Réduction du débit des LSF's par un système d'énumération en treillis.

Bachir BOUDRAA, Malika BOUDRAA, Mouloud DJAMAH, Merouane BOUZID

USTHB, LCPTS, *Faculté d'Electronique et d'Informatique,* BP 32, El alia, Alger, ALGERIE.
b.boudraa@yahoo.fr; mboudraa@usthb.dz.

Bernard GUERIN

ICP-INPG, 46, Avenue Félix Viallet, 38031, Grenoble Cedex, France

ABSTRACT

In the present study, we were interested in the reduction of the bit-rate observed in the speech coder named CELP FS1016 (federal standard developed by the US department of the defense "DoD "). More precisely, the quantization of the Line Spectrum Frequencies (LSF) parameters was concerned. In the standard CELP FS1016, these coefficients are coded with 34 bits. We propose the use of an enumeration technique in conjunction with a treillis search coding schemes that exploits the natural ordering of the LSF. The technique allows reducing the bit rate of the LSF coefficients to 30 bits without decreasing the performance of the coder.

1. INTRODUCTION

Actuellement, la représentation des coefficients LPC par les paires des raies spectrales est très utilisée [1]-[4], car celles-ci possèdent des propriétés naturelles désirables pour une quantification, comme nous allons le préciser dans la section suivante. Plusieurs schémas de quantification des LSF, aussi bien scalaires (QS) que vectorielles (QV), sont rencontrés dans la littérature [1]-[4]. Dans ce travail, nous avons opté pour la QS à cause de la complexité moindre qu'elle présente comparativement à la QV, ce qui était notre exigence de départ, consistant à mettre au point un codeur de parole fonctionnant en full duplex sur DSP TMS 320C25. D'où notre intérêt pour le codeur CELP (Code Excited Linear Prediction) de la norme FS1016 [5], standard développé par le département de la défense des Etats Unis d'Amérique "DoD", qui répondait a priori à notre exigence. Ce dernier utilise 10 coefficients LSF pour représenter l'enveloppe spectrale. Ceux-ci sont quantifiés de façon scalaire et emploient 34 bits/trame, répartis respectivement comme suit : 3, 4, 4, 4, 4, 3, 3, 3, 3, 3 [5]. Dix tables de quantification formant un total de 112 éléments sont utilisées. Une fine analyse de ces tables révélera que le FS1016 utilise une numérotation dont certaines valeurs ne peuvent être retenues car introduisant des combinaisons de coefficients LSF qui rendraient le filtre de synthèse instable. En effet, les 34 bits précédents peuvent donner $N=\prod_{i=1}^{10} 2^{b_i}$, soit 17179869184 combinaisons possibles où b_i serait le nombre de bits accordé pour pouvoir indexer tous les niveaux de quantification de la $i^{ème}$ table. Cependant, l'ordonnancement naturel des LSF va nous permettre de réduire ce nombre à seulement un maximum de 554958388 (100001000100111111111000110100 en binaire) combinaisons acceptables (soit environ 3.25% du nombre total). Ce chiffre n'exige que 30 bits pour sa représentation.

Tout en gardant une complexité raisonnable, nous avons pu réduire ce débit par l'application d'un système d'énumération en treillis sur les 10 tables précédentes.

La quantification par treillis a eu cette appellation suite à son schéma utilisant des chemins sous forme de treillis. Ces chemins passent d'un niveau de reproduction à un autre se trouvant dans des tables différentes. Plusieurs travaux [6],[7] se sont intéressés à l'application de ce type de quantification aux coefficients LSF. Dans notre cas, nos avons adapté l'algorithme de Malone et Fisher [7] qui nous a permis de récupérer 4 bits/trame ce qui équivaut à un gain de 133 bits/sec.

2. DÉFINITION ET PROPRIETES DES LSF

Les coefficients LSF sont une autre représentation des coefficients de prédiction a_k . Quelques propriétés de ces coefficients LSF peuvent être résumées ainsi [8] :

- Une condition nécessaire et suffisante pour la stabilité du filtre de synthèse $1/A(z)$ est que les coefficients LSF doivent respecter la condition suivante :

$$0 < LSF_1 < LSF_2 < ... < LSF_p < 0.5 \qquad (1)$$

0.5 étant la fréquence de Nyquist. Cette propriété exprime l'ordonnancement de ces coefficients.

- En général, la sensibilité spectrale de chaque coefficient LSF est localisée. Cela veut dire qu'un changement dans un coefficient causera un changement dans le spectre de puissance près de son voisinage.

- Les LSF sont une représentation fréquentielle dont la quantification peut aisément incorporer des traits très importants pour la perception et la sensibilité de l'oreille humaine en particulier.

3. QUANTIFICATION PAR ENUMERATION EN TREILLIS

Une analyse des combinaisons des différents éléments des tables de la QS utilisée dans la norme FS1016 montre qu'il y a un nombre très important de vecteurs LSF qui, s'ils étaient utilisés, rendraient le filtre de synthèse

instable. Aussi, ce système est conçu pour éviter les combinaisons donnant des vecteurs non acceptables en sortie du quantificateur (soit par exemple $LSF_2 < LSF_1$).
Dans ce qui va suivre nous proposons d'appliquer une méthode d'énumération utilisant un système en treillis réduisant le nombre de bits nécessaires aux coefficients LSF à 30 bits au lieu de 34 bits, sans changer les tables de la quantification scalaire suscitée.

4. ALGORITHME DE CODAGE

Pour comprendre l'algorithme de quantification suivant, on précise les notations suivantes :

- LSF_i est le coefficient d'ordre i obtenu après analyse.
- Le mot-code du coefficient LSF_i est donné par $L\hat{S}F_{i,j}$.
  L'opération de quantification sera notée :
  $QS_{34}\,(LSF_i) = L\hat{S}F_{i,j}$
- Pour un ordre de prédiction p, le nombre maximal de combinaisons que les tables de cette QS pourraient donner est : $N_{max} = \prod_{i=1}^{p} 2^{b_i}$ où b_i est le nombre de bits accordé à là i^{eme} table.
- $N_{i+1}(j)$ est le nombre de chemins possibles allant du mot-code du coefficient LSF_i dans la table C_i à l'indice j (soit $L\hat{S}F_{i,j}$), pour atteindre le mot-code du dernier coefficient de la table C_p, tout en vérifiant la propriété d'ordonnancement donnée par l'équation (1).

Ainsi, pour une position j dans la table C_{p-1}, le nombre de combinaisons possibles pour atteindre le dernier coefficient dans la table C_p est donné par :

$$N_p(j) = \sum_{\substack{k=1 \\ L\hat{S}F_{p,k} > L\hat{S}F_{p-1,j}}}^{2^{bp}} 1 \quad , \qquad 1 \le j \le 2^{b_{p-1}} \qquad (2)$$

Dans le tableau 1, on donne le nombre de tous les chemins possibles menant de n'importe quelle position dans une table C_i quelconque jusqu'à la table C_{10} en respectant la propriété d'ordonnancement.

Pour chaque $L\hat{S}F_{i,j}$, $1 \le i \le p-1$, $1 \le j \le 2^{b_i}$ on a :

$$N_{i+1}(j) = \sum_{L\hat{S}F_{i+1,k} > L\hat{S}F_{i,j}} N_{i+2}(k) \qquad (3)$$

Autrement dit, d'un niveau j dans la table C_i, on aura N_{i+1} chemins possibles assurant une bonne succession des coefficients *LSF*. Ces chemins peuvent se déduire de la connaissance du nombre de chemin à l'ordre $i+2$. Notons que la progression est du type arrière (ou *backward)*.

- Le total des combinaisons possibles est :

$$N(p,\overline{C}) = \sum_{j=1}^{2^{b1}} N_2(j) \qquad (4)$$

$$Où \;\; \overline{C} = \{C_1, C_2, ..., C_p\}$$

$N_2(j)$ est le nombre de combinaisons possibles vérifiant :

$$L\hat{S}F_{1,j} < L\hat{S}F_{2,k}\,, \quad 1 \le j \le 2^{b1}\,,\ 1 \le k \le 2^{b2}\,.$$

Formellement, le nombre de bits nécessaires est :

$$K = \lceil \log_2(N(p,\overline{C})) \rceil \qquad (5)$$

Dans le cas du codeur CELP FS1016, $N(p,\overline{C})$ vaut 554958388, où $\lceil x \rceil$ désigne le plus petit entier supérieur à x. K vaut 30 bits dans le cas précédent.

- Dans l'algorithme, m désigne le plus petit ordre dans la table C_i tel que le mot-code d'ordre m soit directement supérieur au coefficient LSF_{i-1} dont l'ordre est j dans C_{i-1}, c-à-d : $L\hat{S}F_{i-1,j} < L\hat{S}F_{i,m}$. Par ailleurs, les valeurs N_i sont celles données au tableau. On notera par c le mot-code de l'énumération en treillis qui sera initialisé à zéro.

Tableau 1 : Nombre de chemins possibles N_i, partant de la table C_1 jusqu'à la table C_{10}

	N_i : Nombres de chemin possibles partant de la table C_{i-1}									
Indice dans C_{i-1}	***N2***	***N3***	***N4***	***N5***	***N6***	***N7***	***N8***	***N9***	***N10***	***N11***
1	*97762793*	*8512383*	*896795*	*96022*	*9455*	*1904*	*315*	*52*	*8*	*1*
2	*97762793*	*8512383*	*896795*	*96022*	*9455*	*1904*	*315*	*52*	*8*	*1*
3	*89250410*	*8512383*	*896795*	*96022*	*9455*	*1904*	*315*	*52*	*8*	*1*
4	*80738027*	*8512383*	*896795*	*96022*	*9455*	*1589*	*315*	*52*	*8*	*1*
5	*72225644*	*8512383*	*896795*	*96022*	*9455*	*959*	*263*	*44*	*7*	*1*
6	*55200878*	*8512383*	*800773*	*96022*	*9455*	*644*	*211*	*36*	*6*	*1*
7	*38176112*	*8512383*	*704751*	*77112*	*9455*	*381*	*107*	*20*	*4*	*1*
8	*23841731*	*7615588*	*608729*	*67657*	*7551*	*170*	*63*	*7*	*3*	*1*
9	***554958388***	*6718793*	*512707*	*58202*	*5647*	***9455***	***1904***	***315***	***52***	***8***
10		*5821998*	*416685*	*39292*	*5647*					
11		*4925203*	*320663*	*29837*	*3743*					
12		*4028408*	*243551*	*22286*	*2154*					
13		*3227635*	*175894*	*10992*	*2154*					
14		*2522884*	*117692*	*7249*	*1195*					
15		*1914155*	*78400*	*5095*	*1195*					
16		*1401448*	*48563*	*2941*	*551*					
		97762793	***8512383***	***896795***	***96022***					

Algorithme:

Etape 1 : *Trouver j tel que* $QS_{34}(LSF_1) = L\hat{S}F_{1,j}$ *avec* $1 \le j \le 2^{b_1}$ *et mettre* $j^* = j\text{-}1$

Etape 2 : *Initialiser* $c = 0$ *si* $j^* = 1$ *sinon*

$$c = \sum_{k=1}^{j^*} N_2(k)$$

Etape 3 : *Pour* $i = 2$ *à* p

$$m = \arg\min_{k=1..2^{b_i}} (L\hat{S}F_{i-1,j^*} < L\hat{S}F_{i,k})$$

Trouver j tel que $QS_{34}(LSF_i) = L\hat{S}F_{i,j}$

avec $1 \le j \le 2^{b_i}$ *et mettre* $j^* = j\text{-}1$

si $j^* \ne m$ *calculer* $c = c + \sum_{k=m}^{j^*} N_{i+1}(k-1)$

Etape 4 : *Le mot-code d'énumération en treillis sera c.*

Pour une meilleure illustration de l'algorithme, nous prenons l'exemple de la quantification du vecteur LSF suivant :
LSF = [0.0409 0.0869 0.1312 0.1621 0.2340 0.2649 0.3052 0.3321 0.3830 0.4073],
obtenu après analyse d'une trame de 30 ms du signal parole. Le déroulement de l'opération de quantification suit les étapes suivantes : dans les deux premières étapes, on cherche le mot-code du premier coefficient LSF_1, en appliquant l'algorithme de la quantification scalaire du FS1016. On s'intéresse ici à son indice. La table C1 de la norme donne un indice $j=6$ et par conséquent on aura : $j^*=6\text{-}1=5$. Comme $j^* \ne 1$, on calculera c en utilisant le tableau 1.

$$c = \sum_{k=1}^{5} N_2(k) = 437739667 \qquad (6)$$

Dans l'étape 3, on déroule une boucle qui commence de LSF_2 pour atteindre LSF_{10}. Pendant ce déroulement, on détermine d'abord l'ordre m comme indiqué plus haut. Dans le cas de LSF_2, m vaut *6*. Par la suite, on cherche, comme dans la première étape, l'indice j du représentant du coefficient LSF_2. Dans cet exemple j vaut *13* et on aura alors $j^* = 12$.
Comme $j^* \ne m$, on calcule c comme suit :

$$c = c + \sum_{k=m}^{j^*} N_{i+1}(k-1) \qquad (7)$$

Ainsi, pour le cas du LSF_2 on aura:

$$c = 437739667 + \sum_{k=6}^{12} N_3(k-1), \quad \text{soit} \quad c = 483874423.$$

Les mêmes opérations seront réitérées pour les autres coefficients. A ce niveau de la quatrième étape, on transmettra seulement l'indice c obtenu à la fin du traitement de la table de LSF_{10} c'est-à-dire c*=483874423* , soit 30 bits, car c vaut en binaire:

011100111111001001000101001100.

5. ALGORITHME DE DECODAGE

Au niveau de la réception, le nombre c sera réceptionné pour être utilisé dans l'opération inverse, selon l'algorithme de décodage suivant:

Algorithme:

Etape 1. *Trouver* $1 \le j \le 2^{b_2}$ *tel que*

$$\sum_{k=1}^{j-1} N_2(L\hat{S}F_{1,k-1}) \le c < \sum_{k=1}^{j} N_2(L\hat{S}F_{1,k-1})$$

Mettre $j^*=j\text{-}1$ *et* $L\hat{S}F_1 = L\hat{S}F_{1,j^*}$

Etape 2. *Calculer* $c = c - \sum_{k=1}^{j^*} N_2(L\hat{S}F_{i,j^*})$

Etape 3. *Pour* $i=2$ *à* p

$$m = \arg\min_{k=1..2^{b_i}} (L\hat{S}F_{i-1,j^*} < L\hat{S}F_{i-1,j^*})$$

Mettre $j = m$ *si* $c= 0$ *sinon*

trouver $1 \le j \le 2^{b_i}$ *tel que*

$$\sum_{k=m}^{j-1} N_{i+1}(L\hat{S}F_{1,k-1}) \le c < \sum_{k=m}^{j} N_{i+1}(L\hat{S}F_{1,k-1})$$

Mettre $j^* = j\text{-}1$ *et* $L\hat{S}F_i = L\hat{S}F_{i,j^*}$

Si $j^* \ne m$ *calculer* $c = c - \sum_{k=1}^{j^*} N_{i+1}(L\hat{S}F_{i,j^*})$

Etape 4. *Le vecteur LSF reconstitué est donné par* $\{L\hat{S}F_1, L\hat{S}F_2, L\hat{S}F_3, \cdots, L\hat{S}F_p\}$

Dans le cas de l'exemple précédent, c a été trouvé égal à *486 314 316*. Dans la première étape de l'algorithme, on cherchera l'indice j tel que c vérifie :

$$\sum_{k=1}^{j-1} N_2(k-1) \le c < \sum_{k=1}^{j} N_2(k-1) \qquad (8)$$

D'après le tableau 1, le représentant du premier coefficient a un indice égal à *6*. D'où $L\hat{S}F_1 = 0.0425$ et, selon l'algorithme, l'indice j^* sera égal à $6 - 1 = 5$.
A la deuxième étape on retranchera de c la valeur $\sum_{k=1}^{5} N_2(k-1) = 437739667$ pour avoir une nouvelle valeur de $c = 48574649$.
A l'étape trois, une boucle qui commencera à partir du coefficient LSF_2 jusqu'au dernier coefficient LSF_{10} itérera les opérations indiquées dans l'algorithme.
Pour le coefficient LSF_2, par exemple, on effectuera les opérations suivantes. D'abord, trouver m tel que $m = M_2(L\hat{S}F_{1,6})$. D'après la table C_2 de la QS du FS1016 [5], m vaut *6*. Comme $c \ne 0$, on cherchera j vérifiant :

$$\sum_{k=m}^{j-1} N_3(k-1) \le c < \sum_{k=m}^{j} N_3(k-1) \qquad (9)$$

j vaut donc 13 selon le tableau 1, car les deux bornes inférieure et supérieure sont égales respectivement à *46134756* et à *4936239*. On met $j^*=j-1=12$ et on attribue à LSF_2 la valeur 0.0838 [5].
De nouveau, $j^* \neq m$, c sera alors comparé à *46134756* et ce pour avoir une nouvelle valeur de c égale à *2439893 (c-46134756 = 2439893).*

Les mêmes étapes que précédemment seront réitérées pour le reste des coefficients *LSF.*

6. RESULTATS ET INTERPRETATIONS

- *Complexité*

Dans notre cas, nous ajoutons (6x8+4x16)=112 additions et autant de comparaisons pour la partie codage. Ceci correspond 0.007 Mflops, qui reste très faible devant la complexité initiale (environ 8.8 Mflops). Pour le décodeur, 278 additions et 674 comparaisons sont ajoutées, comparativement au décodeur initial, soit 0.032 Mflops.

En ce qui concerne l'occupation mémoire, celle des données augmente légèrement. 110 mots de 32 bits sont nécessaires pour stocker les valeurs entières du tableau 1. Ceci est valable aussi bien pour le codeur que pour le décodeur. Pour la mémoire programme, celle-ci augmente de 10% pour le codeur et de 23% pour le décodeur.

- *Distorsion spectrale*

Des tests sur le standard FS1016, aussi bien avec la QS qu'avec l'application du système d'énumération en treillis, ont été effectués sur deux bases de données de parole, l'une extraite de TIMIT [9] et l'autre Arabe (PAPE) [10]. Pour les deux bases de données, les trames silencieuses des débuts et des fins de phrases ont été limitées à un maximum de 2 trames.
Un total de 218 334 trames de 30 ms a été utilisé. Les performances du quantificateur sont évaluées par la distorsion spectrale moyenne SD qui est souvent utilisée comme mesure objective de la performance d'encodage. SD fournit une bonne corrélation avec la perception auditive humaine.

Le traitement de 218 334 trames a donné une distorsion spectrale moyenne de 1.73 dB, 13.88 pour les % 2-4 dB et 0.22 pour les % supérieurs à 4 dB, confirmant les résultats donnés dans la littérature concernant ce standard [2]-[5].

La sensibilité aux erreurs binaires mérite d'être approfondie ultérieurement. En effet, nous avons constaté une légère dégradation des performances de cette modification du codeur (sans protection) dès que le BER (bit error rate) dépasse 10^{-2} (cas d'un BSC). Les mêmes distorsions ne sont observées qu'aux environs de $5x10^{-3}$ dans le cas du FS 1016 classique.

7. CONCLUSION.

Dans cet article, nous avons présenté le système d'énumération en treillis que nous proposons d'appliquer à la quantification des coefficients LSF en vue de diminuer le débit alloué à ces coefficients dans le standard CELP FS1016, sans changer ses tables de quantification. En fait, nous avons noté que seulement 3.25 % des combinaisons totales que l'on peut former avec les éléments de ces tables de quantification, peuvent vérifier la propriété d'ordonnancement des coefficients LSF et conduisent ainsi à un filtre stable. L'application d'un système d'énumération en treillis nous a permis de diminuer le nombre de bits nécessaires au codage des LSF de 34 bits à 30 bits. Les performances de ce système d'énumération ont été évaluées sur 218 334 vecteurs LSF et ont donné une distorsion spectrale de 1.73 dB, avec des outliers de 13.88 % pour 2-4 dB et 0.22 % pour ceux >4dB, résultat analogue à celui du standard FS1016, mais avec un gain en débit de 133 bits/s.

BIBLIOGRAPHIE

[1] K. Paliwal, W. Kleijn. Quantization of LPC parameters, (Chapter 12). In *speech Coding and Synthesis*. Ed. by W. Kleijn and K. Paliwal, Elsevier, Amsterdam,pp. 433-466, 1995.

[2] ICASSP'95, session: Spectral quantization, 10 articles différents. In *IEEE-ICASSP*, pp. 716-755, 1995.

[3] ICASSP'96. Session: Spectral quantization, 10 articles différents. In *IEEE-ICASSP*, Vol. 1, pp. 737-776, 1996

[4] ICASSP'97. Session: Speech coding at low bit rates, 14 articles différents. In *IEEE-ICASSP*, Vol. 2, pp. 1555-1610, 1997

[5] J.P. Campbell Jr., V.C. Welch, T.E. Tremain. An expandable error protected 4800 bps CELP coder. In *IEEE-ICASSP*, pp. 735-738, 1989.

[6] M.W. Marcellin & T.R. Fischer. Trellis coded quantization of memoryless and Gauss-Markov sources. *In IEEE Trans. on Com.,* Vol. 38, pp. 83-93 Jan 1990.

[7] T. Malone, T. R. Fisher. Enumeration and trellis searched coding schemes for speech LSP parameters. In *IEEE Trans. on speech and audio proc.*, vol.1, N°3, pp. 304-314, July 1993.

[8] F. Itakura. Line spectrum representation of linear predictive coefficients of speech signals. In *JASA*, vol. 57, suppl 1, pp. S35 (A), 1975.

[9] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue. Timit acoustic-phonetic continuous speech corpus. In *Linguistic Data Consortium*, 1993.

[10] M. Boudraa, B. Boudraa, B. Guérin. Twenty List of Ten Arabic Sentences for Assessment. I*n Acustica associated with Acta-Acustica,* vol.86, pp. 870-882, 2000.