

Un logiciel de codage de la parole basé sur le FS1016

M. DJAMAH, M. BOUDRAA, B. BOUDRAA, M. BOUZID

Laboratoire Communication Parlée et Traitement de Signal (LCPTS)
Faculté Génie-Electrique, USTHB, BP32, EL-ALIA, ALGERIE

ABSTRACT

This paper describes a speech coding software based on the Federal Standard FS1016 coder. The objective of this work is to have in our laboratory a basic speech coder to use it as background for our research works. The modifications and the extensions of the basic coder (to improve the quality) can be done easily by using the object-oriented programming.

1. INTRODUCTION

Le FS1016 (Federal Standard FS1016) est un codeur basé sur la technique CELP [Sch85] [Mor91] et fonctionnant à 4,8 Kbits/s. Le FS1016 a été conjointement développé par la défense nationale des Etats-Unis, le DoD (Departement of Defense), et les laboratoires AT&T Bell pour le compte du DoD. Depuis sa standardisation beaucoup de recherches se font pour son amélioration. Le logiciel que nous avons implanté en « C++ BUILDER 5 » permet de simuler un système de transmission basé sur le standard FS1016. La programmation de toute la chaîne de transmission exclusivement en Orienté Objet rend cette dernière très adaptée à de futures extensions et modifications. Notons que ce logiciel est une conversion à l'Orientée Objet sous Windows du programme en langage C (destiné à l'origine à fonctionner sous station Sun) mis sur Internet par le DoD.

2. PRINCIPE DU CODEUR CELP BASE SUR LE FS1016

Le codeur CELP basé sur le standard FS1016 (figure 1) utilise une fréquence d'échantillonnage de 8 KHZ et une fenêtre d'analyse de 30 ms divisée en quatre sous-fenêtres de 7.5 ms. Le signal synthétique \hat{s} est obtenu par le passage d'un signal d'excitation à travers un filtre de prédiction linéaire $1/A(z)$. L'excitation est calculée par sous-fenêtre. Elle est le résultat de l'addition de deux excitations élémentaires: la première est un vecteur-code extrait du dictionnaire adaptatif à l'indice ia et pondéré par un gain ga . La deuxième est un vecteur-code extrait du dictionnaire stochastique à l'indice is et pondéré par un gain gs . L'analyse du codeur CELP consiste à trouver les paramètres d'excitation (les indices et les gains) de manière à minimiser l'erreur perceptuelle (sortie du filtre perceptuel $A(z)/A(z/\alpha)$ avec $\alpha=0,8$) entre le signal de parole originale s et le signal synthétisé \hat{s} . L'analyse est

dite en boucle fermée. Le codeur contient donc à la fois le module d'analyse et le module de synthèse. L'allocation des bits des différents paramètres à transmettre est donnée à la table 1 [Fen92].

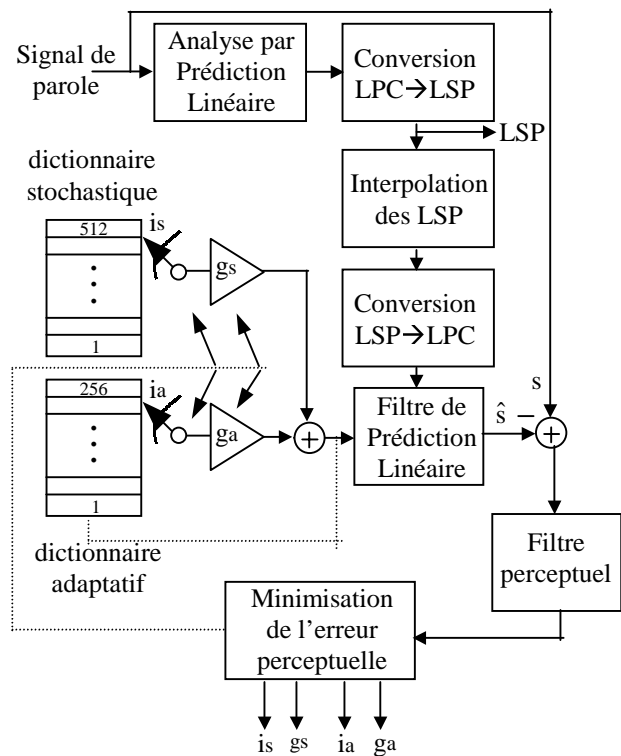


Figure1: Principe du Codeur CELP basé sur le FS1016

Table 1: Allocation des bits selon la norme FS1016

Paramètres/fenêtre.	Bits/fenêtre	Débits
10 LSP	34 [3,4,4,4,4,3,3,3,3,3]	1,13333
4 Indices ia	8 + 6 + 8 + 6	1,600
4 gains ga	4 x 5	
4 Indices is	4 x 9	1,86667
4 gains gs	4 x 5	
Bit de synchronisation	1	0,2
Bits de correction d'erreur	4	
Bit non utilisé	1	
Débit total	144	4, 8 Kbits/s

2.1 Analyse par prédiction linéaire

L'analyse par prédiction linéaire à court terme est réalisée une fois par fenêtre en boucle ouverte sur le signal de parole. Elle comporte l'estimation de 10 coefficients de prédiction a_i par la méthode d'autocorrélation sur des fenêtres de 30 ms (pondéré par la fenêtre de Hamming) et une expansion en largeur de bande de 15HZ. Cela consiste à remplacer les coefficients de prédiction a_i par les coefficients $a'_i = a_i \gamma^i$ ($\gamma = 0.994$) [Fen92]. Ces derniers définissent le filtre de prédiction linéaire $1/A(z)$. Les coefficients a'_i sont convertis aux coefficients LSP (Line Spectral Pairs) qui sont plus adaptés à la transmission.

Les valeurs des coefficients LSP, utilisés dans la détermination de l'excitation des sous-fenêtres, sont obtenues par interpolation linéaire de deux ensembles de coefficients LSP (calculés pour deux fenêtres successifs n et $n+1$) pour former un ensemble intermédiaire pour chacune des 4 sous-fenêtres de la fenêtre à coder (figure 2). L'interpolation linéaire utilise la pondération de la table 2 [Fen92].

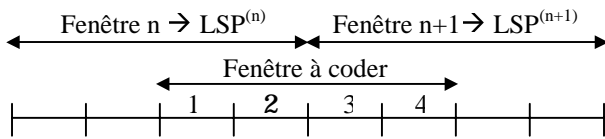


Figure2: Interpolation des LSP selon la norme FS1016

Table 2: Pondération des LSP des sous-fenêtres

Numéro Sous-fenêtre	LSP ⁽ⁿ⁾	LSP ⁽ⁿ⁺¹⁾
1	7/8	1/8
2	5/8	3/8
3	3/8	5/8
4	1/8	7/8

2.2 Recherche dans le dictionnaire adaptatif

Le dictionnaire adaptatif, modélise la périodicité du signal à long terme due au Pitch [Kro90]. Il est constitué d'une mémoire tampon composée de 256 vecteurs-codes correspondants aux excitations passées. Le FS1016 utilise 128 retards entiers et 128 retards fractionnaires allant de 20 à 147 échantillons (table 3) [Fen92].

Table 3: Distribution des retards d'après le FS1016

Intervalle de retard	Résolution
20 – 25 2/3	1/3 échantillon
26 – 33 3/4	1/4 échantillon
34 – 79 2/3	1/3 échantillon
80 – 147	1 échantillon

Le dictionnaire adaptatif est actualisé à chaque passage d'une sous-fenêtre à l'autre. Pour une sous-fenêtre impaire le codage concerne les 256 retards d'où la nécessité de 8 bits pour leur codage. Pour une sous-fenêtre paire, les

retards sont recherchés par la méthode « Delta » et codés avec un offset de 6 bits relativement au retard de la sous-fenêtre précédente. Des sous multiples du retard sélectionné sont examinés pour ne retenir que celui qui est à 1/2 dB du critère de sélection utilisé. Cela a pour effet le lissage des retards correspondant au contour du Pitch et qui est important pour le codage « Delta » [Fen92].

2.3 Recherche dans le dictionnaire stochastique

Le dictionnaire stochastique est composé de 512 vecteurs-codes. Il modélise le signal qui provient du filtre inverse $A(z)$ et auquel la structure périodique due au Pitch est soustraite. Ce signal présente une distribution d'échantillons similaire à celle d'une Gaussienne. C'est pourquoi le dictionnaire stochastique est un dictionnaire entrelacé (déplacement de 2 échantillons) dont les éléments sont quantifiés à 3 niveaux (-1,0,1) et distribués selon une Gaussienne de moyenne nulle et de variance unitaire [Fen92].

Le codeur transmet, à chaque fenêtre de 30 ms, une trame de 144 bits (table 1) à travers le canal de transmission. Après décodage de cette trame, le décodeur CELP reconstruit le signal de parole synthétique en utilisant le même module de synthèse que celui du codeur. C'est-à-dire par passage de l'excitation à travers le filtre de prédiction linéaire. Finalement un postfiltrage adaptatif [Ram88] peut être ajouté pour améliorer le signal de parole synthétique de sortie.

3. STRUCTURE DU LOGICIEL

Le codeur basé sur le FS1016 est implanté exclusivement en utilisant la programmation Orientée Objet [Leb95] qui par sa structure de Classes d'Objets permet une meilleure modulation que la programmation conventionnelle. Notre logiciel est basé sur 7 classes. Chaque classe est destinée à un traitement bien spécifique. Toutes les classes ainsi définies sont reliées entre elles par des héritages simples. L'organisation de ces classes est donnée à la figure 3.

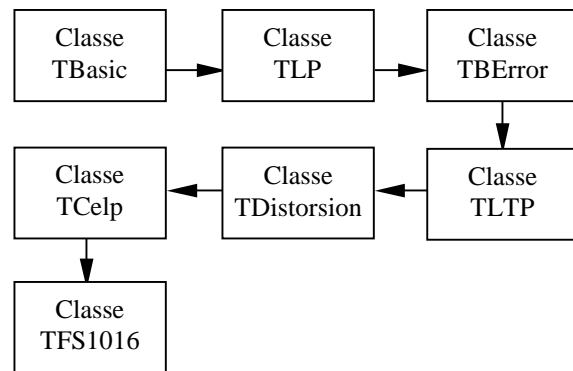


Figure3: Organisation du logiciel en Classes à héritages simples.

Classe TBasic: Contient des fonctions élémentaires: lecture écriture dans les fichiers, copie de données, etc.

Classe TLP: contient les fonctions destinées à la prédiction linéaire : calcul des coefficients de prédiction LPC, passage des LPC aux coefficients LSP et inversement, codage des LSP, décodage des LSP, etc.

Classe TBEror: contient les fonctions destinées au traitement des erreurs de transmission: génération du bruit du canal, corruption des bits de transmission, correction d'erreur de transmission [Jos89] , etc.

Classe TLTP: contient les fonctions destinées au prédicteur long-terme: construction et actualisation du dictionnaire adaptatif, constitution par interpolation d'un vecteur-code à retard fractionnaire, recherche de l'indice ia et du gain ga de l'excitation adaptative, codage et décodage de l'indice et du gain du pitch, etc

Classe TDistorsion: contient les fonctions de calcul de distorsion pour une évaluation objective de la qualité du codeur : calcul du rapport signal sur bruit moyen, calcul de la distorsion spectrale causée par la quantification des coefficients LSP, etc.

Classe TCelp: on trouvera essentiellement les fonctions nécessaires au calcul de l'excitation stochastique: chargement du dictionnaire stochastique, recherche de l'indice is et du gs de l'excitation stochastique, codage et décodage de l'indice et du gain, interpolation des LSP, préfiltrage, postfiltrage, etc

Classe TF1016: c'est la classe finale, elle hérite de toutes les fonctions (non privées) déclarées dans les classes ascendantes. Elle Contient principalement 4 fonctions:

1) fonction CelpInit: Initialisation des variables, exemple: mise à zéro des mémoires des différents filtres, mise à zéro du dictionnaire adaptatif, etc.

2) fonction CelpAnal: Exécute le traitement d'analyse effectué au niveau du codeur (figure 1).et fournit comme résultats une trame de 144 bits correspondants aux codages des paramètres à transmission (table 1).

3) fonction CelpSynt: Exécute le traitement de synthèse effectué au niveau du décodeur. Partant d'un vecteur de 144 bits correspondants aux codes reçus, une éventuelle corruption de ce vecteur est effectuée pour simuler les erreurs de transmission [Jos89]. Après traitement de ces erreurs, on procède au décodage des différents paramètres pour construire le signal synthétique.

4) fonction CelpExec: Exécute toute la chaîne de transmission (codage et décodage). En premier lieu la fonction d'initialisation CelpInit est appelée. Ensuite dans une boucle principale, s'arrêtant au nombre de fenêtres de parole à traiter, on appelle les fonctions CelpAnal et CelpSynt

4. MODIFICATION DU TRAITEMENT

La plupart des fonctions des classes citées plus haut sont déclarées comme étant « Virtual » [Leb95], elles peuvent

donc être redéfinies. Pour modifier la structure du traitement sans modifier la structure de base du codeur (c'est-à-dire sans modifier le contenu des fonctions déjà définies ni de leur enchaînement), il suffit de dériver une nouvelle classe à partir de la classe TFS1016 et de redéfinir les fonctions concernées par les modifications à apporter (exemple: nouvelle méthode d'interpolation ou de quantification des coefficients LSP, etc). Le lancement de la fonction CelpExec à partir d'un objet dont le type est la classe qui a été dérivée de TFS1016, entraîne automatiquement le remplacement de l'ancien contenu des fonctions par le nouveau contenu des fonctions redéfinies. Ce qui correspond au lancement de toute la chaîne de transmission avec toutes les modifications apportées.

5. PRESENTATION DU LOGICIEL

Le logiciel permet de choisir deux types de traitements:

1) Analyse : dans ce cas toute la chaîne de transmission (codage et décodage) est exécutée sur un signal de parole stocké dans un fichier sur le disque. Le logiciel fournit comme résultat le signal de parole synthétisé ainsi qu'un fichier texte, qu'on appellera fichier « Canal ». Le nombre de lignes de ce fichier correspond au nombre de fenêtres de parole traitées et chaque ligne contient les 144 bits correspondants aux codes des paramètres (indices, gains et coefficients LSP) transmis au canal (table 1).

2) Synthèse : effectue la partie synthèse (décodage) à partir du fichier « Canal » et fournit le signal de parole synthétisé.

On donne à la figure 4 la boîte de dialogue « Multi-Pages » permettant de fixer les conditions d'analyse. La page « Parameters » est celle qui est visible.

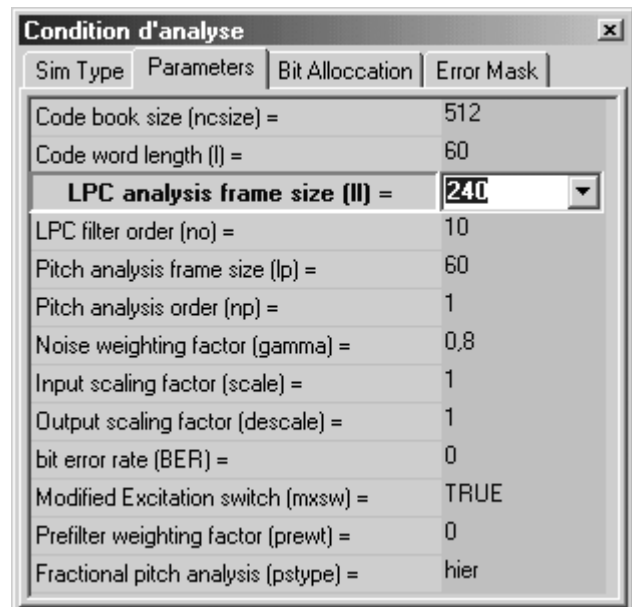


Figure 4: boîte de dialogue « Multi-Pages » permettant de fixer les conditions d'analyse.

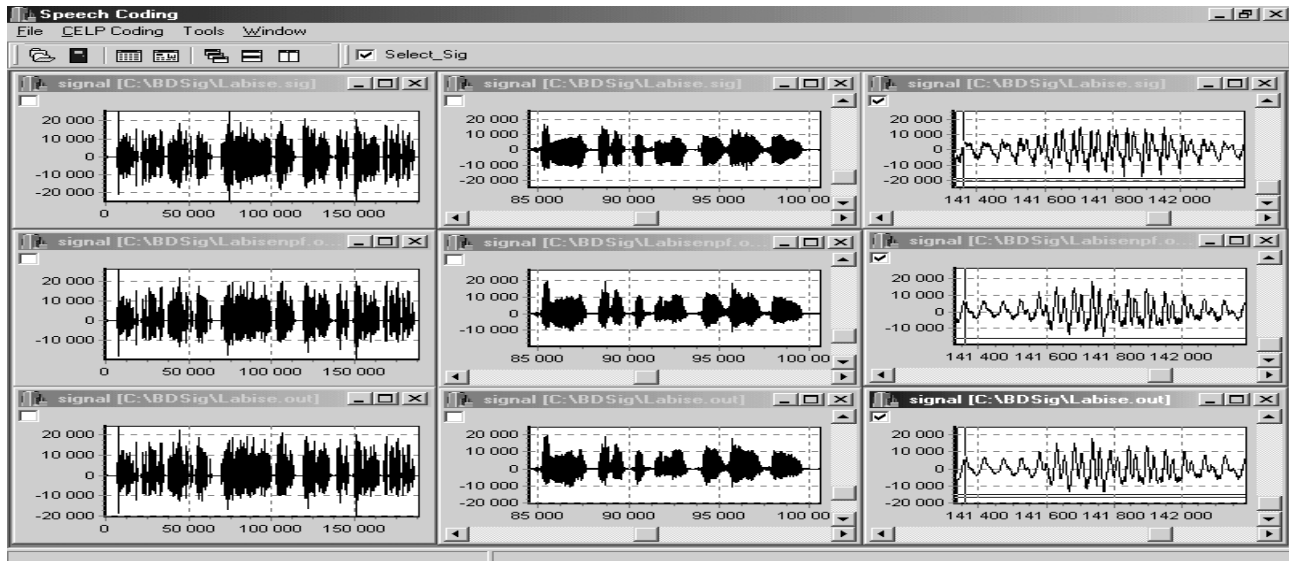


Figure 5: Exemple d'affichage de résultats: à gauche de haut en bas on affiche: le signal de parole originale, le signal de parole synthétisé (sans puis avec postfiltrage), à droite on effectue des zooms sur ces mêmes signaux.

Une fois le traitement effectué, le logiciel permet de visualiser, simultanément, plusieurs signaux sur l'écran. Pour chaque signal des zooms et des défilements (à gauche et à droite) peuvent être effectués. Une option appelée « Select_Sig » permet de sélectionner plusieurs signaux. L'application d'une opération (exemple : un zoom) sur un des signaux sélectionné entraînera automatiquement l'application de la même opération sur tous les signaux sélectionnés (les signaux « cochés »). On donne à la figure 5 un exemple d'affichage des résultats obtenus après traitement d'un signal de parole. A gauche, on affiche le signal de parole original, le signal de parole synthétisé (sans puis avec postfiltrage). A droite on effectue des zooms sur les mêmes signaux.

6. CONCLUSION

Ce logiciel a été conçu dans le but de fournir au sein de notre laboratoire de recherche un codeur de base, notamment le FS1016. L'utilisation du principe « d'héritage » et de « redéfinition de fonctions » utilisé dans la programmation Orientée-Objet [Leb95], rend le logiciel très adapté à de futures modifications et extensions. De futurs travaux pourront donc être facilement « intégrer » dans notre logiciel par « dérivation » d'une nouvelle Classe correspondante au nouveau codeur. Ce dernier pourra ainsi être comparé au FS1016. Notons que dans la chaîne de transmission, le traitement des erreurs de canal est optionnel et peut donc être désactivé. L'étude en détail du traitement des erreurs de canal dans la norme FS1016 reste à faire. Dans la prochaine version du logiciel, il serait intéressant d'implanter le nouveau standard MELP [Sup97]. Notons enfin qu'une description détaillée des fonctions (nom, entrées, sorties) utilisées dans le traitement du codeur de base est nécessaire pour permettre une redéfinition correcte de ces fonctions.

BIBLIOGRAPHIE

- [Sch85] Schroeder M.R, Atal B.S (March 1985), "Code-Excited Linear Prediction (CELP): High-quality speech at very low bit rates", Proc. ICASSP, pp. 937-940.
- [Mor91] N.MOREAU (1991), "Codage prédictif du signal de parole à débit réduit: une présentation unifiée », Annales des Télécom., vol. 46, n° 3-4, pp. 223-239.
- [Fen92] Fenichel R, Bodson D (1992), "Details to assist in implementation of Federal Standard 1016 CELP" Technical Information Bulletin 92-1, National Communication system
- [Kro90] P. Kroon & B. Atal (1990), "Pitch Predictors with High Temporal Resolution," Proc. ICASSP pp. 661-664.
- [Ram88] Ramamoorthy, Jayant, Cox, & Sondhi (Feb. 1988), "Enhancement of ADPCM Speech Coding with Backward-Adaptive Algorithms for Postfiltering and Noise Feedback," IEEE JOSAIC, Vol. 6, NO. 2, pp. 364-382.
- [Leb95] Leblanc G. (1995) BORLAND C++ version 4, Programmation Windows, Troisième tirage, Edition EYROLLES.
- [Jos89] Joseph P., Campbell, Vanoy C. Welch and Thomas E. Tremain (1989), "An Expandable Error-Protected 4800 bps CELP Coder (U.S. Federal Standard 4800 bps Voice Coder)," Proc. ICASSP, pp. 735-738
- [Sup97] Supplee L., Cohn R., Collura J., McCree A. (1997) "MELP: The new Federal Standard AT 2400 BPS", IEEE, pp. 1591-1594