Amélioration d'un codeur de parole à très bas débit par indexation d'unités de taille variable

Christophe Baverel[1,2], Philippe Gournay[1], Gerard Chollet[2]

[1] THALES Communication, 66 rue du fossé blanc, BP 156, 92231 Gennevilliers Cedex
christophe.baverel@fr.thalesgroup.com
philippe.gournay@fr.thalesgroup.com

[2] ENST, Département TSI, 46 rue Barrault, 75634 Paris cedex 13
baverel@tsi.enst.fr
chollet@tsi.enst.fr

Résumé – L'objectif de cet article est de démontrer la faisabilité du codage de la parole à très bas débit (environ 400 bits/s) par indexation d'unités de parole naturelles de taille variable, avec une bonne qualité de parole restituée. L'approche ALISP (Automatic Language Independent Speech Processing), telle qu'elle est décrite dans la thèse de Jan CERNOCKÝ, permet d'atteindre ces débits. Elle souffre toutefois d'un certain nombre de défauts qui limitent la qualité de la parole reproduite. Nous proposons quelques solutions alternatives, pour la segmentation du signal à coder, la recherche des unités de synthèse, et la concaténation de ces unités, pour améliorer cette qualité.

Abstract *– The aim of this paper is to demonstrate the feasibility of a very low bite rate speech coder (about 400 bits per second) by indexing natural speech units of variable size, while providing a good quality of synthesised speech. The ALISP approach (Automatic Language Independent Speech Processing), as described in the PhD thesis from Jan CERNOCKÝ, allows to reach those rates. But it suffers from a few problems, which limit the quality of the resulting speech. We propose some new solutions, with respect to the speech segmentation, the search of synthesis units, and the concatenation of these units in the decoder, in order to improve the quality.*

1. Introduction

Le codage numérique de la parole [1] est maintenant une fonctionnalité présente dans la majorité des systèmes de communication. Dans le domaine grand public, la norme GSM 6.10 à 13 kbits/s est probablement la plus connue et la plus utilisée. Il existe en fait un grand nombre de normes, qui possèdent chacune des caractéristiques différentes (débit, délai algorithmique, complexité...). Pour des applications qui nécessitent un très faible débit binaire (visiophonie, téléphonie sur IP, communications par satellite), il existe par exemple la norme G723.1 à 5.3 kbits/s. Le plus bas débit normalisé se trouve actuellement dans le domaine militaire : il s'agit du standard OTAN STANAG 4479, qui décrit une technique de codage de la parole à 800 bits/s. Ce codeur est particulièrement efficace dans des conditions opérationnelles difficiles (par exemple liaisons HF fortement perturbées), mais la qualité de la parole restituée est assez limitée. Il subsiste toutefois un certain nombre d'applications pour lesquels un débit encore inférieur serait souhaitable.

La plupart des travaux dans le domaine du codage de la parole à très bas débit (moins de 600 bits par seconde) portent sur des codeurs phonétiques, qui nécessitent une transcription complète de la base de donnée de parole enregistrée, utilisée lors de l'apprentissage. Cette opération manuelle est coûteuse et potentiellement source d'erreurs. Une autre technique de codage de la parole, par indexation d'unités de parole de taille variable déterminées de façon automatique (approche ALISP), a été proposée récemment [2]. Cette technique, améliorée dans le cadre du projet SYMPATEX[1] (SYstème de Messagerie unifiée PArole et TEXte), combine des principes de reconnaissance et de synthèse de la parole.

La technique telle qu'elle est décrite dans la référence [2] souffre toutefois d'un certain nombre de limitations qui sont rappelées ci-dessous. L'objectif de cet article est :

- d'une part, de démontrer la faisabilité du codage de la parole à très bas débit (environ 400 bits/s) par indexation d'unités de parole de taille variable, avec une bonne qualité de parole restituée.
- d'autre part, de contribuer à la recherche de solutions alternatives, pour la segmentation du signal à coder, la recherche des unités de synthèse, et la concaténation de ces unités au niveau du décodeur.

2. L'approche ALISP

2.1 Principe général

Lors de la phase d'apprentissage, une procédure automatique de segmentation (décomposition temporelle DT) et d'étiquetage (quantification vectorielle des vecteurs cibles de la DT) détermine un ensemble de 64 classes d'unités acoustiques. A chaque classe est associé un ensemble de 8 représentants issus de la base de donnée de parole d'apprentissage. Lors de la phase de codage d'un signal, une procédure de reconnaissance détermine la succession d'unités acoustiques et identifie le "meilleur" représentant à utiliser en synthèse. On transmet alors au décodeur le numéro de la classe acoustique reconnue, ainsi que l'indice de cette unité représentante. La synthèse (décodage) de la parole se fait par concaténation des représentants, éventuellement en utilisant un synthétiseur paramétrique de type LPC.

[1] Projet labellisé par le RNRT (Réseau National de Recherche en Télécommunications - Ministère français de la Recherche) en 1999 sous le numéro 76.

2.1 Les limitations

Cette approche a permis la mise en œuvre d'un système de codage complet à 400 bits/s [6], ce qui est assez proche du débit phonétique (environ 50 bits/s). Mais malgré tout, cette méthode comporte quelques inconvénients [3] :

1. Dans le système existant, la segmentation initiale de la base d'apprentissage se fait par DT. Cette technique permet de décrire l'évolution des paramètres spectraux sous la forme d'une succession de vecteurs cibles reliés par des fonctions d'interpolation. Elle présente l'inconvénient majeur de segmenter le signal de parole au niveau des zones instables (donc les plus difficilement concaténables lors de la synthèse).
2. L'analyse / synthèse de la parole se fait par l'intermédiaire d'un modèle paramétrique simple de type LPC. Ce modèle introduit de nombreuses dégradations, même en l'absence de tout codage des paramètres. Un système d'analyse / synthèse plus élaboré (de type "harmonique plus bruit" ou HNM [4]) permet d'améliorer le naturel de la parole codée.
3. Les unités représentantes sont simplement les 8 unités les plus longues appartenant à la même classe acoustique. Il est vraisemblablement possible d'améliorer les performances du codeur, à la fois en termes de débit et de qualité de la parole codée, en optimisant le choix de ces unités représentantes.

2.3 L'objectif de notre travail

L'objectif de notre travail était de rechercher un certain nombre des solutions alternatives afin de corriger ces défauts. Il s'agit plus précisément :

1. Dans un premier temps, de démontrer la faisabilité du codage par indexation d'unités de parole de taille variable, si nécessaire en effectuant certaines étapes manuellement (en s'aidant par exemple du spectrogramme) et en se basant sur la transcription phonétique exacte de la phrase à coder et de la base d'apprentissage.
2. D'automatiser progressivement ces différentes étapes, en cherchant avant tout à se passer de la transcription phonétique. On pourra par exemple tenir compte des paramètres prosodiques du signal de parole lors de la segmentation.
3. D'introduire un procédé d'analyse / synthèse HNM.

3. Démonstration de faisabilité

3.1 Un nouveau principe de segmentation

Le système de codage fonctionne par concaténation de segments de parole pris dans une base de données de parole préalablement enregistrée. Le système précédent proposait des segments ALISP pouvant (dans une certaine mesure) être mis en correspondance avec les phones du langage considéré. Ceci avait pour avantage de créer un dictionnaire de segments de taille réduite, mais la concaténation de deux segments successifs s'effectuant sur des zones de transition entre phones, la qualité du signal synthétisé était fortement dégradée, à cause d'un fort bruit de transition.

Il nous a paru plus judicieux de segmenter notre base de données sur les zones de stabilité, c'est à dire les zones où le spectre du signal a une allure stable (voyelles, fricatives, nasales…) ou lorsque l'énergie du signal est quasi nulle (silence, occlusions...). Ceci nous permettra en synthèse de concaténer deux segments qui auront un spectre identique.

Aux instants fortement harmoniques, les deux spectres vont se correspondre, ayant le même fondamental (le pitch est codé à part) et des amplitudes harmoniques équivalentes. Aux instants fortement bruités, les deux spectres de bruit seront semblables et le bruit de concaténation sera noyé. Et aux instants où l'énergie du signal sera faible, le bruit de transition sera également inaudible.

3.2 Expérience 1 : Synthèse par concaténation des formes d'onde

Une première expérience, très simple, a été tentée en segmentant une phrase de test, appartenant à une base de données phonétiquement transcrite, de façon manuelle, en respectant les critères définis dans le §2.2, puis en recherchant manuellement dans la base de données les morceaux qui avaient la même transcription et parmi ceux ci, celui qui lui ressemblait le plus en terme de durée, d'énergie et de fréquence. La phrase de sortie fut crée en concaténant les morceaux trouvés, les uns aux autres, sans aucun traitement.

Celle-ci souffre des problèmes de continuité de pitch et d'énergie, mais la transition entre les différents segments est très satisfaisante, ce qui montre l'intérêt de ce nouveau mode de segmentation et surtout démontre la faisabilité du système.

4. Amélioration de l'analyse / synthèse

4.1 L'analyse / synthèse HNM

Dans le système de codage initial, l'analyse / synthèse du fichier de parole était celle d'un simple vocodeur LPC-10, ce qui introduisait à la base une mauvaise synthèse. Nous avons choisi d'utiliser la méthode HNM [4]. Ce modèle décrit la parole comme la somme d'une partie harmonique et d'une partie bruitée : toutes les 5 ms, on recherche le pitch et l'énergie de ce signal sur une fenêtre de 20 ms, ainsi que les enveloppes spectrales associées aux parties harmoniques et bruitées, accompagnées de leurs gains.

Les informations analysées se décomposent en deux classes : la prosodie qui comprend notamment le pitch et l'énergie (et accessoirement les deux gains qui peuvent être déduit de l'énergie), et le timbre qui est composé des deux enveloppes spectrales. La prosodie est propre à chaque phrase, elle sera donc transmise intégralement après codage, alors que le timbre est relatif aux phones : il sera enregistré au niveau du codeur et du décodeur, seul son indice dans la base de données sera transmis.

Pour chaque segment de la phrase à coder, nous allons rechercher celui qui lui est le plus proche en terme de spectre dans la base de données du codeur. Un nouveau segment de synthèse est constitué en se servant de la prosodie du premier segment et du timbre du second, appartenant au décodeur, temporellement aligné par DTW (Dynamic Time Warping). En concaténant les différentes unités de synthèse, nous obtenons un nouveau signal codé par HNM.

4.2 Expérience 2 : Synthèse HNM

Pour obtenir une continuité du pitch et de l'énergie remarqués précédemment, nous avons introduit une analyse synthèse HNM dans le système de codage. Toujours en segmentant la base de données manuellement à partir des mêmes critères et possédant une sélection d'unités de synthèse ayant la même transcription phonétique, on recherche automatiquement parmi ces unités la meilleure grâce à une DTW.

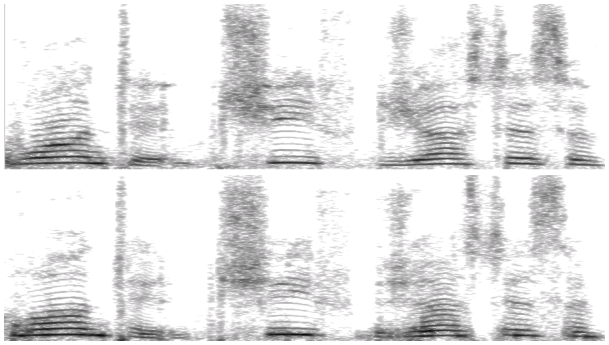

FIG. 1 : Spectrogrammes HNM avec et sans substitution des segments

Le premier spectrogramme de cette figure correspond au résultat d'une analyse synthèse simple par HNM de la phrase à coder, alors que le second est le résultat de cette dernière opération. La prosodie de la phrase synthétisée est exactement celle de la phrase de test et le timbre ne souffre pas de discontinuité aux zones de concaténation. De ce fait, la qualité de la phrase est très peu dégradée par rapport à une phrase analysée et synthétisée par HNM.

5. Segmentation automatique

5.1 Principe de la segmentation

Le signal de parole de notre base de données est échantillonné à 16kHz. Une analyse LPC d'ordre P=16 est effectuée toutes les 5ms sur des fenêtres de 20ms. Les coefficients de prédiction sont convertis en coefficients cepstraux en appliquant la formule suivante pour $n \in [0..P]$:

$$c_n = -a_n - \frac{1}{n}\sum_{k=1}^{n-1} k c_k a_{n-k}$$

Ces coefficients sont très utiles pour comparer deux prédicteurs LPC. En effet, la distance euclidienne entre les vecteurs de coefficients cepstraux est une approximation de la distance RMS log spectrale, la distance exacte (racine carré de l'intégrale de la différence entre les logarithmes des spectres de puissance) étant obtenue lorsque la séquence de coefficients cepstraux est prolongée à l'infini.

Le calcul de la distance euclidienne entre les vecteurs cepstraux successifs va donner une courbe dite de stabilité cepstrale. Les zones de stabilité vont alors correspondre aux zones où le signal sera stable spectralement, c'est-à-dire les zones où la distorsion spectrale sera la plus faible possible. Une recherche des minimums locaux sur la courbe de stabilité déterminera les instants possibles de segmentation du signal de parole considéré, désignés par des croix sur la seconde partie de la figure 2.

Notons également que certains critères pour les minimums locaux doivent être respectés : on définit un temps minimum entre deux minimums successifs, 50ms, et un seuil maximal pour la valeur du minimum considéré.

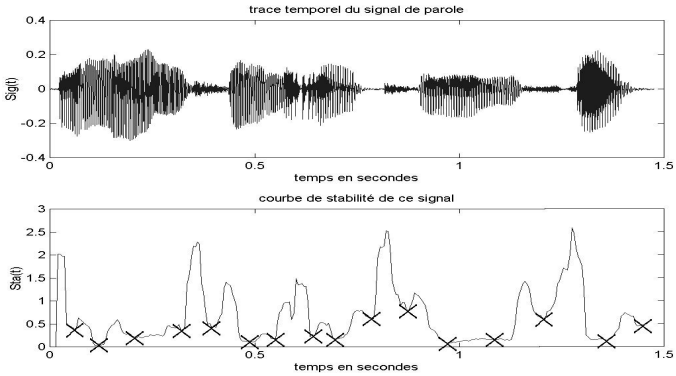

FIG. 2 : Courbe de stabilité cepstrale

5.2 Statistique sur les longueurs des segments

Nos unités constituées commencent et finissent sur des zones de stabilité, donc commencent généralement au milieu d'un phone et finissent au milieu d'un autre phone : ces unités seront alors à cheval sur au moins deux phones, parfois plus si le phone intermédiaire est totalement instable (par exemple certaines consonnes comme le L ou le R). Elles porteront donc le nom de « polyson ». Inversement, certains phones peuvent être au contraire découpés en plusieurs endroits.

Un locuteur parlant à une vitesse moyenne prononce une dizaine de phones par seconde. Comme un polyson comporte généralement 2 moitiés de phones, la segmentation va produire une dizaine de cibles par secondes. Pour cette étude, la base de données utilisée est une locutrice de la Boston University Corpus [5], prononçant une trentaine de minutes de discours.

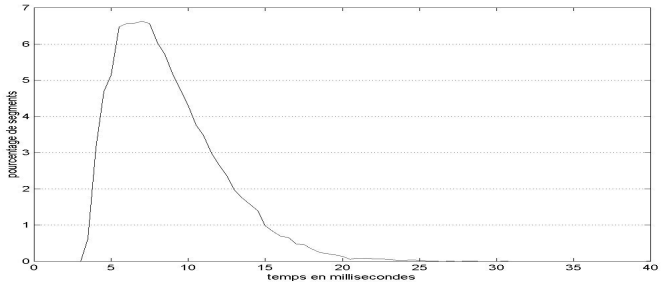

FIG. 3 : Répartition des unités segmentées

La segmentation de cette base de données, de durée totale 1805,6 secondes a produit 20894 cibles, soit environ 11,6 par seconde, ce qui est conforme à nos souhaits. La figure 3 présente l'histogramme de la taille des segments créés, ce qui permet d'avoir une idée de la répartition de ces segments en fonction de leur durée.

5.3 Expérience 3 : Segmentation automatique

Pour cette dernière expérience utilisant la transcription phonétique de la base de données, partant de l'expérience du §3.2, nous avons réalisé une segmentation automatique de la base. Notre méthode de segmentation automatique apporte très peu de dégradation, ce qui montre son efficacité.

Nous avons toutefois constaté que, lors de la création d'un nouveau segment, le chemin de DTW peut mettre en correspondance une trame voisée avec une trame non voisée ou le contraire., ce qui change l'allure du spectre et se traduit par quelques artéfacts. Ce problème intervenant régulièrement, il serait utile de le corriger soit en posant une contrainte sur le chemin de DTW pour éviter de mettre en correspondance deux trames de voisement différent, soit n'utilisant qu'une seule enveloppe spectrale pour modéliser le signal.

Toute la chaîne de codage est actuellement automatisée, la qualité s'est grandement améliorée : au lieu d'une synthèse métallique et d'un fort bruit de transition, il ne reste plus qu'un léger bruit de fond caractéristique du synthétiseur HNM, qui ne gêne ni la compréhension du message, ni l'identification du locuteur. Il ne reste plus qu'à s'affranchir de l'aide de la transcription phonétique.

6. Classification des segments

Dans le système final, il sera nécessaire de classer les segments pour deux raison : tout d'abord parce que mettre des étiquettes sur les segments de la base de données d'apprentissage et de la phrase à coder permettra de se passer de la transcription phonétique. Ensuite, parce qu'un dictionnaire réduit d'unités de synthèse (N représentants par classe) pourra être créé, afin de diminuer la complexité du codeur / décodeur (coût de calcul et espace mémoire).

6.1 Expérience 4 : Recherche exhaustive dans la base d'apprentissage

Avant de créer un dictionnaire réduit de segments, nous avons voulu voir ce que donnerait le fait de parcourir l'ensemble des segments de la base pour rechercher le meilleur possible était significatif. En terme de complexité de calcul et d'espace disque pour sauvegarder tous les segments, cette manipulation n'est pas du tout intéressante pour le codage de la parole, mais elle permet de commencer l'étude sur une classification des segments de manière simple.

Le fichier son résultant s'est révélé parfaitement intelligible, mais on remarque à nouveau un certain nombre d'incohérences de voisement, qu'il faudra corriger par une meilleure recherche du plus proche segment.

6.2 Expérience 5 : Etiquetage ALISP ou phonétique

La suite de cette expérience serait de s'aider d'une transcription de la base de données et de la phrase à coder pour mettre en correspondance deux segments. Plusieurs pistes s'offrent à nous : la reconnaissance à partir des unités ALISP ou un reconnaisseur phonétique : ces deux systèmes attribuent des étiquettes sur segments de paroles. Faute de temps, nous n'avons pas encore réalisé cette expérience.

6.3 Débits attendus

Etant donné que la base de données sera construite avec vraisemblablement 4096 classes de polysons (soit 2^{12}), chaque classe comprenant 4 éléments (soit 2^2), cela signifie que 14 bits seront nécessaires pour coder le timbre d'un segment. A raison de 11 segments par seconde, le codage du timbre d'un signal de parole nécessitera quelques 160bits/s.

Par contre, pour la prosodie, seul le pitch et l'énergie totale seront transmis : les deux gains nécessaires aux filtres de synthèse sont déduits des gains du segment de la base de données et du rapport des énergies. Le codage de cette prosodie nécessite quant à lui environ 200bits/s [6].

Le codeur pourra donc en principe fonctionner aux alentours de 400 bit/s. Ce débit est identique à celui du codeur ALISP d'origine, mais avec une qualité attendue pour la parole reproduite bien supérieure.

7. Conclusion

A partir d'une première expérience faite entièrement manuellement, nous avons démontré la faisabilité d'un codeur par indexation et concaténation d'unité de synthèse de taille variable, fonctionnant avec un débit de transmission de l'ordre de 400 bits/s.

Les dernières étapes pour finir ce codeur à très bas débit est la manière de mettre en correspondance des segments ayant un timbre et un voisement identiques, puis la création d'un dictionnaire d'unités de synthèse, facilitant cette mise en correspondance par un étiquetage.

La base de données étant monolucutrice, l'aspect multilocuteur devra aussi être pris en compte, par exemple en combinant une base de données multilocuteur et une technique de transformation de locuteurs.

8. Références

[1] G. BAUDOIN, J. CERNOCKÝ, P. GOURNAY, G. CHOLLET. « Codage de la parole à bas et à très bas débit ». *Annales des Télécommunications, 55, n° 9-10, 2000*.

[2] J. CERNOCKÝ. « Speech Processing Using Automatically Derived Segmental Units : Applications to very Low Rate Coding and Speaker Verification ». Thèse de Doctorat, Université Paris XI Orsay, Décembre 1998.

[3] J. CERNOCKÝ, G. BAUDOIN, G. CHOLLET. « Unsupervised Learning for Very Low Bit Rate Speech Coding ». *In Proc. SCI'2000*, Orlando, Juillet 2000.

[4] I. STYLIANOU. « Modèles harmoniques plus bruit combinés avec des méthodes statistiques pour la transformation de la parole et du locuteur ». Thèse de Doctorat, ENST Paris, Juin 1996.

[5] M. OSTENDORF, P.J. PRICE et S. SHATTUCK-HUFNAGEL « The Boston University radio news corpus ». *Technical report, Boston university*, February 1995.

[6] Y.-P. NAKACHE, P. GOURNAY, G. BAUDOIN. « Codage de la prosodie pour un codeur de parole à très bas débit par indexation d'unités de taille variable ». *Journées CORESA 2000* des 19 et 20 octobre 2000.