

# Codage échelonnable à granularité fine de la parole: Application au codeur G.729

*Djamah Mouloud O'Shaughessy Douglas*  
INRS-EMT : 800, de la Guichetière Ouest Bureau 6900,  
Montréal (Québec) H5A 1K6 Canada  
djamah@emt.inrs.ca

## RESUME

---

Cet article propose un algorithme de conception d'un quantificateur vectoriel arborescent TSVQ (tree-structured vector quantization) incorporé. Nous modifions la norme G.729 en remplaçant le quantificateur original pour la quantification des fréquences de raies spectrales LSF's (line spectral frequencies) par un quantificateur basé sur une structure arborescente. Le codeur modifié est échelonnable en débit binaire à granularité fine avec un changement graduel de la qualité du signal de parole synthétisé.

## ABSTRACT

---

### **Fine granularity scalable speech coding: Application to the G.729 coder**

This paper proposes an efficient codebook design for tree-structured vector quantization (TSVQ) which is embedded in nature. We modify the speech coding standard G.729 by replacing its original quantizer for line spectral frequencies (LSF's) with TSVQ-based quantizer. The modified coder is fine-granular bit-rate scalable with gradual change in quality for the synthetic speech.

---

MOTS-CLES : Codage de la parole, quantification vectorielle arborescente, échelonnabilité.  
KEYWORDS : Speech coding, tree-structured vector quantization, scalability.

---

## 1 Introduction

Dans certains codeurs de parole standards, l'encodeur ne génère qu'un seul type de flux-binaire à un débit binaire fixe (Chu, 2003). Toutefois, si le trafic dans le canal de transmission (réseau à paquets) est congestionné, les données codées pourraient être perdues. Ce problème peut être résolu en utilisant un flux-binaire incorporé composé d'une couche de base suivie d'une ou plusieurs couches d'amélioration qui sont utilisées pour améliorer la qualité de la parole synthétisée. Ces couches d'amélioration peuvent être écartées, une couche à la fois, lorsque le canal de transmission est congestionné. L'échelonnabilité à granularité fine (Chu, 2006; Chen et Lee, 2003) est une approche dans laquelle le flux-binaire peut être écarté avec une granularité plus fine, bit par bit dans le cas extrême, au lieu d'une couche entière. En général, les codeurs de parole qui utilisent la quantification incorporée sont échelonnables. Un quantificateur est incorporé lorsque le paramètre quantifié peut être successivement raffiné au fur et à mesure que l'indice associé est lu. Le reste du document est organisé comme suit : À la section 2, nous proposons un algorithme de conception d'un dictionnaire à structure arborescente. Les résultats expérimentaux (pour l'introduction de la caractéristique de l'échelonnabilité au codeur G.729) sont donnés à la Section 3 et la conclusion apparaît à la Section 4.

## 2 Conception d'un quantificateur vectoriel arborescent

La technique désignée par le terme *fusion de cellules* a été suggérée par (Riskin et al., 1994) pour la conception d'un quantificateur vectoriel arborescent TSVQ (tree-structured vector quantization). La procédure de conception procède du niveau le plus élevé (de l'arbre) vers les niveaux inférieurs. Récemment (Chu, 2006) a proposé un algorithme de conception d'un quantificateur arborescent multi-étage désigné par MTVQ (multistage TSVQ) pour le codage des fréquences de raies spectrales LSF's (line spectral frequencies). Nous proposons un algorithme de conception d'un quantificateur arborescent TSVQ basé sur la technique de fusion de cellule.

Considérons un quantificateur vectoriel à un seul étage avec un dictionnaire-étage de taille  $N$  (de résolution  $r = \log_2 N$ ):  $\mathbf{Y} = \{\mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_{N-1}\}$ . Pour construire un arbre binaire équilibré, les vecteurs-code  $\mathbf{y}_i, i = 0 \text{ à } N - 1$  sont placés aux positions des nœuds-feuille (le niveau le plus élevé) de l'arbre. Ceci est un procédé d'assignement d'indices (Chu, 2006) décrit par :  $\mathbf{c}_i^{(r)} = \mathbf{y}_{a[i,k]}$  ( $i = 0 \text{ à } N$ ) où la notation  $\mathbf{c}_i^{(r)}$  indique que les vecteurs-code  $\mathbf{y}_{a[i,k]}$  sont placés au niveau  $r$  ( $r = \log_2 N$ ) de l'arbre binaire.  $a[i, k] \in [0, N - 1]$  est désigné comme la séquence d'assignement d'indices avec  $k = 0 \text{ to } N! - 1$ , puisque avec  $N$  indices il y a  $N!$  permutations (séquences). Pour une séquence d'assignement d'indices donnée, le processus de fusion de cellules consiste à fusionner les cellules de résolutions plus élevées pour former les cellules de résolutions inférieures, comme suit :  $\mathbf{R}_i^{(m)} = \mathbf{R}_{2i}^{(m+1)} \cup \mathbf{R}_{2i+1}^{(m+1)}$  (pour  $i = 0 \text{ à } 2^m - 1$  et pour  $m = r - 1 \text{ à } 0$ ). Les vecteurs-code  $\mathbf{c}_i^{(m)}$  peuvent alors être calculés comme les centroïdes des cellules  $\mathbf{R}_i^{(m)}$ . Pour la mesure de performance, on peut se baser sur le critère proposé par (Chu, 2006) et qui peut être exprimé comme suit :

$$D = \sum_{m=0}^{r-1} \sum_{i=0}^{2^m-1} [P_{2i}^{(m+1)} d(\mathbf{c}_i^{(m)}, \mathbf{c}_{2i}^{(m+1)}) + P_{2i+1}^{(m+1)} d(\mathbf{c}_i^{(m)}, \mathbf{c}_{2i+1}^{(m+1)})] \quad (1)$$

où  $P_i^{(m)}$  est la probabilité du vecteur-code  $\mathbf{c}_i^{(m)}$  (définie comme la probabilité qu'un vecteur aléatoire appartienne à la cellule  $\mathbf{R}_i^{(m)}$  ayant comme centroïde le vecteur  $\mathbf{c}_i^{(m)}$ ) avec  $P_i^{(m)} = P_{2i}^{(m+1)} + P_{2i+1}^{(m+1)}$  et  $d(\mathbf{x}, \mathbf{y})$  est la distance entre les vecteurs  $\mathbf{x}$  et  $\mathbf{y}$ .

Une manière simple pour l'élaboration du dictionnaire consiste à évaluer exhaustivement toutes les séquences possibles d'assignement d'indices et de retenir la meilleure séquence qui correspond à celle qui minimise le critère (1). Cette stratégie est appelée *recherche exhaustive conjointe d'assignement d'indices* et le nombre de séquences qui doivent être évaluées est donné par (Chu, 2006) :  $N! = \prod_{i=0}^{\log_2(N/2)} [(N/2^i)! / 2^{(N/2^{i+1})!}]^{2^i}$ ,  $N \geq 2$ . Pour  $N \leq 8$ , la valeur de  $N!$  est relativement faible. Cependant, dans la pratique, la taille  $N$  du dictionnaire-étage peut être relativement élevée et la procédure de recherche exhaustive conjointe devient impraticable. Le problème peut être résolu en divisant le dictionnaire-étage de taille  $N$  en  $N/n$  sous-dictionnaires de taille  $n \leq 8$ . Pour chaque sous-dictionnaire et utilisant la recherche exhaustive conjointe, le sous-arbre optimal (selon (1)) est trouvé. La figure 1 donne un exemple de construction d'un arbre où l'ensemble des nœuds-feuille correspond à un dictionnaire-étage de taille  $N = 32$ . L'objectif ici est de construire un arbre de six niveaux: du niveau  $l = 0$  (un seul vecteur) au niveau  $l = 5$  (32 vecteurs). À partir du dictionnaire-étage de taille  $N = 32$ , quatre sous-dictionnaires de taille  $n = 8$  chacun sont trouvés; puis les quatre sous-arbres

optimaux (utilisant la recherche exhaustive conjointe) correspondants aux quatre sous-dictionnaires sont construits:  $sarbre_i^{(5,2)}$  ( $i = 0 \text{ à } 3$ ) où  $sarbre_i^{(p,q)}$  est le sous-arbre construit du niveau  $p$  au niveau  $q$  de l'arbre. Ainsi le sous-arbre  $sarbre_i^{(p,q)}$  a une hauteur de  $h = p - q$  (du niveau 0 à  $h$ ). Une fois que les sous-arbres  $sarbre_i^{(5,2)}$  ( $i = 0 \text{ à } 3$ ) sont construits, les quatre vecteurs-code  $c_i^{(2)}$  ( $i = 0 \text{ à } 3$ ) correspondants aux nœuds du niveau  $l = 2$  de l'arbre sont calculés et la procédure de recherche exhaustive conjointe peut être appliquée (puisque le nombre de vecteurs-code est faible) pour construire le sous-arbre optimal  $sarbre_0^{(2,0)}$ . La dernière étape de construction est l'opération d'assignement des indices, qui consiste à affecter les vecteurs-code aux nœuds de l'arbre binaire équilibré ainsi construit. Le procédé décrit ci-dessus peut être généralisé pour un arbre binaire de hauteur quelconque.

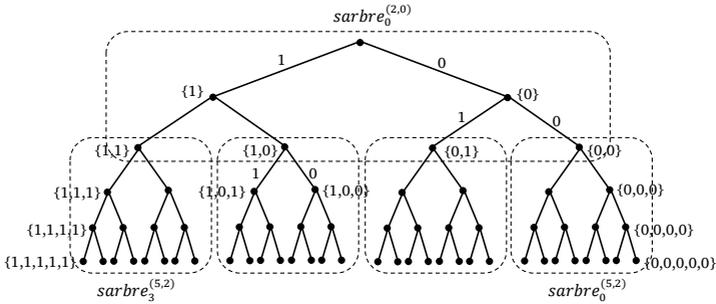


Figure 1: Exemple du principe de construction d'un arbre binaire de hauteur 5 (du niveau 0 à 5) en divisant l'arbre en sous-arbres avec une hauteur maximale de 3 (du niveau 0 à 3) chacun.

La performance de la procédure décrite ci-dessus dépend de la performance de l'algorithme qui consiste à diviser un ensemble de  $N$  vecteurs-code en  $N/n$  sous-ensembles de  $n$  vecteurs-code chacun. Nous proposons un algorithme efficace désigné sous le nom d'algorithme de regroupement. Soit  $I_i^{(n)}$  un ensemble (d'indice  $i$ ) de  $n$  indices de vecteurs-code sélectionnés parmi  $N$  vecteurs (indexés de 0 à  $N$ ). Le nombre d'ensembles d'indices distincts (les  $n$  indices contenus dans l'ensemble  $I_i^{(n)}$  doivent être différents et l'ordre des indices n'est pas pertinent) est donné par  $N2 = N!/[n!(N - n)!]$ . Nous définissons la distance associée à l'ensemble d'indices  $I_i^{(n)}$  par :

$$D[i, n] = \sum_{j \in I_i^{(n)}} P_j (\mathbf{c}_i - \mathbf{y}_j)^T \mathbf{W}(\mathbf{y}_j) (\mathbf{c}_i - \mathbf{y}_j), \quad (2)$$

où  $P_j$  est la probabilité du vecteur-code  $\mathbf{y}_j$  et  $\mathbf{W}(\mathbf{x})$  est une matrice de pondération diagonale dépendant du vecteur  $\mathbf{x}$ . En minimisant le critère (2), le centroïde  $\mathbf{c}_i$  est donné par :  $\mathbf{c}_i = [\sum_{j \in I_i^{(n)}} P_j \mathbf{W}(\mathbf{y}_j)]^{-1} [\sum_{j \in I_i^{(n)}} P_j \mathbf{W}(\mathbf{y}_j) \mathbf{y}_j]$ .

Une séquence de  $N/n$  ensembles (d'indices) disjoints est définie par  $S_k^{(n)} = \{I_{b[i,k]}^{(n)}, i = 0 \text{ à } (N/n) - 1\}$  où  $b[i, k] \in [0, N2 - 1]$  avec  $k$  est l'indice de la séquence et  $N2$  est le nombre d'ensembles d'indices distincts. Nous avons  $\cup_{i=0 \text{ à } (N/n)-1} I_{b[i,k]}^{(n)} = \{0, 1, \dots, N - 1\}$ .

Par la suite, on désignera les paramètres  $N/n$  et  $n$  par la taille et la dimension de la séquence  $S_k^{(n)}$ . Pour trouver la meilleure séquence, nous minimisons la distorsion totale :

$$D_T[k, n] = \sum_{i=0}^{(N/n)-1} D[b[i, k], n]. \quad (3)$$

Selon notre critère d'optimalité (3), la meilleure séquence peut être trouvée en faisant une recherche exhaustive sur toutes les séquences possibles qui produisent différentes valeurs de la distorsion (3). Cependant, cette recherche exhaustive est impraticable parce que le nombre de possibilités est astronomique même pour des valeurs modérées de  $N$  et  $n$ . La complexité élevée, pour la recherche de la séquence optimale, est due à deux paramètres : Le nombre d'ensembles d'indices distincts  $N2$  et le nombre de séquences distinctes. Par exemple  $N2 \approx 10^{12}$  pour  $N = 128$  et  $n = 8$ . Cependant pour  $n = 2$ , la valeur de  $N2$  est réduite à 8128. Nous considérons une procédure sous-optimale qui consiste, dans une première étape, à évaluer un nombre limité de séquences (de dimension  $n = 2$ ) pour en sélectionner les  $M_L$  séquences ayant les plus faibles distorsions. À partir de ces  $M_L$  séquences de dimension  $n = 2$ , un nombre limité de séquences de dimension  $n = 4$  sont construites et évaluées pour ne retenir que les  $M_L$  séquences de plus faibles distorsions. Le procédé est répété jusqu'à ce que la dimension désirée  $n1$  soit atteinte et que la meilleure séquence qui minimise la distorsion globale (3) soit trouvée. L'algorithme est donné à la table 1.

- 
- Entrées: les vecteurs-code  $\{y_j, j = 0 \text{ à } N - 1\}$ , les probabilités  $\{P_j, j = 0 \text{ à } N - 1\}$  et les paramètres  $n1, N_L, M_L$  ( $M_L < N_L$ )
  - 1- Initialisation:  $S_0^{(1)} = \{\{0\}, \{1\}, \dots, \{N - 1\}\}$ ,  $n \leftarrow 2$
  - 2- À partir de la séquence  $S_0^{(1)}$ , extraire un maximum de  $N_L$  séquences  $\{S_k^{(2)}, k = 0 \text{ à } N_L - 1\}$ .
  - 3- À partir des  $N_L$  séquences de l'étape 2, retenir les  $M_L$  séquences distinctes  $\{S_{k(j)}^{(2)}, j = 0 \text{ à } M_L - 1\}$  correspondantes aux  $M_L$  plus faibles valeurs de la distorsion (3).
  - 4- Pour chaque séquence  $S_{k(j)}^{(2)}$ , extraire un maximum de  $N_L$  séquences de dimension  $2n$ .  
Donc nous avons  $N_L M_L$  séquences  $\{S_k^{(2n)}, k = 0 \text{ à } N_L M_L - 1\}$ .
  - 5- À partir des  $N_L M_L$  séquences de l'étape 4, retenir les  $M_L$  séquences distinctes  $\{S_{k(j)}^{(2n)}, j = 0 \text{ à } M_L - 1\}$  correspondantes aux  $M_L$  plus faibles valeurs de la distorsion (3).
  - 6- Test:  $n \leftarrow 2n$ , si  $n = n1$ , aller à l'étape suivante, autrement répéter les étapes 4 à 5.
  - 7- Parmi les  $M_L$  séquences sélectionnées à l'étape 5, retenir la séquence  $S_K^{(n1)}$  qui minimise la distorsion globale (3) à la dimension désirée  $n1$ .
  - Sortie : la séquence  $S_K^{(n1)} = \{I_{b[i, K]}^{(n1)}, i = 0 \text{ à } (N/n1) - 1\}$   
tel que  $\cup_{i=0 \text{ à } (N/n1)-1} I_{b[i, K]}^{(n1)} = \{0, 1, \dots, N - 1\}$
- 

Table 1: Algorithme de regroupement (divise un ensemble de  $N$  vecteurs-code en  $N/n$  sous-ensembles de  $n$  vecteurs-code chacun)

Pour un quantificateur MTVQ à  $K$  étages, la structure est conçue en deux étapes : en premier lieu les  $K$  dictionnaires-étage (correspondants aux niveaux les plus élevés des  $K$  arbres binaires) sont construits et optimisés conjointement; en seconde étape, la construction de la structure MTVQ est complétée de façon séquentielle en utilisant la méthode décrite plus haut pour compléter la construction de l'arbre binaire équilibré associé à chaque étage.

L'extraction des  $N_L$  séquences de dimension  $2n$   $\{S_k^{(2n)}, k = 0 \text{ to } N_L - 1\}$  à partir d'une séquence de dimension  $n$   $\{S_k^{(n)}\}$  peut être faite en étendant et en généralisant la méthode présentée par (Chu, 2006). Soit la séquence  $S_k^{(n)} = \{I_{b[i,k]}^{(n)}, i = 0 \text{ to } (N/n) - 1\}$ ; un ensemble d'indices de dimension  $2n$  est formé par l'opération d'union entre deux ensembles (de dimension  $n$  chacun) pris parmi les ensembles de la séquence  $S_k^{(n)}$  :

$$\left\{ I_{b[i,k]}^{(n)} \cup I_{b[j,k]}^{(n)} \right\} \quad i = 0 \text{ à } \frac{N}{n} - 2, j = i + 1 \text{ à } \frac{N}{n} - 1. \quad (4)$$

Les indices  $i$  et  $j$  sont choisis de telle manière que les indices contenus dans l'ensemble de dimension  $2n$  (résultant de l'opération (4)) doivent être différents. Ainsi nous avons un total de  $N3 = (N/n)! / (2! [(N/n) - 2]!)$  ensembles d'indices distincts de dimension  $2n$   $\{I_i^{(2n)}, i = 0 \text{ à } N3 - 1\}$ . En utilisant l'équation (2), les distances  $\{D[i, 2n], i = 0 \text{ à } N3 - 1\}$  associées aux ensembles  $\{I_i^{(2n)}, i = 0 \text{ à } N3 - 1\}$  sont calculées. Les distances sont ordonnées dans l'ordre croissant, avec les ensembles associés placés dans le même ordre. Pour les ensembles d'indices ainsi ordonnés et utilisant l'ensemble ayant la plus faible distance (le premier ensemble dans la liste ordonnée) comme référence, on élimine tous les autres ensembles de la liste qui ne sont pas disjoints avec la référence. On continue le processus avec le prochain ensemble dans la liste ordonnée jusqu'au point où tous les ensembles restants dans la liste soient disjoints. À la fin du processus une seule séquence de  $N/2n$  ensembles reste et forme la première séquence de dimension  $2n$ ; Nous pouvons appliquer la même méthode à plusieurs reprises pour extraire d'autres séquences; nous recommençons avec la liste ordonnée intacte, nous ignorons le premier ensemble et utilisons le deuxième ensemble dans la liste ordonnée comme référence puis nous appliquons le même procédé pour extraire une autre séquence.

La performance de l'algorithme de regroupement peut être améliorée. Considérons une séquence  $S_k^{(n)}$  de  $N/n$  ensembles d'indices; nous pouvons permuter les indices (un indice à la fois) entre deux ensembles de la séquence  $S_k^{(n)}$ . Cette opération produit une nouvelle séquence dont la distorsion totale (3) est évaluée. L'opération de permutation est retenue si la distorsion totale est minimisée, autrement l'opération est annulée. L'opération de permutation peut être appliquée pour tous les indices d'un ensemble et pour toutes les combinaisons possibles de deux ensembles de la même séquence  $S_k^{(n)}$ . Pour réduire la complexité, la procédure est appliquée seulement à la dimension désirée  $n1$  (l'étape 7 de l'algorithme de regroupement) pour les  $M_L$  séquences de plus faibles distorsions.

### 3 Évaluation des performances

Le codeur standard G.729 (ITU-T Recommend, 2007) opère sur des trames de parole de 10 ms (80 échantillons pour une fréquence d'échantillonnage de 8 kHz), où 80 bits sont utilisés pour chaque trame (un débit de 8 kbit/s). Pour la quantification des coefficients LSF, une prédiction MA commutée de 4ème ordre est utilisée pour prédire les coefficients LSF de la trame courante. La différence entre les coefficients calculés et les coefficients prédits est quantifiée en utilisant une structure MSVQ à deux étages. Le premier étage est un quantificateur de dimension 10 utilisant un dictionnaire de 128 entrées (7 bits). Le deuxième étage est un quantificateur structuré de 10-bits, qui est implémenté sous forme d'une structure SVQ utilisant deux dictionnaires de dimension 5

chacun (les cinq coefficients LSF inférieurs et les cinq coefficients LSF supérieurs) et contenant 32 entrées (5 bits) chacun. Ainsi, un total de 18 bits par trame est utilisé pour la quantification des coefficients LSF: 17 bits sont utilisés pour quantifier les vecteurs erreur-de-prédiction de dimension 10 et un bit est utilisé pour sélectionner un des deux prédicteurs possibles. Pour concevoir le quantificateur des coefficients LSF basé sur une structure MTVQ, les structures arborescentes associés au premier dictionnaire-étage, au deuxième dictionnaire-étage (partie inférieure), et au deuxième dictionnaire-étage (partie supérieure) sont séparément générés en utilisant le procédé de conception décrit dans la Section 2 (en utilisant la distance Euclidienne). Finalement, nous obtenons un quantificateur MTVQ correspondant à trois structures TSVQ de résolutions (hauteurs) 7, 5 et 5 bits avec trois nouveaux dictionnaires-étages (de dimensions 10, 5 et 5) ayant les mêmes vecteurs-code que ceux du quantificateur original mais arrangés dans différents ordres. L'encodeur G.729 modifié utilise les nouveaux dictionnaires (ayant les mêmes vecteurs-code que ceux des dictionnaires originaux mais disposés dans différents ordres) pour produire un flux binaire. Le décodeur modifié utilise un quantificateur prédictif basé sur une structure MTVQ pour le décodage incorporé (de 0 à 17 bits) des indices des vecteurs erreurs-de-prédiction des coefficients LSF. Dans le cas où aucun bit n'est perdu, le codeur G.729 modifié a la même performance que le codeur standard.

En utilisant le codeur G.729 modifié, 194 779 vecteurs LSF tests sont encodés utilisant 17 bits, utilisant la distance euclidienne pondérée (Paliwal et al., 1993), avec les indices résultants décodés d'une façon incorporée (de 0 à 17 bits). Pour le décodage incorporé, le schéma 2 d'allocation de bits de la table 2 est utilisé. La distorsion spectrale moyenne (SD) et le pourcentage des « outliers » pour l'ensemble des données test sont tracés à la figure 2. On observe une dégradation progressive de la distorsion spectrale moyenne quand le nombre de bits est décrémenté, confirmant que le quantificateur est échelonnable bit-par-bit. La table 2 donne deux schémas d'allocation de bit pour le décodage incorporé des indices des vecteurs-code où le schéma 1 correspond à celui utilisé par (Chu, 2006). Pour les deux schémas la priorité est accordée au premier étage. Pour le schéma 2 la priorité est accordée au deuxième étage partie-inférieure, en assignant les bits disponibles à ce dernier d'abord. La figure 3 donne la note PESQ (ITU-T Recommend, 2005) pour les deux schémas d'allocation de bit de la table 2 où un signal de parole non corrompu de durée 10.67 minutes est utilisé. Comparé au schéma 1, le schéma 2 améliore la performance (en termes de note PESQ). Il est bien connu que l'oreille humaine ne peut pas résoudre les différences aux hautes fréquences avec autant de précision qu'aux basses fréquences (les composantes basses fréquences sont plus importantes); c'est-à-dire, les coefficients LSF inférieurs sont plus importants que les coefficients LSF supérieurs. En outre, les sensibilités spectrales des coefficients LSF sont localisées. Donc, lorsque le choix se présente, il serait plus efficace (lors du décodage) de distordre les cinq coefficients LSF quantifiés supérieurs (tout en maintenant les cinq coefficients inférieurs intacts) que de distordre les dix coefficients LSF en même temps. La figure 4 donne les notes PESQ obtenues pour le codeur G.729 modifié en utilisant de la parole propre et de la parole corrompue (par le bruit de voiture avec un SNR de 10, 15, 20 et 25 dB) comme signal d'entrée de durée 10.67 min. Une dégradation graduelle de la qualité est obtenue pour la parole synthétique (propre et corrompue) quand le nombre de bits disponible pour le décodage des coefficients LSF est décrémenté un-par-un. Pour la parole propre et corrompue le changement de la qualité est faible quand les cinq coefficients LSF supérieurs sont distordus (correspondant à  $m = 12$  à 16).

Nombre de bits (m)	Scheme 1	Scheme 2
17	(7, 5, 5)	(7, 5, 5)
16	(7, 5, 4)	(7, 5, 4)
15	(7, 4, 4)	(7, 5, 3)
14	(7, 4, 3)	(7, 5, 2)
13	(7, 3, 3)	(7, 5, 1)
12	(7, 3, 2)	(7, 5, 0)
11	(7, 2, 2)	(7, 4, 0)
10	(7, 2, 1)	(7, 3, 0)
9	(7, 1, 1)	(7, 2, 0)
8	(7, 1, 0)	(7, 1, 0)
7	(7, 0, 0)	(7, 0, 0)
6	(6, 0, 0)	(6, 0, 0)
5	(5, 0, 0)	(5, 0, 0)
4	(4, 0, 0)	(4, 0, 0)
3	(3, 0, 0)	(3, 0, 0)
2	(2, 0, 0)	(2, 0, 0)
1	(1, 0, 0)	(1, 0, 0)
0	(0, 0, 0)	(0, 0, 0)

Table 2: Deux schémas d'allocation de bit utilisés pour le décodage des coefficients LSF pour le codeur G.729 modifié.

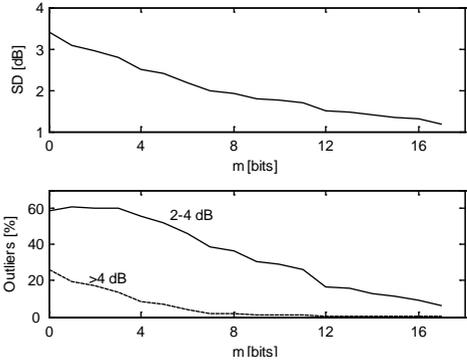


Figure 2: Résultats obtenus pour un quantificateur LSF prédictif basé sur une structure MTVQ (17 bits pour les indices). En Haut: Distorsion spectrale moyenne (SD) en fonction du nombre de bits utilisés pour le décodage utilisant le schéma 2 d'allocation de bit (Table 2). En Bas: Pourcentage de «outliers» (SD dans l'intervalle [2-4] dB et SD >4 dB) en fonction du nombre de bits utilisés pour le décodage.

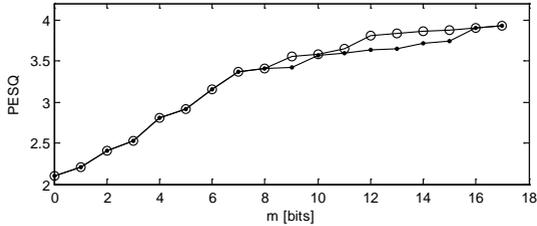


Figure 3 : La note PESQ (pour les deux schémas d'allocation de bits de la table 2) en fonction du nombre de bits utilisés pour le décodage des coefficients LSF pour le codeur G.729 modifié. De la parole non corrompue de durée 10.67 minutes est utilisée.

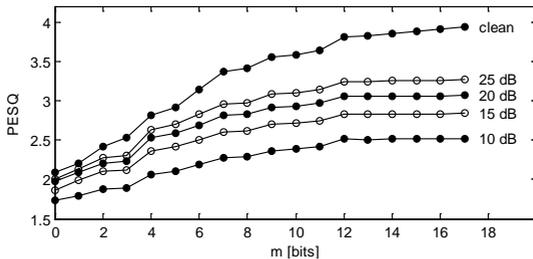


Figure 4: La note PESQ (utilisant le schéma 2 d'allocation de bit de la Table 2) en fonction du nombre de bits utilisés pour le décodage des coefficients LSF pour le codeur G.729 modifié. La parole test utilisée a une durée de 10.67 minutes. Du haut vers le bas: parole propre et parole corrompue par le bruit de voiture avec un SNR de 25, 20, 15 et 10 dB.

## 4 Conclusion

Un algorithme efficace de conception d'un quantificateur TSVQ binaire équilibré basé sur la technique de fusion de cellules est proposé. La structure arborescente est conçue depuis le niveau le plus élevé de l'arbre vers les niveaux les plus bas. L'idée principale de la méthode proposée est basée sur la construction d'un arbre binaire d'une certaine taille (certaine hauteur) comme une connexion de sous-arbres de petites tailles. Le codeur standard G.729 est modifié en remplaçant le quantificateur prédictif original basé sur une structure MSVQ (pour la quantification des coefficients LSF) par un quantificateur prédictif basé sur une structure MTVQ conçu à partir du quantificateur original. Le codeur modifié est échelonnable en débit binaire avec une granularité fine avec un changement graduel de la qualité quand le débit varie de 6300 bits/s à 8000 bits/s, avec un incrément de 100 bits/s. À 8000 bits/s le codeur modifié a la même performance (pour un canal non bruité) que le codeur standard puisque (dans ce cas) le décodeur modifié utilise les mêmes vecteurs-code que ceux du décodeur standard. L'algorithme de regroupement proposé dans ce travail peut être aussi utilisé pour concevoir des algorithmes de recherche rapides (Djamah et al. 2012)

## Références

- Chu, W. C. (2003). *Speech Coding Algorithms: Foundation and Evolution of Standardized Coders*. John Wiley & Sons.
- Chu W.C. (2006). Embedded quantization of line spectral frequencies using a multistage tree structured vector quantizer. *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 4, pages 1205-1217.
- Chen, F., Lee, I. (2003). CELP based speech coding with fine granularity scalability. *IEEE ICASSP*, pages II-145–II-148.
- Djamah, M., O'Shaughnessy, D. (2012). Fine granularity scalable speech coding using embedded tree-structured vector quantization. *Speech Communication* 54(1) pages 23-39.
- ITU-T Recommend. (2005). Perceptual Evaluation of Speech Quality (PESQ), an Objective Method for End-to-End Speech Quality Assessment of Narrow-Band Telephone Networks and Speech Codecs. *ITU*.
- ITU-T Recommend. (2007). Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP). *ITU*.
- Riskin, E., Ladner R., Wang R., Atlas L. (1994). Index Assignment for Progressive Transmission of Full-Search Vector Quantization. *IEEE Transactions on Image Processing*, vol.3, no.3, pages 307-312.
- Paliwal, K., Atal, B. S. (1993). Efficient vector quantization of LPC parameters at 24 bits/frame. *IEEE Trans Speech Audio Process.*, vol. 1, no. 1, pages 3–14.