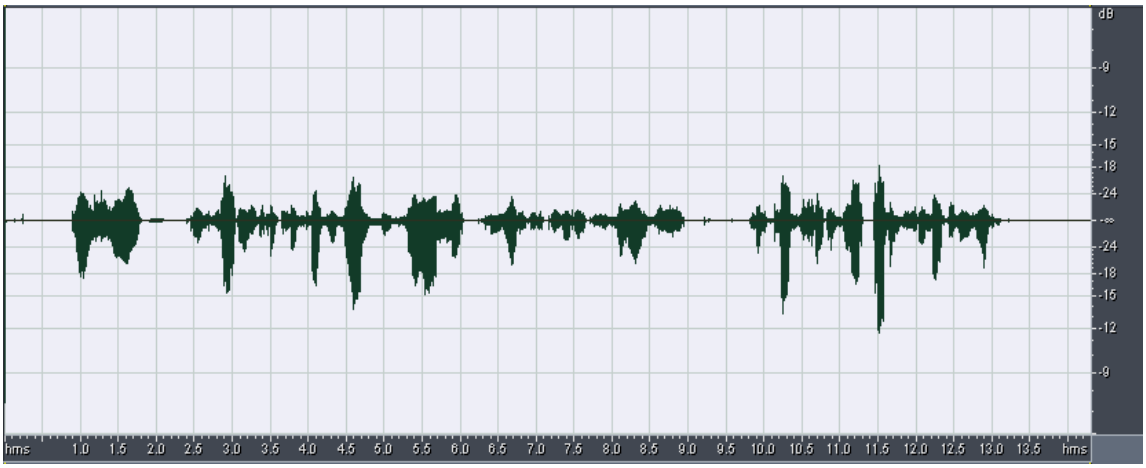


Fig.3.13 Signal résiduel correspondant.

Ce signal résiduel est identique au signal parole original avec une énergie plus faible.

3.2 Estimation du pitch

Le pitch représente la fréquence fondamentale du signal résiduel, estimé une fois par trame.
Comme déjà indiqué au paragraphe 2.4.2, les valeurs du pitch décrites dans l'équation 2.8 sont des valeurs entières. En effet, des valeurs entières du pitch (avec une résolution de 1 échantillon pour une fréquence de 8 kHz) sont suffisantes pour l'implémentation de notre codeur WI. Les valeurs minimales et maximales de la période du pitch dans notre implémentation sont égales à 20 et 120 respectivement. On pouvait étendre cet intervalle de 20 à 147 puisque, de toute manière, on alloue 7 bits pour quantifier le pitch ($147-20+1=2^7$).

Dans cette implémentation, on adopte l'algorithme tiré du EVRC (Enhanced Variable Rate Codec) [18] qui appartient à la seconde catégorie. On donne une brève description de cet algorithme :

Si ($\beta_o > \beta_1 + 0.4$)
{
si ($|d_0 - d_1| > 15$)
$d_{opt} = d_o$
sinon
$d_{opt} = [(d_0 + d_1)/2.0]$
}
sinon
$d_{opt} = d_1$

où β_0 et β_1 sont des fonctions de confiance qui indiquent le degré de fiabilité des pitchs estimés (do et d1).

La figure 3.14 nous donne les résultats obtenus pour le pitch estimé à partir du signal résiduel.

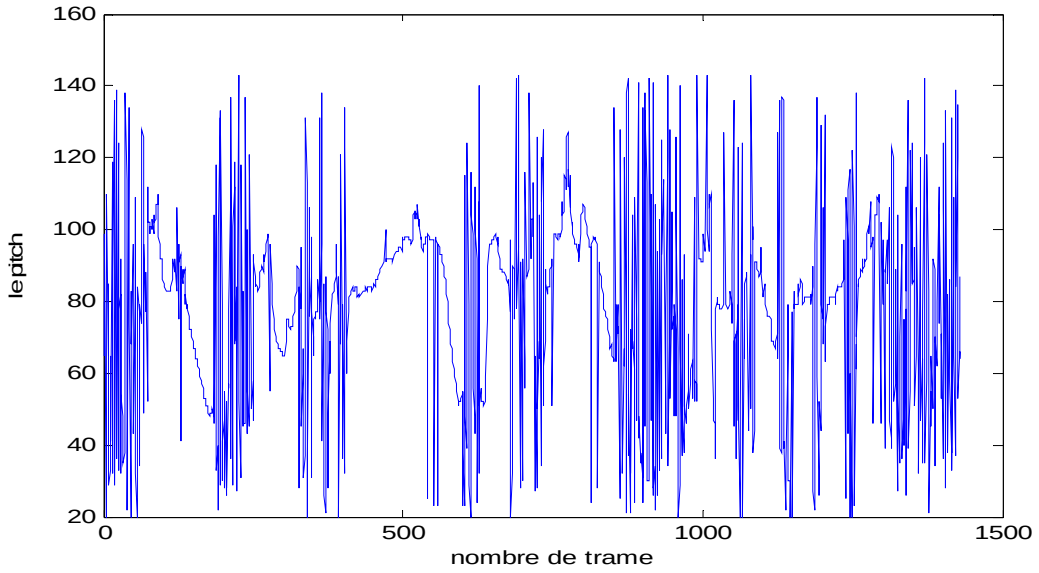


Fig.3.14 Estimation du pitch.

3.3 Interpolation du pitch

Afin d'avoir une meilleure précision dans l'extraction des formes d'ondes caractéristiques, on effectue l'opération d'interpolation du pitch qui nous donne une valeur a chaque point d'extraction.

L'interpolation est réalisée grâce aux équations (2.9), (2.10) et (2.11).

Remarque :

On a utilisé dans notre implémentation le pitch estimé pour une trame car on a remarqué qu'il n'y a pas une grande variation entre le pitch estimé et les pitchs interpolés afin de faciliter la simulation.

3.4 Extraction des formes d'ondes caractéristiques

L'extraction des formes d'ondes se fait à partir du signal résiduel et du pitch estimé comme mentionné dans le paragraphe 2.4.4.

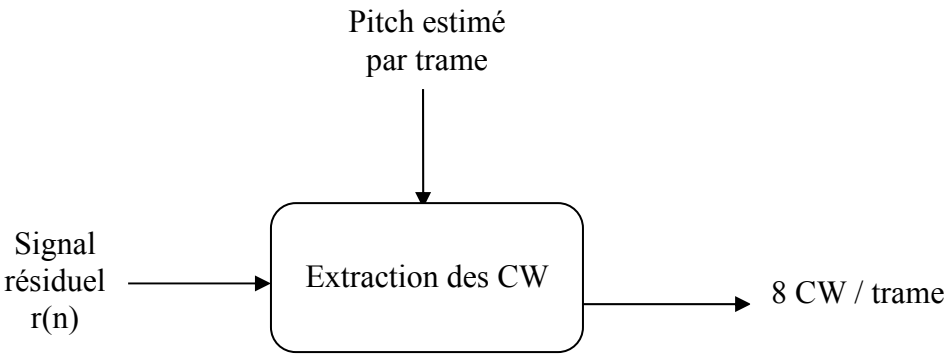


Fig.3.15. Schéma bloc de l'extraction de formes d'ondes caractéristiques.

La figure 3.16 montre le signal résiduel original avec les points d'extraction situés à n=20, 40, …, 160.

Les fenêtres d'extraction peuvent sortir en dehors des extrémités de la trame, d'où la nécessité d'avoir un certain nombre d'échantillons passés et futurs. Puisque la plus grande longueur d'une CW est P_{max}, le nombre d'échantillons passés nécessaires doit être au moins égal à $P_{max} / 2 = 73$ (figure 3.17). Même chose pour le nombre d'échantillons futurs.

Les fenêtres d'extraction successives se recouvrent presque tout le temps. En d'autres termes, deux CW adjacentes peuvent partager les mêmes segments du signal résiduel. Plus encore, puisque chaque point d'extraction peut avoir un déplacement de ε (entre –16 et 16), deux CW adjacentes peuvent même être identiques. Un exemple d'un tel cas est donné dans la figure 3.18a et 3.18b où les CW extraites aux points n = 100 et n = 120 sont les mêmes. Même chose pour n = 20 et n = 40.

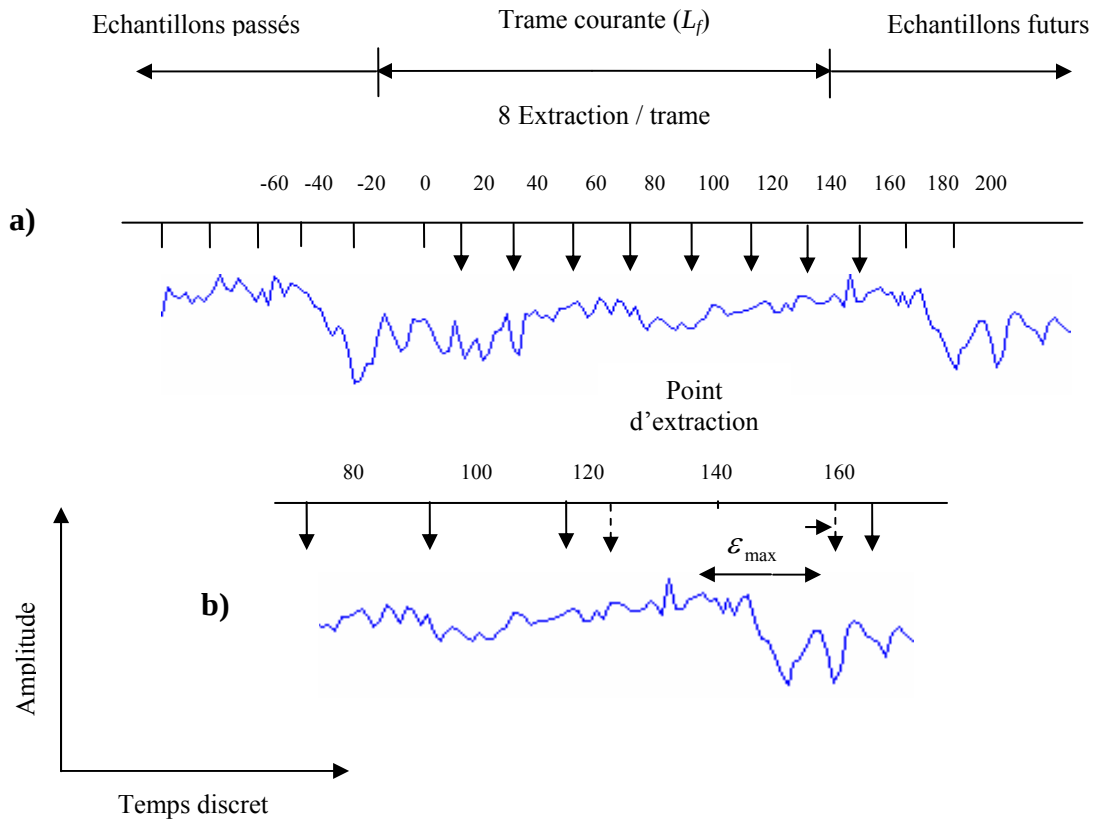


Fig.3. 16 Exemple d'un point d'extraction libre. **(a)** Les positions originales des points d'extractions des 8 CW. Chaque point d'extraction peut être déplacé légèrement jusqu'à ce que les extrémités de la fenêtre d'extraction soient dans des régions de faible énergie. **(b)** Illustration détaillée pour le point d'extraction à n=140.

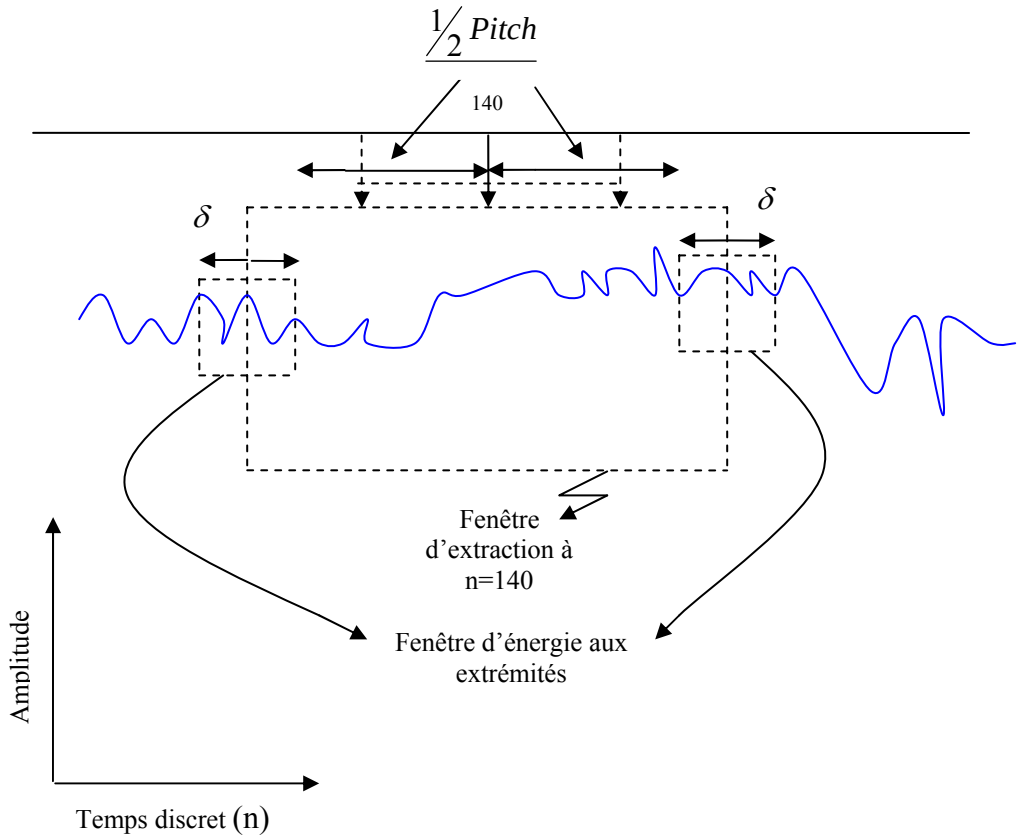


Fig. 3.17 La fenêtre d'extraction au point n=140. Les deux fenêtres d'énergie aux extrémités sont illustrées clairement et sont de longueur δ . La fenêtre d'extraction est de longueur égale à la période du pitch.

Pour la parole voisée, chaque CW extraite peut être considérée comme une période individuelle du pitch. Pour la parole non voisée, les CW sont assimilables à des segments de bruit de longueurs variables.

Dans notre implémentation, la taille d'une trame est de 160 échantillons. Puisqu'on a 8 extractions dans chaque trame et chaque extraction contient au minimum 20 échantillons (P_{min}), alors, chaque échantillon dans une trame appartient au moins à une CW si ε est à zéro.

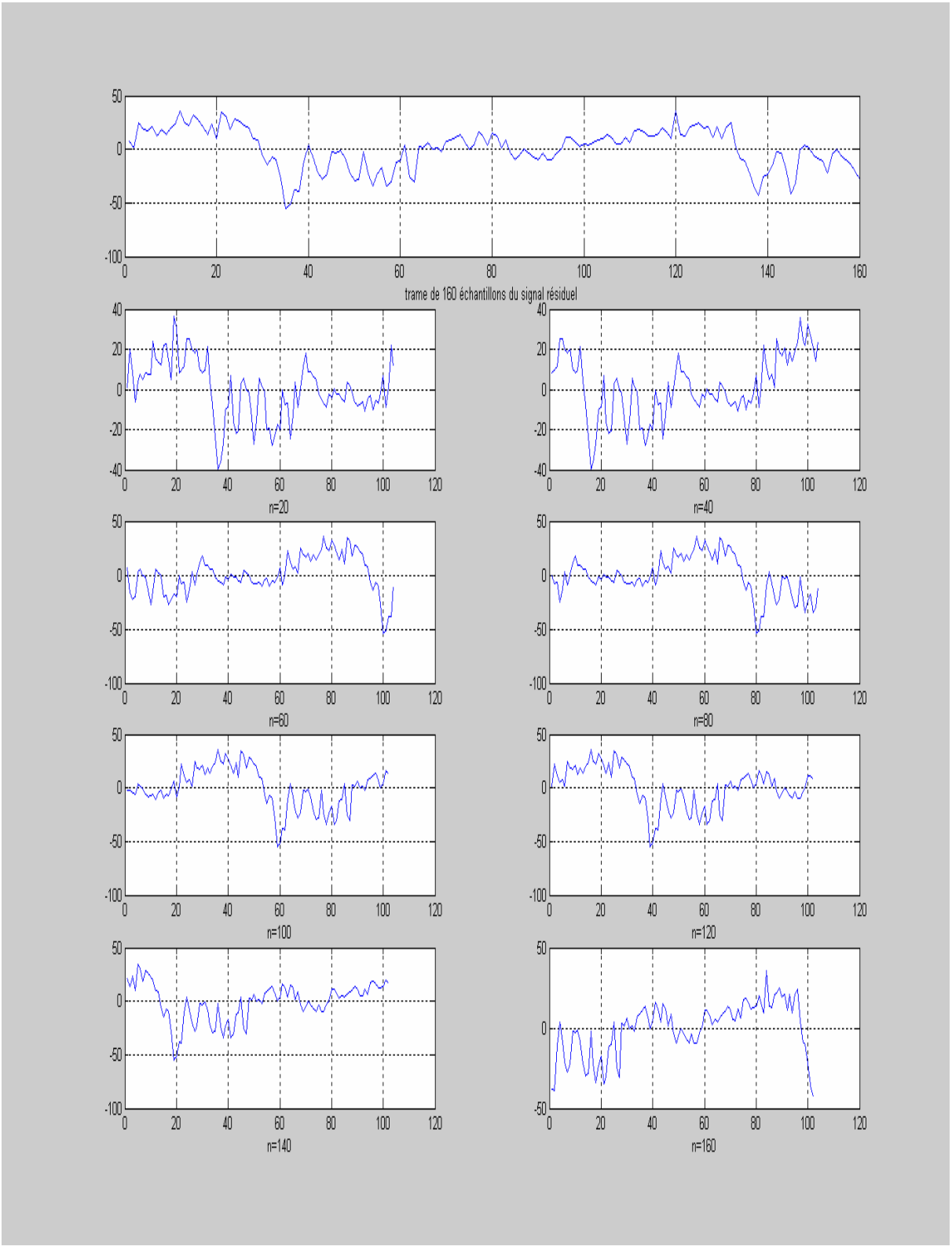


Fig.3.18a 8 CW extraites à partir d'une trame du signal résiduel (signal original d'une femme).

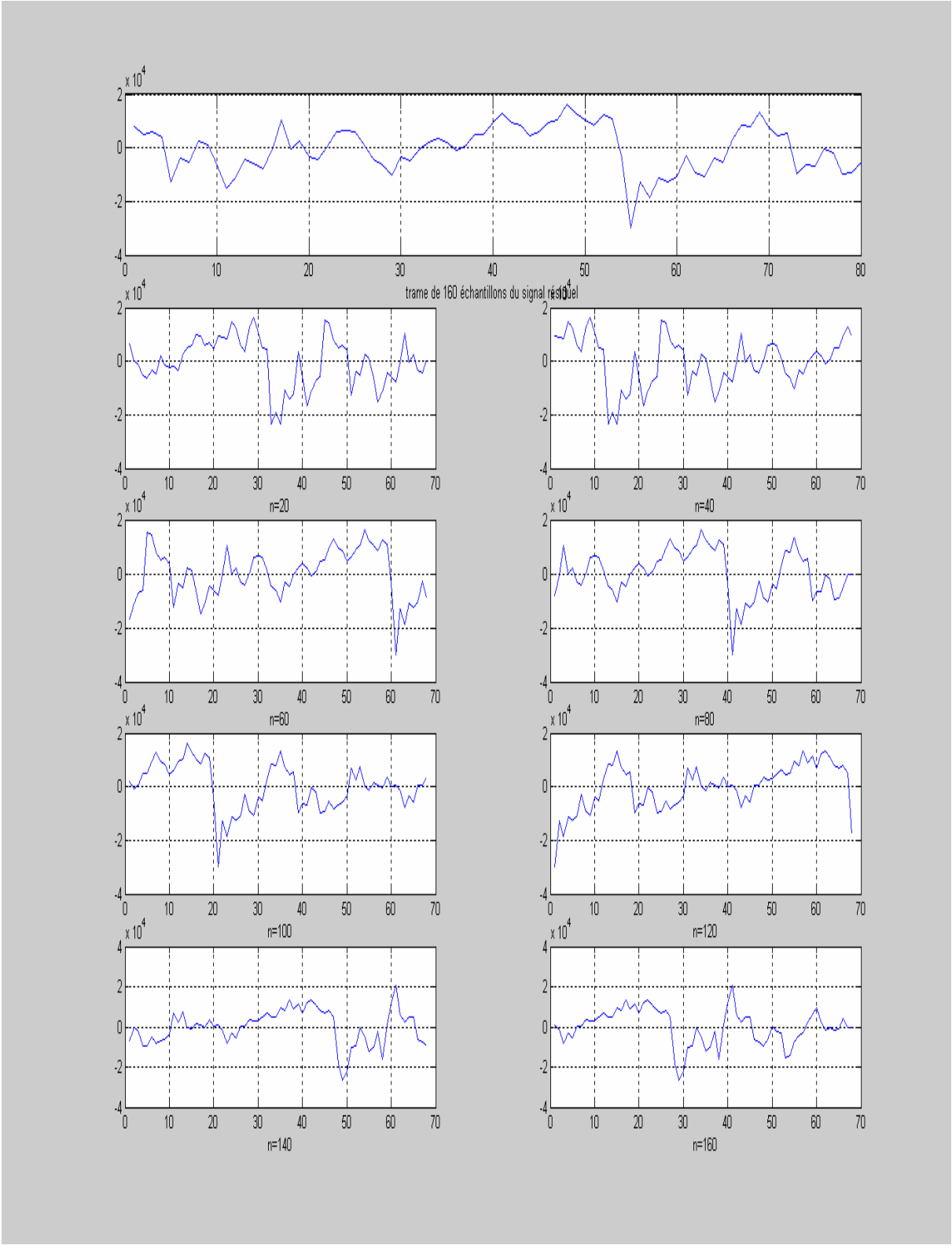


Fig.3.18b 8 CW extraites à partir d'une trame du signal résiduel (signal original d'un homme).

On remarque des graphes 3.18a et 3.18b que les formes d'ondes extraites de la parole d'une voix féminine sont caractérisées par une fréquence plus élevée par rapport à la voix masculine. En conséquence les CW extraites de la voix féminine apporteront plus d'informations. On peut aussi d'après le graphe de l'extraction savoir si c'est un homme ou une femme qui parle.
La procédure d'extraction dans le processeur donne une description en DTFS (Discrete Time Fourier Series) pour chaque CW. En général, ces CW ne sont pas en phase, ceci dit, les caractéristiques principales dans les formes d'ondes ne sont pas alignées. Afin d'avoir une description précise des CW et de leur évolution dans la trame, on doit établir un alignement de ces CW.

Dans notre implémentation, cet alignement est réalisé dans le processeur à la fréquence des sous-trames. Plus précisément, cela se fait pour chaque deux CW successives (la CW courante et la CW précédente). Le processeur aligne la CW courante avec celle précédente en introduisant un décalage temporel circulaire à la trame courante. Puisque la représentation en DTFS nous permet de considérer la CW comme une seule période d'un signal périodique, ce décalage temporel circulaire est, en réalité, équivalent à l'adition d'une phase linéaire aux coefficients DTFS.

La figure 3.19 et 3.20 et 3.21 montre une surface des CW non alignée extraite d'une seule trame, une surface alignée de ces CW et leur signal résiduel correspondant.

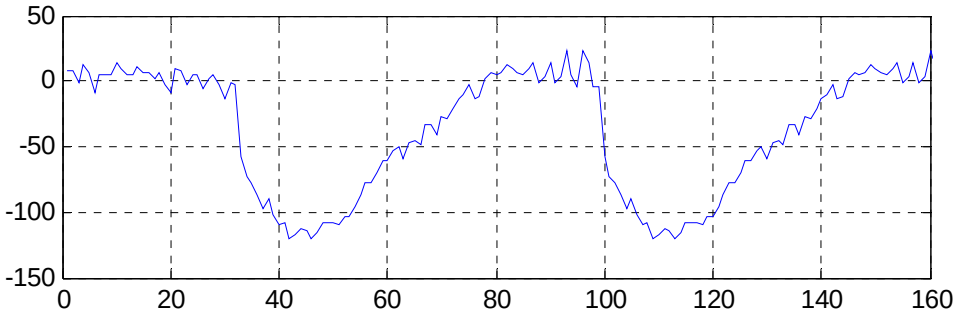


Fig.3.19 Une trame (20 ms) de parole (fichier original1.wav).
La trame du signal résiduel correspondant.
Le pitch vaut 68 échantillons.

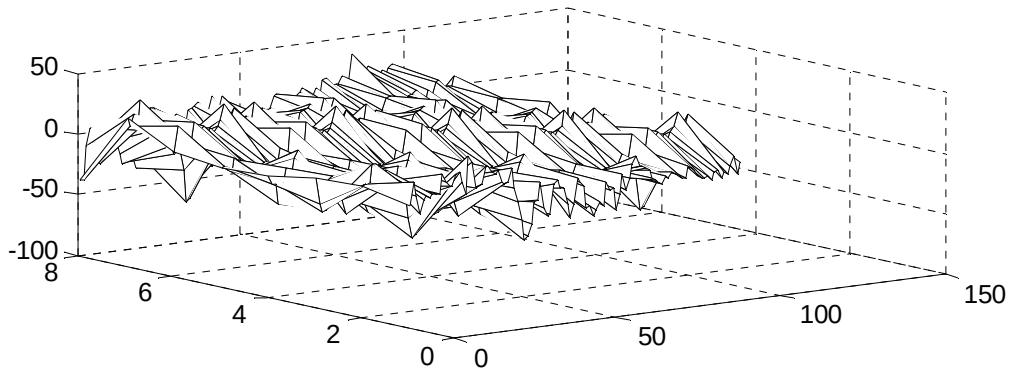


Fig.3.20 Extraction et formation de la surface d'évolution de 8 CW

Conclusion :

Durant ce chapitre nous avons implémenté une méthode d'extraction de formes d'ondes caractéristiques dans le codeur WI. Pour cela nous sommes passé par l'analyse LPC, l'estimation du pitch, interpolation du pitch, extraction des formes d'ondes caractéristiques et enfin construction de la surface d'évolution afin d'avoir une description précise des CW et de leurs évolution dans la trame (comme celle illustrée dans la figure 3.20 et 3.21).
Nous avons pu obtenir des résultats encourageants, mais l'inexistante d'un standard WI et le manque de travaux sur ce codeur à augmenter la difficulté de continuer dans ce codeur.

Conclusion générale

Le codeur de la parole par interpolation de formes d'ondes tire profit de la périodicité fondamentale du discours exprimé. En mettant à jour cette nature périodique, la qualité perceptuelle du discours reproduit peut être améliorée. Ceci est réalisé par le prototype d'extraction des formes d'onde caractéristiques, aux intervalles réguliers et à la transmission de ces derniers.

Nous avons simulé la première partie du codeur WI (jusqu'à l'extraction des CW) en utilisant le langage C.

Dans le premier chapitre, nous avons fourni de l'information de fond au sujet du codage de la parole qui a entouré les propriétés des sons articulés et des attributs de base des codeurs de la parole. Nous avons donné aussi une vue d'ensemble de l'analyse prédictive linéaire à court terme, qui est basée sur la production de la parole et la perception, et donne les différents aspects des codeurs de la parole basés sur l'analyse LP. L'accent est mis sur des méthodes d'obtenir et d'améliorer la performance du filtre inverse. Celles-ci incluent les divers algorithmes pour obtenir un ensemble de coefficient $\{a_i\}$ (10 / trame) (Annexe A).

Le second chapitre présente le concept des codeurs de la parole et les détails du fonctionnement du codeur WI jusqu'à l'extraction des formes d'ondes caractéristiques. Les dérivations mathématiques pour chaque processeur ont été formulées.

Dans le troisième chapitre, nous nous sommes concentrés sur nos tests et résultats en arrivant à notre but principal « Extraction des CW ». Dans notre études, les formes d'onde caractéristiques sont extraites chaque 2.5 ms, c'est-à-dire (8 CW / trame).

En conclusion, le codeur WI offre beaucoup de dispositifs qui ne sont pas communs dans la plupart des codeurs conventionnels de la parole à de bas débit binaire. Certains de ces dispositifs sont énuméré comme suit :

- Le succès du codeur WI est de grande partie due à sa capacité inhérente de produire un niveau précis de la périodicité pour le discours exprimé, même aux débits binaires extrêmement bas. En outre, le WI fournit un excellent et efficace cadre pour analyser, contrôler et régler la périodicité du discours exprimé.

- Un avantage important que WI offre est qu'il décompose la parole en paramètres relativement désaccouplés : les coefficients de LP, puissance, le pitch, les formes d'ondes caractéristiques (SEW/REW). Elle permet aux paramètres d'être quantifié plus efficacement. Elle permet également aux paramètres d'être manipulés et contrôlé séparément, donc plus de précision et d'efficacité.

Perspectives futures:

- Implémenter les différentes parties restantes du codeur WI.
- Quantifier les différents paramètres du codeur à savoir : les LSF, le pitch, la puissance et les formes d'ondes caractéristiques.
- Implémentation de l'algorithme final sur un chip (DSP).

Annexe A

Algorithme de Levinson-Durbin :

Les coefficients d'autocorrelation R(k), k=0,1,……..,P sont utilisées pour obtenir les coefficients du filtre *LP* après résolution du système linéaire (1.13)

Il s'agit donc d'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 $O(p^3)$.

L'algorithme qui va être décrit profite de la structure particulière (Toeplitz symétrique) de la matrice d'autocorrélation pour résoudre (1.13) par une récurssion sur l'ordre de prédiction: autrement dit, ils fournissent 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.

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

$$R(i,j) = R(|i-j|) = R(k) \tag{A.1}$$

Initialisation:

$$a_m(0) = 1, \qquad (m = 1,2,\ldots..,p) \qquad E_0 = R(0) = {\sigma_x}^2$$

Récurssion:

pour: $m = 1,2,...p.$

$$k_m = -\frac{1}{E_{m-1}}\left[R(m) - \sum_{k=1}^{m-1} \alpha_{m-1}(k)R(m-k)\right] \tag{A.2}$$

pour k=1,2,…m-1.

$$\alpha_k(m) = \alpha_k(m-1) - k_m\alpha_{m-k}(m-1) \tag{A.3}$$

$$E_m = E_{m-1}(1-{k_m}^2) \tag{A.4}$$

Les coefficients $a_k(m)$ résultant, quand $m = p$ représentent les coefficients de prédiction d'un prédicteur linéaire d'ordre p :

La valeur de k_m joint à la propriété : $-1 \leq k_m \leq 1$

Cette relation est une condition nécessaire et suffisante pour que le filtre soit stable.

La méthode d'autocorrélation garantit la stabilité du filtre, de plus le calcul de $R(i)$ nécessite un fenêtrage de $S(n)$ par un la fenêtre de Hamming.

Annexe B

Constantes utilisées dans le codeur WI :

Symbole	Valeur	Description
L_f	160	Longueur de la trame
L_{sf}	20	Longueur de la sous-trame = R_{extr}
N	10	Ordre du filtre LP
γ	0.98829	Facteur d'extension de la largeur de bande du filtre LP
L_w	240	Longueur de la fenêtre d'analyse LP
P_{min}	20	Valeur minimale du pitch
P_{max}	120	Valeur maximale du pitch
R_{extr}	8	Nombre d'extractions de CW par trame
δ	10	Longueur de la fenêtre d'énergie à chaque extrémité de la CW
α	0.1	Facteur de pré-accentuation
ε	16	Décalage maximal du point d'extraction

Table B.1 Constantes utilisées dans la simulation

Bibliographie

[1] T.Dutoit, "*Introduction au Traitement Automatique de la Parole*", Faculté Polytechnique de Mons 1989.

[2] R.Boite et M.Kunt,"*Traitement de la parole*", Presses Polytechniques Romandes, première édition.

[3] M. Xie et D.Berkani. "*Amélioration des performances des codeurs de parole*"Août 97

[4] F.Merazka, "Techniques *de codage de la parole : applications aux LSPs et aux systèmes VoIP*", Thèse de Doctorat d'État, Présenté a l'École National Polytechnique Alger 2004.

[5] F.Merazka, "*quantification des paramètres LSF*", Thèse de Magistèr, a l'École National Polytechnique Alger 1997.

[6] F.Itakura and S. Saito, "*Analysis synthesis telephoney based upon the maximum likelihood method*" in Rep 6 th Int. Congr. on acoustics, Kohasi, Ed. Tokyo, Japan Aug. 21-28, 1968, C-5-5.

[7] J. D Markel and A. H. Gray, Jr "*A linear prediction vocoder simulation based upon the autocorrelation method*", IEEE Trans Acoust. Speech. Signal Processing vol ASSP622, PP.124-134, Apr. 1974.

[8] P.Kroon and B.S. Atal, "*Predictive coding of speech using analysis-by-synthesis techniques*", in Advances in *Speech Signal Processing* S. Furui and M.M. Sondhi, Eds New York: Markel- Dekker, pp 141-164. *1991*.

[9] A. H. Gray, and J. D. Markel "*Quantization and bit allocation in speech processing*", IEEE *Trans, on Acoustic, Speech Signal Processing,* vol. ASSP-24, pp. 459-473, Oct. 1976.

[10] P. E. Papamichlis, "*Practical Approaches to Speech Coding*", Prentice-Hall, Englewood Cliffs, N. J. 1987.

[11] D.O'Shaugnessy,"*speech communication, Human and machine.* Reading", MA:Addison-Wesley, 1987.

[12] J.D. Markel and A. H. Gray, Jr., "Linear prediction of speech", New York: Springer-Verlag, 1976.

[13] F.Itakura, "Line *spectrum representation of linear predictive coefficients of speech signals*" *J. Acoust. Soc. Amer.*, vol. 57, suppl. 1 p. S35(A), 1975.

[14] F. K. Soong and B. H. Juang, "*Line spectrum pair (LSP) and speech data compression*", in *Proc. IEEE Int Conf. Acoust. Speech, Signal Processing,* San Diego, CA, pp.1.10.1-1.10.4, Mar.1984.

[15] B.S Atal, R,V Cox and P.Kroon,"*Spectral quantization and interpolation for CELP coders*", in Proc. IEEE int. Conf. On Acoustics, speech and signals.

[16] S. Wang, A. Sekey, and A. Gersho, "*An objective measure for predicting subjective*"

[17] R. Laroia, N Phambo, and N,Favardin,"*Robust abs=d efficient quantization of speech LSP parameter using structured vector quantizer*", in Proc.IEEE Int. Conf on acoustics , speech , and Sig.processing(Toronto, Canada) ,may 1991 pp 641-644.

[18] Alexis Pascal Bernard, "*Source-Channel Coding of Speech*", Master of Science in Electrical Engineering University of California Los Angeles,1998.

[19] A. Das, A. V. Rao, and A. Gersho, " Variable-dimension vector quantization," IEEE Signal Processing Letters, vol. 3, pp. 200-202, July 1996.

[20] W. B. Kleijn, " Continuous representations in linear predictive coding," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Toronto), pp. 201-204, May 1991.

[21] W. B. Kleijn, " Encoding speech using prototype waveforms," IEEE Trans. Speech and Audio Processing, vol. 1, pp. 386-399, Oct. 1993.

[22] W. B. Kleijn and J. Haagen, " A general Waveform-Interpolation structure," Proc. European Signal Processing Conf. (Edinburg), pp. 1665-1668, Sept. 1994.

[23] W. B. Kleijn and J. Haagen, " Speech coder based on decomposition of characteristic waveforms," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Detroit), pp. 508., 511, May 1995.

[24] W. B. Kleijn, Y. Shoham, D. Sen, and R. Hagen, " A low-comlexity Waveform Interpolation coder," Proc. IEEE Int. Conf. on Acoustics, Speech. Signal Processing (Atlanta), pp. 212-215, May 1996.

[25] J. Thyssen, W. B. Kleijn, and R. Hagen, " Using a perception-based frequency scale in Wavefomi Interpolation," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Munich, Germany), pp. 1595-1598, Apr. 1997.

[26] K. K. Paliwal and B. S. Atal, " Efficient vector quantization of LPC parameters at 24 bits/frame," IEEE Trans. Speech and Audio Processing, vol. 1, pp. 3-14, Jan. 1993.

[27] W. B. Kleijn and K. K. Paliwal, eds., Speech Coding and Synthesis. Elsevier, 1995.

[28] W. B. Kleijn and J. Haagen, " Transformation and decomposition of the speech signal for coding," IEEE Signal Processing Letters, vol. 1, pp. 136-138, Sept. 1994.

[29] Telecommunications Industry Association, TIA/EIA/PN-3292, EIA/TIA Interim Standard, Enhanced Variable Rate Codec (EVRC), Mar. 1996.

[30] D. O'Shaughnessy, Speech Communication: Human and Machine. Addison-Wesley Publishing Company, 1987.

[31] L. R. Rabiner and R. W. Schafer, Digital Processing of Speech Signais. Prentice-Hall, 1978.

[32] V. C. Welch and T. E. Tremain, " A new government standard 2400 bps speech coder," Froc IEEE Workshop on Speech Coding for Telecom. (Sainte-Adèle, Québec), pp. 41-42, Oct. 1993.

[33] A. Gersho and R. M. Gray, Vector Quantization and Signal Compression. Kluwer Academic Publishers, 1992.

[34] R. M. Gray, " Vector quantization," IEEE Trans. Acoustics, Speech, Signal Processing, vol. ASSP-34, Apr. 1984.

[35] W. B. Kleijn and W. Granzow, " Methods for waveform interpolation in speech coding," Digital Signal Processing, vol. 1, pp. 215-230, Jan. 1991.

[36] A. S. Spanias, " Speech coding: A tutorial review," Proc. IEEE, vol. 82, pp. 1541-1582, Oct. 1994.

[37] G. Kubin, B. S. Atal, and W. B. Kleijn, " Performance of noise excitation for unvoiced speech," Proc. IEEE Workshop on Speech Coding for Telecom. (Sainte-Adèle, Québec), pp. 35-36, Oct. 1993.

[38] I. A. Atkinson, A. M. Kondoz, and B. G. Evans, " Time envelope vocoder, a new LP based coding strategy for use at bit rates of 2.4 kb/s and below," IEEE J. Selected Areas Commun., vol. 13, pp. 449-457, Feb. 1995.

[39] I. A. Atkinson, A. M. Kondoz, and B. G. Evans, " Time envelope LP vocoder : A new coding technique at very low bit rates," Proc. European Conf. on Speech Commun. and Technology (Madrid), pp. 241-244, Sept. 1995.
[40] 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 J. Selected Areas Commun., vol.10, pp. 830-849, June 1992.

[41] A. McCree, K. Truong, E. B. George, T. P. Barnwell III, and V. Viswanathan, " A 2.4 kbit/s MELP coder candidate for the new U.S. Federal Standard," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Atlanta), pp. 200-203, May 1996.

[42] W. B. Kleijn, Analysis-by-Synthesis Speech Coding Based on Relaxed Waveform Matching Constraints. PhD thesis, Delf University of Technology, Delf, The Netherlands, Dec.1991.

[43] H. Yang and W. B. Kleijn, " Pitch-synchronous subband representation of the linear prediction residual of speech," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Seattle), pp. 529-532, May 1998.

[44] G. H. Golub and C. F. V. Loan, Matrix Computations. The John Hopkins University Press, second ed., 1989.

[45] P. Kabal and R. P. Ramachandran, " The computation of line spectral frequencies using Chebyshev polynomials," IEEE Trans. Acoustics, Speech, Signal Processing, vol. ASSP-34, pp. 1419-1426, Dec. 1986.

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

[47] W. B. Kleijn and J. Haagen, " Waveform interpolation for coding and synthesis," in Speech Coding and Synthesis (W. B. Kleijn and K. K. Paliwal, eds.), pp. 175-208, Elsevier, 1995.

[48] B. S. Atal, V. Cuperman, and A. Gersho, eds., Advances in Speech Coding. Kluwer Academic Publishers, 1991.

[49] J. Stachurski, A Pitch Pulse Evolution Model for Linear Predictive Coding of Speech. PhD thesis, McGill University, Montreal, Canada, May 1997.

[50] M. Leong, " Representing voiced speech using protoype waveform interpolation for low rate speech coding," Master's thesis, McGill University, Montreal, Canada, Nov. 1992.

[51] M. Leong. and P. Kabal, " Smooth speech reconstruction using Prototype Waveform Interpolation," Proc. IEEE Workshop on Speech Coding for Telecom. (Sainte-Adèle, Québec), pp. 39-41, Oct. 1993.

[52] K. Yaghmaie and A. M. Kondoz, "Multiband prototype waveform analysis-synthesis for very low bit rate speech coding," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Munich, Germany), pp. 1571-1574, Apr. 1997.

[53] D. Marston and F. Plante, " PWI speech coder in the speech domain," Proc. IEEE Workshop on Speech Coding for Telecom. (Pennsylvania), pp. 31-32, Sept. 1997.

[54] J. Stachurski and P. Kabal, " A pitch pulse evolution model for a dual excitation linear predictive speech coder," Proc. Seventeenth Biennial Symposium on Communications (Kingston), pp. 107-110, May 1994.

[55] D. W. Griffin and J. S. Lim, " Multiband excitation vocoder," IEEE Trans. Acoustics, Speech, Signal Processing, vol. 36, pp. 1223-1235, Aug. 1988.

[56] J. C. Hardwick and J. S. Lim, " A 4800 bps improved multi-band excitation speech coder," Proc. IEEE Workshop on Speech Coding for Telecom. (Vancouver), Sept. 1989.

[57] Y. Shoham, " Low-rate speech coding based on time-frequency interpolation," Proc. Int. Conf. on Spoken Language Processing, pp. 37-40, Oct. 1992.

[58] A. McCree and W. B. Kleijn, " Mixed Excitation Prototype Waveform Interpolation for low bit rate speech coding," Proc. IEEE Workshop on Speech Coding for Telecom. (Sainte-Adèle, Québec), pp. 51-52, Oct. 1993.

[59] Y. Tanaka and H. Kimura, " Low-bit-rate speech coding using a two-dimensional transform of residual signais and Waveform Interpolation," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Adelaide), pp. 173-176, Apr. 1994.

[60] D. Sen and W. B. Kleijn, " Synthesis methods in sinusoidal and Waveform-Interpolation coders," Proc. IEEE Workshop on Speech Coding for Telecom. (Annapolis), Sept. 1995.

[61] M. R. Zad-Issa and P. Kabal, " A new LPC error criterion for improved pitch tracking," IEEE Workshop on Speech Coding (Pocono Manor, PA), pp. 1-2, 1997.

[62] B. Sylvestre, " Time-scale modification of speech: A time-frequency approach," Master's thesis, McGill University, Montreal, Canada, Apr. 1991.

[63] Y. Jiang and V. Cuperman, " Encoding prototype waveforms using a phase codebook," Proc. IEEE Workshop on Speech Coding for Telecom. (Annapolis), pp. 21-22, Sept. 1995.

[64] M. Festa and D. Sereno, " A speech coding algorithm based on prototype interpolation with critical bands and phase coding," Proc. European Conf on Speech Commun. And Technology (Madrid), pp. 229-232, Sept. 1995.

[65] I. S. Burnett and G. J. Bradley, " Low complexity decomposition and coding of prototype waveforms," Proc. IEEE Workshop on Speech Coding for Telecom. (Annapolis), pp. 23-24, Sept. 1995.

[66] I. S. Burnett and G. J. Bradley, " New techniques for multi-prototype waveform coding at 2.84 kb/s," Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing (Detroit), pp. 261-264, May 1995.

[67] I. S. Burnett and J. Ni, " Waveform Interpolation and analysis-by-synthesis I a good match," IEEE Workshop on Speech Coding (Pocono Manor, PA), pp. 29-30, Sept. 1997.

[68] A. Gersho, "Advances in Speech and Audio Compression," *Proceedings of the IEEE*, vol. 82, pp. 900–918, June 1994.

[69] E. L. T. Choy, "Waveform Interpolation Speech Coder at 4 kb/s," Master's thesis, McGill University, Montreal, Canada, Aug. 1998.