

**DETERMINATION ET CODAGE DE L'EXCITATION  
DANS UN CODEUR CELP**

P.DYMARSKI \*, N.MOREAU \*\*, W.VOS \*\*\*

\* Ecole Polytechnique de Varsovie, 15 rue Nowowiejska, VARSOVIE

\*\* TELECOM Paris, 46 rue Barrault, 75013 PARIS

\*\*\* Université Technique de DELFT, 4 Mekelweg, 2600 GA DELFT

**RÉSUMÉ**

Dans un codeur de parole à débit réduit de type CELP, le problème essentiel est de réaliser une bonne modélisation du signal d'excitation du filtre de synthèse. On propose d'abord une formalisation matricielle de ce problème. On montre ensuite qu'entre l'algorithme optimal inaccessible et l'algorithme standard rudimentaire existent plusieurs algorithmes intermédiaires mettant en jeu des coûts de calcul raisonnables. Ces algorithmes réalisent des factorisations QR et sont bâtis sur des orthogonalisations de Gram-Schmidt. On s'intéresse aussi à la quantification des gains des différents vecteurs sélectionnés dans le (ou les) dictionnaire. On montre qu'il est préférable de coder les gains relatifs aux vecteurs orthogonalisés et normés. On présente des résultats de simulation relatifs à un codeur pour un débit voisin de 9 kbit/s.

**ABSTRACT**

The goal of this paper is to investigate several algorithms that construct the synthesis filter's input in the CELP coder. A matrix formulation of this problem is recalled. Iterative algorithms are then analyzed in order to extend the results presented in a previous paper. These results showed that, between the inaccessible optimal algorithm and the rudimentary standard algorithm there are several intermediate algorithms with acceptable computational complexity. Starting with the standard iterative algorithm, the QR factorization is introduced into the computation loop. Gains can be computed relatively to both the original codebook vectors and the orthogonalized codebook vectors. We show that these two sets of gains have different coding properties. These algorithms are evaluated with respect to their computational cost and SNR improvement using realistic values for the parameters. The bit rate chosen for this test is around 9 kbit/s.

**I. INTRODUCTION**

On considère le schéma de principe d'un codeur CELP proposé Fig. 1 où tous les signaux qui apparaissent sont des vecteurs que l'on supposera, pour simplifier et sans perte de généralité, de même dimension N. La méthode la plus générale pour modéliser l'excitation du filtre de synthèse consiste à définir K dictionnaires, a priori distincts, représentés par les matrices  $C_{(1)} \dots C_{(K)}$  de dimension respective  $N \times \Delta_1 \dots N \times \Delta_K$  et à rechercher un vecteur d'excitation sous la forme d'une combinaison linéaire :

$$\hat{r} = \sum_{k=1}^K g_k c^{j(k)}$$

où  $c_{j(k)}^{j(k)}$  est le  $j(k)^{ème}$  vecteur colonne du dictionnaire  $C_{(k)}$  [1]. Pour simplifier, on suppose, à partir de maintenant, que les K vecteurs  $c^{j(k)}$  appartiennent au même dictionnaire C. La structure de ce codeur CELP, apparemment simple, reste encore trop complexe pour pouvoir utiliser directement les résultats classiques de théorie de l'information et du traitement du signal. Il s'agit pourtant d'un simple problème de minimisation au sens des moindres carrés avec, toutefois, deux caractères spécifiques. La minimisation porte sur des signaux perceptuels reliés aux signaux composant le dictionnaire d'excitation par une opération de filtrage dont les caractéristiques changent toutes les 20 ms. On cherche à sélectionner un nombre très restreint de vecteurs, typiquement 2 ou 3, parmi un grand nombre, typiquement 256 ou 512.

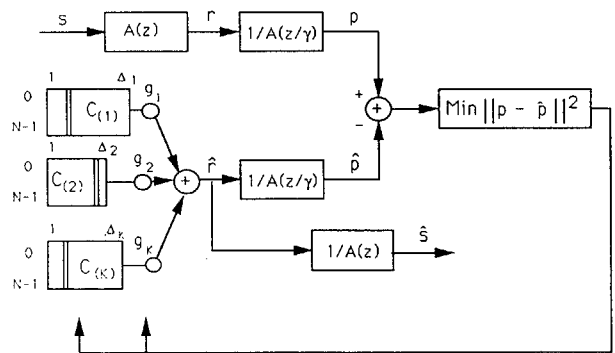


Fig.1

Dans cet article, on propose d'abord une formalisation matricielle de ce problème en faisant apparaître clairement une matrice Toeplitz triangulaire inférieure caractérisant le filtrage et en mettant en évidence une matrice particulière pour décrire la procédure de sélection. On montre qu'entre l'algorithme optimal inaccessible et l'algorithme standard rudimentaire existent plusieurs algorithmes intermédiaires mettant en jeu des coûts de calcul raisonnables. Ces algorithmes réalisent des factorisations QR et sont bâtis sur des orthogonalisations de Gram-Schmidt. On s'intéresse aussi à la quantification des gains des différents vecteurs sélectionnés dans le (ou les) dictionnaire. On montre qu'il est préférable de coder les gains relatifs aux vecteurs orthogonalisés et normés. On présente des résultats de simulation relatifs à un codeur pour un débit voisin de 9 kbit/s.



## II. FORMALISATION DU PROBLEME

### II.1 Expression du critère

On rappelle brièvement quelques résultats présentés dans [2]. On minimise la norme euclidienne :

$$E = \| p - \hat{p} \|^2$$

où le vecteur perceptuel  $p$  est un vecteur connu et où le vecteur perceptuel modélisé  $\hat{p}$  dépend non seulement des paramètres inconnus  $j(1) \dots j(K)$  et  $g_1 \dots g_K$  mais aussi de l'excitation  $\hat{r}$  correspondant à la fenêtre précédente. La relation :

$$\hat{p}_n = \sum_{i=0}^n h_i \hat{r}_{n-i} + \sum_{i=n+1}^{\infty} h_i \hat{r}_{n-i} = \sum_{i=0}^n h_i \hat{r}_{n-i} + \hat{p}_r^0$$

où  $h_i$  est la réponse impulsionnelle du filtre perceptuel  $1/A(z/\gamma)$  dans la fenêtre courante, s'écrit de façon matricielle en mettant en évidence une matrice de Toeplitz triangulaire inférieure  $H$  :

$$\hat{p} = H \hat{r} + \hat{p}^0$$

Il existe une relation équivalente entre le signal résiduel  $r$ , le signal perceptuel  $p$  et la contribution  $p^0$  des fenêtres précédentes. On obtient donc :

$$E = \| (H r + p^0 - \hat{p}^0) - \sum_{k=1}^K g_k H c^{j(k)} \|^2$$

On simplifie les notations en posant  $p = H r + p^0 - \hat{p}^0$  et on adopte les notations suivantes. A partir du dictionnaire d'excitation  $C$ , on obtient le dictionnaire filtré  $F$  avec  $F = H C$ . On construit la matrice  $A$  par extraction des  $K$  vecteurs  $f^{j(1)} \dots f^{j(K)}$  de  $F$ . On introduit la matrice  $\Pi$  telle que  $A = F \Pi$ . La  $k^{\text{eme}}$  colonne de la matrice  $\Pi$  prend la valeur 1 à l'indice  $j(k)$  et les valeurs 0 ailleurs. On remarque que la matrice  $\Pi^T \Pi$  est la matrice identité de dimension  $K \times K$  et que la matrice  $\Pi \Pi^T$  est une matrice carrée de dimension  $\Delta \times \Delta$  comportant des 1 uniquement sur la diagonale principale aux positions définies par  $j(1) \dots j(K)$ . On définit le vecteur  $g = (g_1 \dots g_K)^T$ . Avec ces notations, le critère s'écrit :

$$E(\Pi, g) = \| p - F \Pi g \|^2 = (p - F \Pi g)^T (p - F \Pi g)$$

### II.2 Minimisation du critère

Si on impose  $\Pi = \Pi_0$ , on sait [3] que la minimisation de  $E(\Pi_0, g)$  par rapport à  $g$  conduit aux systèmes d'équations normales :

$$A^T A g_{\text{opt}} = A^T p$$

et que la valeur du critère pour  $g = g_{\text{opt}}$  est :

$$E(\Pi_0, g_{\text{opt}}) = \| p \|^2 - \| \hat{p}_{\text{opt}} \|^2 = p^T p - p^T A (A^T A)^{-1} A^T p$$

La détermination de l'excitation dans un codeur CELP revient donc à obtenir  $\Pi_{\text{opt}}$  par maximisation de :

$$\| \hat{p} \|^2 = p^T F \Pi (\Pi^T F^T F \Pi)^{-1} \Pi^T F^T p$$

puis à calculer  $g_{\text{opt}}$  par résolution du système d'équations normales. On sait qu'il existe un algorithme optimal mais qu'il est inaccessible. Les algorithmes standards consistent à calculer les indices et les gains de façon itérative, l'expression  $A^T A$  étant alors un scalaire.

### II.3 Cas particulier : Dictionnaire filtré orthogonal

On considère le cas où le dictionnaire filtré  $F$  est composé de vecteurs orthogonaux [4]. Si l'on applique les résultats précédents, on doit d'abord maximiser :

$$\| \hat{p} \|^2 = p^T F \Pi (\Pi^T F^T F \Pi)^{-1} \Pi^T F^T p = \tilde{p}^T \Pi \Pi^T \tilde{p} = \sum_{k=1}^K \tilde{p}_{j(k)}^2$$

où  $\tilde{p} = F^T p$  est le vecteur perceptuel transformé. Les indices sélectionnés  $j(1) \dots j(K)$  correspondent donc aux composantes les plus significatives du vecteur  $\tilde{p}$ . Après avoir choisi ces indices, on calcule le vecteur des gains :

$$g_{\text{opt}} = (\Pi^T F^T F \Pi)^{-1} \Pi^T F^T p = \Pi^T \tilde{p} = \left[ \tilde{p}_{j(1)} \dots \tilde{p}_{j(K)} \right]^T$$

Le modèle du signal perceptuel a pour expression :

$$\hat{p} = F \Pi \Pi^T \tilde{p} = F \begin{bmatrix} 0 \dots 0 & \tilde{p}_{j(1)} & 0 \dots 0 & \tilde{p}_{j(K)} & 0 \dots 0 \end{bmatrix}^T$$

L'algorithme optimal, appliqué sur un dictionnaire filtré orthogonal, consiste donc à transformer le vecteur  $p$  par  $F^T$ , à forcer à zéro les composantes les moins significatives du vecteur transformé  $\tilde{p}$  puis à revenir au modèle  $\hat{p}$  par la transformée  $F$ . Il s'agit d'un codage par transformée.

## III. ALGORITHMES ITERATIFS

### III.1 Algorithme itératif standard

On calcule, à chaque étape, un indice et un gain. Ce calcul réclame la connaissance de l'énergie des vecteurs filtrés  $\alpha^j = \langle f^j, f^j \rangle$  et des intercorrélations initiales  $\beta^j = \langle f^j, p \rangle$  et nécessite éventuellement la réactualisation de ces intercorrélations. A partir de maintenant, on introduit les gains relativement aux vecteurs normalisés que l'on note  $\tilde{g}$ . Les équations normales, dans le cas scalaire, se réduisent à  $\sqrt{\alpha^j} \tilde{g} = \beta^j$ . L'actualisation des intercorrélations se faisant de la façon suivante :

$$\beta_{k+1}^j = \langle f^j, p - \sum_{n=1}^k \tilde{g}_n \frac{f^{j(n)}}{\| f^{j(n)} \|} \rangle = \beta_k^j - \tilde{g}_k \langle f^j, \frac{f^{j(k)}}{\| f^{j(k)} \|} \rangle$$

et en appelant  $r_{jk}$  la projection de  $f^j$  sur le vecteur normalisé sélectionné, on obtient l'algorithme standard :

Pour  $j = 1 \dots \Delta$

$$\bullet \alpha^j = \langle f^j, f^j \rangle \quad \text{et} \quad \beta^j = \langle f^j, p \rangle$$

Pour  $k = 1 \dots K$

$$\bullet j(k) = \underset{j}{\text{Arg Max}} \left( \frac{\beta_k^j}{\sqrt{\alpha^j}} \right)^2$$

$$\bullet \tilde{g}_k = \frac{\beta_k^{j(k)}}{\sqrt{\alpha^{j(k)}}}$$

• Pour  $j = 1 \dots \Delta$  (si  $k < K$ )

$$\bullet r_{kj} = \frac{1}{\sqrt{\alpha^{j(k)}}} \langle f^{j(k)}, f^j \rangle$$

$$\bullet \beta_{k+1}^j = \beta_k^j - r_{kj} \frac{\beta_k^{j(k)}}{\sqrt{\alpha^{j(k)}}}$$

### III.2 Réduction de la sous optimalité par factorisation QR

On rappelle [3] que la factorisation QR d'une matrice de rang complet  $A$  consiste à déterminer une matrice orthogonale  $Q$  et une matrice triangulaire supérieure  $R$  telles que  $A = QR$ . Cette décomposition est unique si les vecteurs de  $Q$  forment une base orthonormée et si les éléments sur la diagonale principale de  $R$  sont tous positifs.  $R$  coïncide alors avec la matrice obtenue par décomposition de Choleski de la matrice définie positive  $A^T A$ . Deux algorithmes de Gram-Schmidt ("classique" et "modifié") sont présentés dans [3]. Si le calcul explicite des vecteurs de  $Q$  n'est pas nécessaire, on montre [2] qu'il existe un troisième algorithme fournissant les éléments  $r_{jk}$  de la matrice  $R$  en fonction des vecteurs colonnes  $f^j$  de  $A$  :

Pour  $j = 1 \dots K$

$$\bullet \alpha_k^j = \langle f^j, f^j \rangle$$

Pour  $k = 1 \dots K$

$$\bullet r_{kk} = \sqrt{\alpha_k^k}$$

• Pour  $j = k+1 \dots K$

$$\bullet r_{kj} = \frac{1}{r_{kk}} \langle f^k, f^j \rangle - \sum_{i=1}^{k-1} r_{ik} r_{ij}$$

$$\bullet \alpha_{k+1}^j = \alpha_k^j - r_{kj}^2$$

La première façon de réduire la sous optimalité de l'algorithme standard est de réaliser l'optimisation globale des gains après avoir déterminé  $j(1) \dots j(K)$ . Cela consiste à appliquer l'algorithme standard (sans le calcul des gains  $\tilde{g}_k$ ) puis un

algorithme de Gram-Schmidt quelconque, par exemple l'algorithme modifié récursif, sur les  $K$  vecteurs  $f^{(1)} \dots f^{(K)}$ . Il faut toutefois rajouter le calcul des intercorrélations entre l'erreur de modélisation et les vecteurs orthogonalisés puisqu'elles interviennent dans le calcul des gains. La deuxième façon de réduire la sous optimalité consiste à optimiser les gains à chaque itération. On montre qu'il suffit d'incorporer l'algorithme de Gram-Schmidt modifié dans la boucle de calcul de l'algorithme standard. Il faut modifier la façon de calculer les gains  $\tilde{g}_k$  et l'actualisation des intercorrélations  $\beta_k^j$ . Par contre, on continue à chercher les indices  $j(k)$  comme auparavant. On doit donc utiliser  $\alpha_k^j$  dans le calcul de  $j(k)$  alors que, dans le calcul de  $\tilde{g}_k$ , on utilise  $\alpha_k^{j(k)}$ . On obtient l'algorithme :

Pour  $j = 1 \dots \Delta$

- $\alpha_k^j = \langle f^j, \tilde{f} \rangle$  et  $\beta_k^j = \langle f^j, p \rangle$

Pour  $k = 1 \dots K$

- $j(k) = \text{Arg Max}_j \left( \frac{\beta_k^j}{\sqrt{\alpha_k^j}} \right)^2$
- $\tilde{g}_k = \frac{\beta_k^{j(k)}}{\sqrt{\alpha_k^{j(k)}}}$
- Pour  $j = 1 \dots \Delta$  (si  $k < K$ )
  - \*  $r_{kj} = \frac{1}{\sqrt{\alpha_k^{j(k)}}} [\langle f^{j(k)}, f^j \rangle - \sum_{i=1}^{k-1} r_{ij(k)} r_{ij}]$
  - \*  $\alpha_{k+1}^j = \alpha_k^j - r_{kj}^2$
  - \*  $\beta_{k+1}^j = \beta_k^j - r_{kj} \frac{\beta_k^{j(k)}}{\sqrt{\alpha_k^{j(k)}}}$

Pour obtenir l'algorithme RMGS, non seulement le calcul des gains mais aussi la recherche de l'indice optimal à l'étape  $k$  doivent se faire relativement à des vecteurs orthogonalisés. Il suffit donc de remplacer  $\alpha_k^j$  par  $\alpha_k^j$  dans le calcul de  $j(k)$  pour obtenir l'algorithme RMGS décrit dans [2].

### III.3 Algorithmes itératifs dans le cas général

On dispose de  $K$  dictionnaires d'excitation représentés par les matrices  $C_{(1)} \dots C_{(K)}$ . Un algorithme itératif se décompose en deux parties. A l'étape  $k$ , on calcule d'abord les énergies  $\alpha_k^j$  et les intercorrélations  $\beta_k^j$  (ce calcul est réalisé lui-même soit par initialisation à la première étape, soit par actualisation sinon) puis on utilise ces valeurs pour calculer l'indice  $j(k)$  et le gain correspondant  $\tilde{g}_k$ . Dans le cas général, il suffit de préciser, à chaque étape, sur quel intervalle porte le calcul des énergies et des intercorrélations et sur quel intervalle il faut rechercher un vecteur. Si l'on note  $j \in C_{(k)}$  le fait que l'indice  $j$  parcourt l'ensemble des indices du dictionnaire  $C_{(k)}$ , on obtient la forme général de l'algorithme RMGS :

Pour  $j \in C_{(1)} \dots C_{(K)}$

- $f^j = H c^j$
- $\alpha_k^j = \langle f^j, \tilde{f} \rangle$  et  $\beta_k^j = \langle f^j, p \rangle$

Pour  $k = 1 \dots K$

- $j(k) = \text{Arg Max}_{j \in C_{(k)}} \left( \frac{\beta_k^j}{\sqrt{\alpha_k^j}} \right)^2$
- $\tilde{g}_k = \frac{\beta_k^{j(k)}}{\sqrt{\alpha_k^{j(k)}}}$
- Pour  $j \in C_{(k+1)} \dots C_{(K)}$ 
  - \*  $r_{kj} = \frac{1}{\sqrt{\alpha_k^{j(k)}}} [\langle f^{j(k)}, f^j \rangle - \sum_{i=1}^{k-1} r_{ij(k)} r_{ij}]$
  - \*  $\alpha_{k+1}^j = \alpha_k^j - r_{kj}^2$
  - \*  $\beta_{k+1}^j = \beta_k^j - r_{kj} \frac{\beta_k^{j(k)}}{\sqrt{\alpha_k^{j(k)}}}$

Cette formulation n'est pas classique puisque, d'habitude, on calcule l'énergie  $\alpha_k^j$  des vecteurs filtrés et les intercorrélations  $\beta_k^j$  uniquement sur le premier dictionnaire, on détermine un indice et un gain, on évalue l'erreur de modélisation puis on recommence ces opérations sur le deuxième dictionnaire, etc ... Si l'on compare la complexité du traitement de l'algorithme dans sa formulation classique, i.e. avec évaluation de l'erreur de modélisation à chaque étape, et dans la formulation proposée, on peut remarquer que l'algorithme standard doit être utilisé dans sa forme classique mais que la formulation proposée est préférable lorsque l'on utilise l'algorithme RMGS.

## IV. ALGORITHME DE CODAGE DES GAINS

### IV.1 Introduction

Dans la littérature, l'approche la plus fréquemment utilisée pour coder les gains consiste à réaliser une quantification scalaire non uniforme des gains  $g_k$  relatifs aux vecteurs  $f^{j(k)}$ . On propose une nouvelle méthode exploitant la distribution particulière des gains  $\tilde{g}_k$  relatifs aux directions orthonormées  $f_{\text{orth}(k)}^{j(k)}$  [5]. Pour cela, on applique l'algorithme standard à un signal composé de 4 doubles phrases phonétiquement équilibrées, prononcées par 2 locuteurs féminins et 2 locuteurs masculins, l'ensemble représentant environ 24 secondes. Le dictionnaire d'excitation est, dans cette expérience, un dictionnaire mixte composé de 256 vecteurs de 40 échantillons, la moitié de ces vecteurs sont construits à partir de l'excitation passée (dictionnaire prédictif), l'autre moitié étant peuplée par une variable aléatoire gaussienne (dictionnaire stochastique). La Fig. 2 montre les histogrammes des gains  $g_k$ , des rapports  $g_{k+1}/g_k$ , des gains  $\tilde{g}_k$  et des rapports  $\tilde{g}_{k+1}/\tilde{g}_k$ . Seules les fenêtres où le signal a une énergie supérieure à un seuil donné sont utilisées. On observe que les rapports  $\tilde{g}_{k+1}/\tilde{g}_k$  semblent de bons candidats, les gains  $\tilde{g}_k$  paraissant plus corrélés entre eux que les  $g_k$ . On examine les problèmes suivants : comment coder le premier gain  $\tilde{g}_1$ , est-il intéressant d'utiliser une quantification vectorielle, la quantification est-elle possible dans la boucle, que se passe-t-il lorsque l'on recherche un vecteur dans un dictionnaire prédictif puis  $K-1$  vecteurs dans un dictionnaire stochastique ?

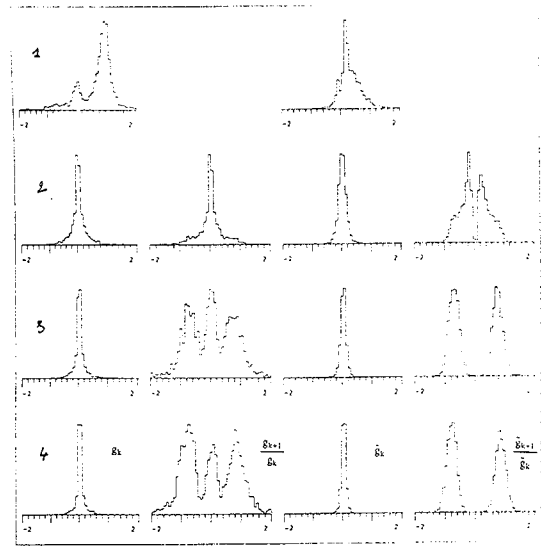


Fig. 2

### IV.2 Algorithmes de quantification

La Fig. 2 montre que le premier gain a une dynamique importante. On recherche un paramètre qui a un comportement meilleur en terme de quantification, i.e. un paramètre avec une variance faible. Il existe plusieurs solutions. Etant donnée la relation :

$$\|\hat{p}\|^2 = \sum_{k=1}^K \tilde{g}_k^2 = \tilde{g}_1^2 \left( 1 + \left( \frac{\tilde{g}_2}{\tilde{g}_1} \right)^2 \left( 1 + \dots + \left( \frac{\tilde{g}_K}{\tilde{g}_{K-1}} \right)^2 \right) \right)$$



la connaissance de  $\|\hat{p}\|^2$  et des rapports successifs permet de calculer tous les gains. Puisque, à l'émetteur, on ne connaît  $\|\hat{p}\|^2$  qu'après la détermination des gains, on remarque qu'un codage dans la boucle est impossible. Par contre, une quantification vectorielle de l'ensemble des paramètres ne pose pas de difficultés particulières. Pour pouvoir introduire le codage des gains dans la boucle, on procède de la façon suivante. On remplace d'abord l'énergie du signal perceptuel synthétique par celle du signal original. Cette approximation, d'autant meilleure que la modélisation est mieux faite, est compensée par l'introduction d'un facteur  $\lambda$  défini par :

$$\|\hat{p}\|^2 = \lambda \|p\|^2$$

Cette opération n'est pas, a priori, intéressante, puisque l'on doit coder maintenant l'énergie  $\|p\|^2$  et le coefficient  $\lambda$ . On joue sur le fait que l'on peut coder grossièrement, par exemple par l'intermédiaire d'une quantification logarithmique sur 5 bits, l'énergie toutes les 20 ms et que la distribution de  $\lambda$  est meilleure que celle de  $\tilde{g}_1$ . Plusieurs variantes sont possibles. Etant donné que :

$$\|\hat{p}\|^2 = \|p\|^2 = \|s\|^2 \prod_{i=1}^P \frac{1 - k_i^2}{1 - k_i'^2}$$

où les  $k_i$  sont les coefficients de corrélation partielles du filtre  $A(z)$  et les  $k_i'$  sont les coefficients du filtre  $A(z/\gamma)$ , on peut modifier la définition de  $\lambda$  en posant :

$$\|\hat{p}\|^2 = \lambda \|s\|^2 \prod_{i=1}^P \frac{1 - k_i^2}{1 - k_i'^2}$$

L'introduction du codage des gains dans la boucle est toujours impossible. Etant donnée la contribution importante du premier gain dans l'expression de l'énergie du vecteur perceptuel synthétique, on préfère poser :

$$\tilde{g}_1^2 = \lambda \|p\|^2$$

L'algorithme retenu est finalement le suivant. Toutes les fenêtres d'analyse (20 ms), on réalise une quantification scalaire non uniforme de  $\|p\|^2$  sur 5 bits. Chaque sous-fenêtre (5 ms), on teste la valeur de  $\|s\|^2$  dans cette sous-fenêtre. Si cette valeur est supérieure à un seuil ( $10^{-5}$ ), on applique l'algorithme :

- $\lambda^{\text{cod}} = \left( \frac{\tilde{g}_1^2}{\|p\|^2} \right)^{\text{cod}}$
- $\tilde{g}_1^{\text{cod}} = \sqrt{\lambda^{\text{cod}} \|p\|^2}$
- Pour  $k = 2 \dots K$
- \*  $x_k^{\text{cod}} = \left( \frac{\tilde{g}_k}{\tilde{g}_{k-1}} \right)^{\text{cod}}$
- \*  $\tilde{g}_k^{\text{cod}} = x_k^{\text{cod}} \tilde{g}_{k-1}^{\text{cod}}$

Si l'énergie est inférieure au seuil, on transmet des zéros. Cet algorithme est introduit dans la boucle de calcul.

### IV.3 Utilisation d'un dictionnaire prédictif

L'algorithme précédent ne pose pas de problème si l'on utilise un dictionnaire mixte. Par contre, si l'on recherche un premier vecteur dans un dictionnaire prédictif, lors d'une transition silence phonème, on peut très bien avoir  $\|s\|^2$  supérieur au seuil fixé, un premier gain égal à zéro et les suivants différents de zéro. On modifie l'algorithme de la façon suivante. A la 2<sup>ème</sup> étape, lorsque  $\tilde{g}_1$  est égal à zéro, on calcule  $\tilde{g}_1$  à partir de  $\tilde{g}_2$ , par exemple en écrivant que  $\tilde{g}_1 = 1.5 \tilde{g}_2$  (le choix de 1.5 correspond à la valeur la plus probable). On applique, ensuite, l'algorithme précédent. Au décodage, on récupère un premier gain différent de zéro mais cela n'est pas grave puisqu'il est associé à un vecteur nul.

## V. RESULTATS DE SIMULATION

Ces simulations sont réalisées dans les conditions suivantes. On réalise une analyse LPC d'ordre 8 sur des fenêtres de 160 échantillons par la méthode d'autocorrélation (algorithme de Schur) puis on code les Log Area Ratio sur 36 bits. Le débit nécessaire pour transmettre les coefficients du filtre de synthèse est donc de 1.8 kbit/s. L'excitation est composée de 3 vecteurs de dimension  $N = 40$ , le premier étant extrait d'un dictionnaire prédictif de 128 vecteurs et les 2 suivants d'un dictionnaire stochastique de 128 vecteurs peuplés de valeurs aléatoires gaussiennes. Toutes les 20 ms, on code l'énergie du perceptuel sur 5 bits. Toutes les 5 ms, on code le coefficient  $\lambda$  et chaque rapport des gains sur 4 bits et chaque indice sur 7 bits. Toutes les tables de codage utilisées ont été construites par l'algorithme LBG. Il faut transmettre également le signe du premier gain. Le débit de ce codeur est donc de 8.85 kbit/s. En exploitant le caractère Toeplitz du dictionnaire prédictif (décalage d'un échantillon entre 2 vecteurs voisins), on montre que l'algorithme itératif standard réclame 6.8 Mflops ( $10^6$  multiplications/accumulations flottantes par seconde). On applique cet algorithme à un signal composé de 4 doubles phrases phonétiquement équilibrées. On obtient un rapport signal/bruit segmental de 10,46 dB sans codage des gains et de 10,44 dB avec codage. La Fig. 3 donne les rapports signaux/bruits segmentaux en fonction de la complexité du traitement dans les cas suivants :

- 1 - Algorithme itératif standard.
- 2 - Algorithme avec optimisation des gains à chaque itération.
- 3 - Algorithme RMGS
- 4 - Algorithme itératif standard avec un dictionnaire mixte où l'on regroupe le dictionnaire prédictif et le dictionnaire stochastique (augmentation du débit de 600 bits/s).
- 5 - Comme 4 mais algorithme RMGS.
- 6 - Algorithme itératif standard mais on impose le caractère Toeplitz au dictionnaire stochastique.
- 7 - Algorithme itératif standard mais on supprime l'utilisation des dictionnaires filtrés et on impose le caractère Toeplitz à la matrice  $H^T H$ .
- 8 - Comme 6 mais algorithme RMGS.
- 9 - Comme 7 mais algorithme RMGS.

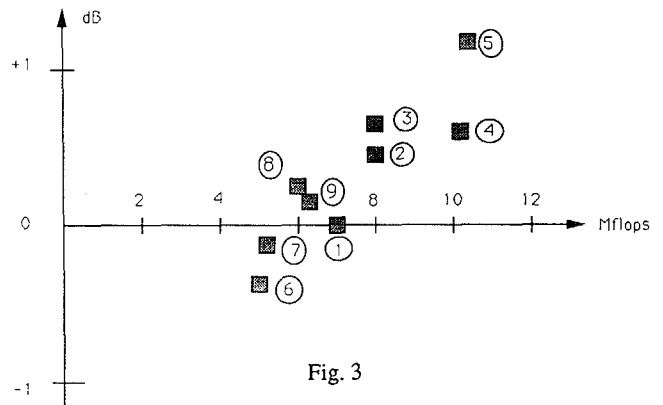


Fig. 3

## VI. REFERENCES

- [1] G.DAVIDSON, A.GERSHO "Multiple\_stage Vector Excitation Coding of Speech Waveforms" ICASSP 88
- [2] P.DYMARSKI, N.MOREAU, A.VIGIER "Optimal and sub-optimal algorithms for selecting the excitation in linear predictive coders" ICASSP 90
- [3] G.GOLUB, C.VAN LOAN "Matrix Computations" Johns Hopkins University Press, 1983 (Second Edition 1989)
- [4] E.OFER, D.MALAH, A.DEMBO "A Unified Framework for LPC Excitation Representation in Residual Speech Coders" ICASSP 89
- [5] W.VOS "Quantization of the gains in the CELP coder" Document interne TELECOM Paris, Février 1991