Codage de la prosodie pour un codeur de parole à très bas débit par indexation d'unités de taille variable

Yves-Paul Nakache[(1)], Philippe Gournay[(2)], Geneviève Baudoin[(3)].
ypn@free.fr , philippe.gournay@tcc.thomson-csf.com, baudoing@esiee.fr

[(1)]Ecole Supérieure d'Ingénieurs en Electrotechnique et Electronique
BP 99 93162 Noisy Le Grand CEDEX

[(2)]Thomson-CSF Communications, 66 rue du Fossé Blanc,
BP 156, 92231 Gennevilliers CEDEX

[(3)]Département Signaux et Télécommunications, ESIEE
BP 99 93162 Noisy Le Grand CEDEX

Cet article décrit la manière dont ont été codés les paramètres prosodiques d'un codeur de parole à très bas débit (400 bits/s) combinant reconnaissance et synthèse vocale. Ce codeur utilise des segments de parole de longueurs variables. Les classes d'unités acoustiques sont déterminées de façon automatique. Le codage de la prosodie repose principalement sur l'utilisation de la segmentation obtenue lors du codage de l'enveloppe spectrale et exploite les caractéristiques prosodiques communes des segments codés par une même unité représentante.

This paper describes how to code prosodic parameters of a very low bit rate speech coder (400 bps). This coder combines speech recognition and synthesis, using segments of speech which have variable lengths. The classes of acoustic units are found by an automatic process. The coding of prosody is based mainly on the segmentation, which is obtained by coding of the spectral shape. Similar prosodic characteristics of the segments that are coded by the same representative units are used by this coder of prosody.

Introduction

Le procédé de codage de la parole mis en œuvre à bas débit [2] (de l'ordre de 2400 bits/s) est généralement celui du vocodeur, qui se base sur un modèle complètement paramétrique du signal de parole. Il comporte une information de voisement, qui décrit le caractère périodique ou aléatoire du signal, la fréquence fondamentale ou PITCH pour les sons voisés, l'évolution temporelle de l'énergie, et l'enveloppe spectrale du signal (généralement modélisée par un filtre LPC). Ces paramètres sont déterminés trame par trame, typiquement toutes les 10 à 30 ms. Au niveau du décodeur, une procédure de synthèse reproduit le signal de parole à partir de la valeur quantifiée des paramètres du modèle.

Jusqu'à présent, le plus bas débit normalisé pour un codeur de parole utilisant cette technique est de 800 bits/s. Ce codeur, normalisé en 1994, est décrit par le standard OTAN STANAG 4479 [4]. Il repose sur une technique d'analyse trame par trame (22.5 ms) de type LPC10 (Linear Predictive Coding) mais exploite au maximum la redondance temporelle du signal de parole en regroupant les trames 3 par 3 avant encodage des paramètres.

Cependant la qualité et l'intelligibilité de la parole reproduite par ces techniques de codage ne sont plus acceptables à partir du moment où le débit est inférieur à 600 bits/s. Pour réduire davantage le débit nous devons nous tourner vers les vocodeurs segmentaux de type *phonétiques* (avec des segments de durée variable).

Dans cet autre type de vocodeur, la procédure d'encodage est essentiellement un système de reconnaissance automatique de la parole en flot continu, qui segmente et étiquète le signal de parole selon un certain nombre d'unités de parole de taille variable. Ces unités phonétiques sont codées par indexation dans un petit dictionnaire. Le décodage repose sur le principe de la synthèse de la parole par concaténation, à partir de l'index des unités phonétiques et de la prosodie.
Cependant, le développement des codeurs phonétiques nécessite des connaissances importantes en phonétique et en linguistique, ainsi qu'une phase de transcription phonétique d'une base de données d'apprentissage, qui est financièrement coûteuse et potentiellement source d'erreurs. Pour cette raison, Les codeurs phonétiques ne peuvent que difficilement s'adapter à une nouvelle langue ou un nouveau locuteur.

Une dernière technique [1, 2, 3], proposée récemment par l'ENST Paris, l'ESIEE et l'Université de Brno (république tchèque) et développée dans le cadre du projet SYMPATEX[1] (SYstème de Messagerie unifiée PArole et TEXte), permet de contourner les problèmes liés à cette transcription phonétique de la base de données d'apprentissage en déterminant les unités de parole de façon automatique et indépendamment de la langue.

I Le codage de la parole par indexation d'unités de taille variable

Le fonctionnement de ce codeur se décompose en deux étapes. Lors de l'étape d'apprentissage, une procédure automatique détermine un ensemble de 64 classes d'unités acoustiques. A chacune de ces classes est associé un modèle statistique (modèle de Markov : HMM) ainsi qu'un petit nombre d'unités représentantes. Dans le système actuel, les unités représentantes sont simplement les 8 unités les plus longues appartenant à la même classe acoustique. Lors du codage d'un signal (fig.1), une procédure de reconnaissance (algorithme de Viterbi) détermine la succession d'unités acoustiques et identifie le "meilleur" représentant à utiliser en synthèse. Ce choix se fait en utilisant un critère de distance spectrale en utilisant un algorithme de DTW (Dynamic Time Warping). On transmet 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.

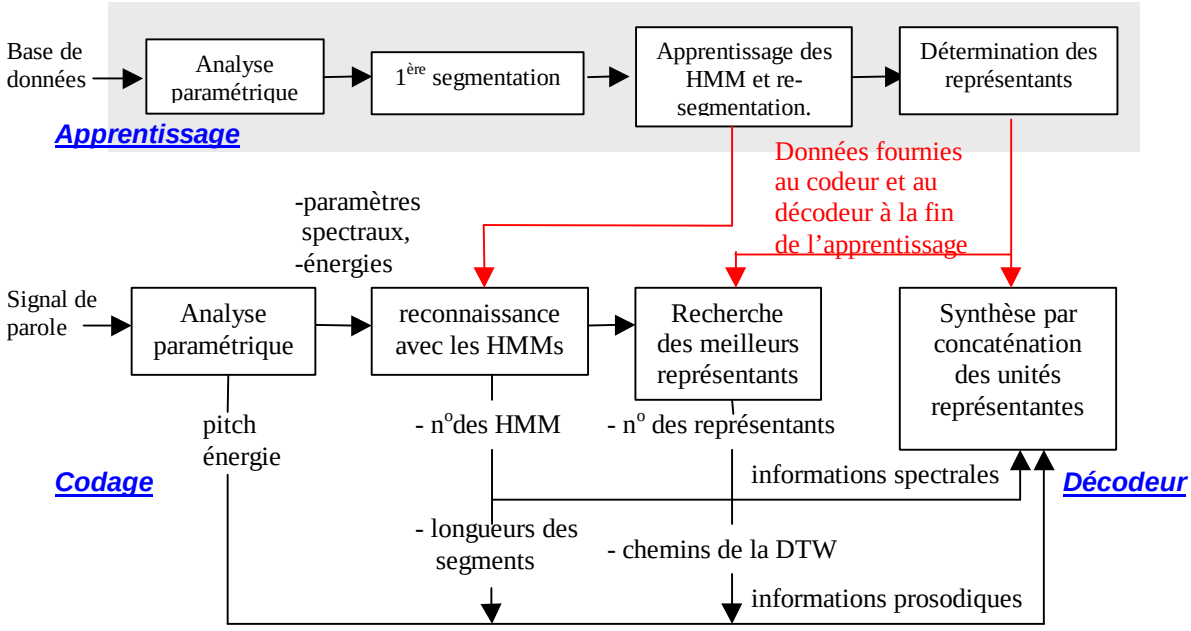

Fig.1 Schéma du codage, décodage et apprentissage

[1] Projet labellisé par le RNRT (Réseau National de Recherche en Télécommunications) en 1999 sous le numéro 76, voir site WEB http://www.telecom.gouv.fr/rnrt/index_net.htm

Cette technique permet le codage de l'enveloppe spectrale du signal en 185 bits/s environ pour un système monolocuteur, pour une moyenne d'environ vingt et un segments par seconde. L'objectif de cet article est de présenter une technique de codage de la prosodie pour un codeur monolocuteur complet à 400 bits/s. Nous avons utilisé pour nos travaux une locutrice de la base de données anglophone "Boston University Corpus".

Nous allons tout d'abord mettre en évidence dans le chapitre suivant le rôle des représentants pour le codage de la prosodie et plus particulièrement de l'énergie. Le codage des autres paramètres prosodiques sera détaillé dans les chapitres III, IV, V et VI. Enfin, dans le dernier chapitre, nous discuterons brièvement des résultats et des améliorations qu'il est possible d'apporter au codeur

II Utilisation des représentants pour le codage de l'énergie

Le terme "prosodie" regroupe principalement l'énergie du signal, le pitch et éventuellement une information de voisement. Pour pouvoir effectuer l'alignement temporel des représentants nous avons aussi besoin des chemins de la DTW. Il faudrait donc transmettre la longueur des segments puis le chemin sur chaque segment.

Pour arriver à atteindre de très bas débits, nous avons constaté qu'il était nécessaire d'utiliser le point fort du codeur : la segmentation. Puisque nous avons utilisé une segmentation acoustique automatique pour regrouper des trames ayant des propriétés spectrales communes, nous pouvons nous demander si des informations prosodiques ne peuvent pas aussi être déduites de cette segmentation.

A partir des unités acoustiques déterminées pendant la phase d'apprentissage du codeur, nous avons classé puis analysé tous les segments de la base de données. Pour chaque classe de segments ainsi constituée, il se dégage une certaine cohérence dans la forme des contours des énergies. Nous relevons de plus des ressemblances frappantes entre les contours d'énergies des représentants alignés par DTW et les contours de l'énergie du signal à coder (fig.2).

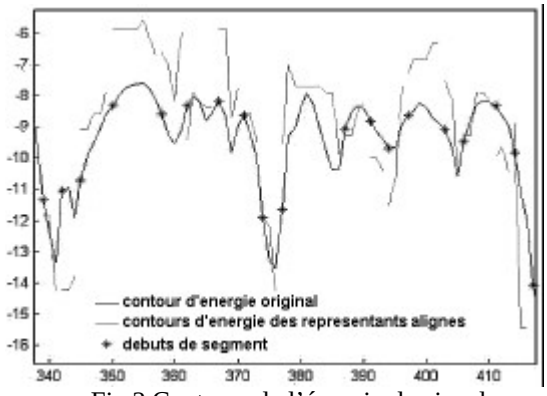

Fig.2 Contours de l'énergie du signal *à coder et des représentants alignés*

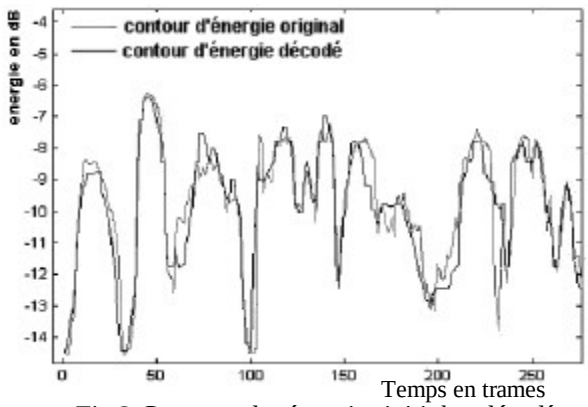

*Fig.3 Contours des énergies initial et décod*é

Nous avons fait le choix d'utiliser les contours d'énergie des représentants comme base de travail pour reconstruire le contour énergétique du signal à coder. Après avoir translaté les contours d'énergie des représentants en se référant à la première énergie de chaque segment, nous modifions ensuite la pente des contours de manière à ce qu'elle relie la première énergie du segments suivant. Nous récupérons de cette manière les variations du contour de l'énergie du signal (fig.3).

En codant les énergies de début de chaque segment sur 4 bits nous obtenons pour ce codage segmental un débit de 80 bits/s. L'élaboration du dictionnaire de représentants devrait tenir compte du contour énergétique du signal pour aboutir à un codage de l'énergie de bonne qualité.

III Codage de la longueur des segments

Le chapitre II montre comment il est possible d'utiliser la segmentation du signal pour coder l'énergie. Cependant, pour pouvoir tirer parti de cette segmentation, il faut évidemment transmettre la longueur des segments. Nous avons en moyenne 21 segments par seconde lorsque nous travaillons avec notre locutrice anglosaxonne. La taille des segments varie en fonction de la classe d'unités acoustiques et du représentant. Il apparaît que, pour la majorité des unités acoustiques, le nombre de segments en fonction de leur longueur x décroît en $1/x^{2.6}$. Il semble donc intéressant de chercher à coder ces valeurs en utilisant un code à longueur variable tel que celui de Huffman.

En utilisant les longs mots de code pour coder les longueurs des grands segments, le débit, s'il n'est pas vraiment constant, reste compris dans une plage de variation limitée. En effet, ces longs segments réduisent le nombre de segment par seconde et donc le nombre de longueurs à coder. Le débit obtenu pour le codage des longueurs des segments est de l'ordre de 55 bits/s avec un codage de type Huffman.

Pour voir s'il était possible d'améliorer encore les performances du codage des longueurs des segments, nous avons effectué une analyse des longueurs des segments selon leur numéro de représentant. Nous avons constaté que pour certains représentants la taille des segments varie peu et qu'elle est dans la plupart des cas égale à 3 trames. En tenant compte de cette propriété, il serait possible de ne plus coder la longueur de ces segments moyennant une contrainte lors de l'apprentissage et de la reconnaissance pour la fixer à 3 trames. Enfin il faudrait intervenir sur la phase de segmentation pour limiter la longueur des grands segments.

IV Codage du voisement

Le modèle paramétrique (analyse / synthèse) mis en œuvre dans le système actuel comporte une information de voisement binaire (voisé / non voisé). le signal est voisé lorsqu'il est possible de déterminer un pitch. Il arrive parfois que les limites des zones voisées se situent aux frontières des segments ; malheureusement, dans la plupart des cas, le voisement change en plein milieu d'un segment.

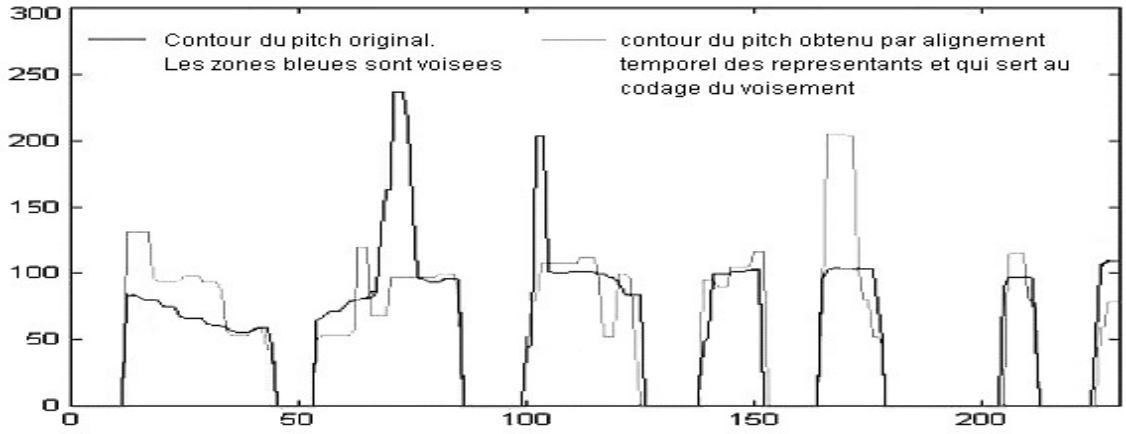

Fig.4 Information de voisement récupérable à partir du pitch des représentants

Nous avons d'abord remarqué que le voisement du fichier original est très similaire à celui obtenu simplement en alignant les contours des pitchs des représentants (fig.4).
Nous avons également remarqué que les extrémités des zones voisées de ces deux signaux différent rarement de plus de trois trames. Ces décalages sont de 1, 2 voire 3 trames à droite ou à gauche de l'extrémité de la zone voisée dans 90% des cas. Nous avons donc choisi de nous baser sur le voisement des représentants pour coder l'information de voisement, puis de la corriger de la manière suivante : Lorsque nous détectons une extrémité d'une zone de voisement sur les représentants choisis pour la synthèse nous allons apporter une information complémentaire au décodeur : cette information est le décalage qu'il faut apporter à cette extrémité, exprimé en nombre de trames, pour obtenir la position exacte de l'extrémité de voisement du signal de parole original. Nous utilisons pour cela un code de longueur variable.

Avec cette méthode, il est possible de coder l'information de voisement sur environ 22 bits par seconde.

En étudiant l'information de voisement des représentants, nous avons constaté que certains d'entre eux codaient des segments qui présentaient la particularité d'être tous entièrement voisés tandis que d'autres codaient des segments qui étaient tous entièrement non-voisés. Lors du décodage du voisement il est possible de commettre une erreur et de décaler l'information de voisement « d'une extrémité ». La conséquence serait d'inverser l'information de voisement que l'on voulait transmettre. Pour éviter cette erreur il suffirait de regarder quels sont les segments situés dans une zone considérée par le décodeur comme étant voisée et de vérifier que parmi les segments qui la composent y figurent des représentants qui ne codent que des segments entièrement voisés et qu'il n'y a aucun représentant codant des segments entièrement non voisés. Nous avons ainsi un moyen naturel de vérification de l'information de voisement transmise. Il faudrait donc d'une part optimiser le codage du voisement en créant une table connue du décodeur pour y répertorier les représentants en catégories : totalement voisés, totalement non voisés, autres, et d'autre part favoriser l'utilisation de représentants renforçant cette classification segmentale « voisé/non voisé ».

V Codage du pitch par approximation linéaire

En modélisant par un unique segment de droite l'évolution du pitch sur chacune des zones voisées du signal, il serait possible de n'envoyer au décodeur que le début et la fin de chaque zones voisées ainsi que les valeurs correspondantes du pitch. Sur nos fichiers de parole nous n'avons en moyenne que 3 ou 4 zones voisées par seconde de parole. Cette approche mono-segmentale présenterait l'avantage de minimiser à la fois le nombre de bits utilisés et la complexité algorithmique ; toutefois, dans la majorité des cas elle ne permettrait pas de transmettre fidèlement l'intonation du locuteur.

Pour rendre compte plus fidèlement de ces variations, il faut transmettre plusieurs valeurs de pitch par zone voisée. Nous allons une fois de plus profiter des caractéristiques de notre codeur pour approximer le contour du pitch par une succession de segments linéaires tout en limitant le débit.

Effectuer cette approximation linéaire revient à savoir quel est le nombre de segments nécessaire, comment déterminer les débuts et les fins de ces segments, et quel critère il faut utiliser pour minimiser les données à transmettre au décodeur. Nous avons en fait simplifié ce problème en nous limitant à la recherche d'un sous-ensemble de points, parmi N possibles, permettant d'approximer une courbe par approximation linéaire.

De la même manière que pour l'énergie, nous n'allons utiliser que les débuts de segment comme emplacement potentiel de valeurs de pitch à transmettre. Nous codons le contour du pitch zone voisée par zone voisée. Les N points dont nous parlions sont donc les N débuts de segments constituant la zone voisée en cours de traitement, ce qui nous permet de coder plus efficacement la position de ces valeurs de pitch. A partir des extrémités de la zone voisée on applique la méthode mono-segmentale pour commencer notre procédure d'approximation linéaire du pitch. Partant de cette unique droite joignant les valeurs du pitch aux deux extrémités de la zone voisée, nous cherchons le début de segment dont le pitch est le plus éloignée de cette droite. Si cette distance est supérieure à une distance seuil, nous décomposons l'unique droite de départ en deux droites en prenant le début du segment trouvé comme nouvelle valeur de pitch à transmettre. La même opération est effectuée sur ces deux zones jusqu'au moment où l'on passe sous la distance seuil (fig.5).

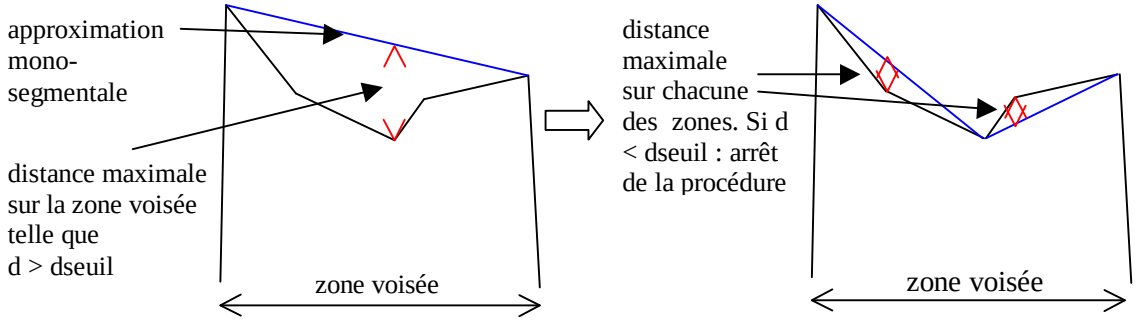

Fig.5 Procédure de codage du pitch par approximation linéaire

Cependant, si une erreur importante est commise lorsque l'on joint le début de deux segments consécutifs pour approximer le contour de pitch original, nous allons avec notre méthode d'approximation linéaire augmenter exagérément le nombre de valeurs de pitch à transmettre sans espoir de faire diminuer l'erreur commise.

Pour ne pas cumuler cette erreur et provoquer une augmentation inutile du nombre de valeurs de pitch à transmettre, nous remplaçons le contour de pitch original de ce segment par la droite joignant les pitchs situés à ses extrémités lorsque la distance entre ces deux contours est supérieure à une valeur seuil maximale. Nos calculs de distance pour la détermination des valeurs de pitch à transmettre sont donc effectués à partir de ce nouveau contour "nettoyé".

Nous constatons en fait que, le suiveur de pitch utilisé dans le système existant étant peu performant, les variations importantes au milieu de ces segments étaient pour la plupart dues à une mauvaise estimation du pitch (doublement de période pitch). Par conséquent, l'approximation faite en joignant les extrémités de certains segments améliore au contraire le résultat final.

Pour coder la position des pitchs nous n'avons qu'à préciser combien de débuts de segments séparent la valeur précédente de pitch à transmettre de la prochaine valeur. Pour limiter le débit nous utilisons là aussi un code de longueur variable. Le nombre de débuts de segments à "laisser passer" avant d'atteindre la prochaine valeur de pitch est généralement codée sur 2 bits.

Pour coder les valeurs des pitchs en représentation logarithmique, nous utilisons un pas de quantification adaptatif en minimisant l'erreur quadratique commise entre le contour du pitch original « corrigé » et le contour retenu. Tout d'abord nous déterminons les valeurs minimum et maximum de la période pitch sur le fichier à coder. Puis on calcule le pas de quantification en

fonction du nombre de bits retenu pour coder les valeurs des pitchs. Nous allons donc transmettre sur 8 bits la valeur minimale de la période pitch sur l'ensemble du fichier, puis coder sur 2 bits le pas de quantification utilisé pour coder le pitch. Ensuite on code sur 6 bits la différence entre la période pitch à coder et la période pitch minimale.

Cette méthode prend en compte les erreurs de détection de pitch, utilise la segmentation, limite le nombre de bits utilisés et tient compte de l'effet de la quantification sur l'erreur de codage. Nous obtenons par cette méthode un débit d'environ 65 bits/s lorsque nous quantifions les pitchs sur 5 bits avec une distance maximale sur la période pitch de 7 échantillons.

Le codage des paramètres prosodiques nécessite l'alignement du pitch et de l'énergie des représentants. Il nous reste donc à étudier le codage permettant l'alignement temporel des représentants.

VI Codage de l'alignement temporel

L'alignement temporel est effectué en suivant le chemin de la DTW qui a servi à déterminer le meilleur représentant. Une première idée que nous pouvons avoir pour le codage des chemins de la DTW serait de se constituer un dictionnaire de chemins pour chaque représentant. On peut imaginer que, pour chaque représentant, il n'y ait qu'un nombre réduit d'alignements possibles différents. A partir de ce dictionnaire de chemins il suffirait d'identifier quel chemin permettrait le meilleur alignement et seul son numéro serait transmis au codeur. Il faudrait étudier la possibilité d'effectuer une quantification vectorielle sur l'ensemble de ces chemins de la DTW. Nous formerions ainsi autant de dictionnaires de chemins qu'il y a de représentants.

Il est très possible que ce soit toujours les mêmes « alignements» qui soient utilisés à la synthèse pour un représentant donné. Pour le savoir, il faudrait connaître le nombre de trames qui composent les représentants alignés et quels sont les vecteurs cepstraux de ces représentants qui sont le plus utilisés. On pourrait à partir de ces résultats supprimer les trames des représentants qui sont peu utilisées ou qui apportent une faible amélioration lors de la synthèse par rapport à d'autres trames voisines du représentant, voire constituer des représentants à partir des vecteurs cepstraux qui sont le plus sollicités lors de l'alignement. Nous aurions par la même occasion la possibilité de se constituer de moins grands représentants ce qui améliorerait le débit du codage des chemins de la DTW.

En fait, nous avons constaté avec notre codeur que des "formes caractéristiques" de chemins apparaissent (fig.6).
Pour la plupart, ces chemins peuvent être classés en deux catégories :

1. La première concerne les segments les plus courts (environ 3 trames). Les chemins suivent alors réellement une trajectoire proche de la diagonale, ce qui est un peu artificiel puisque nous forçons la correspondance entre les premières et les dernières trames du segment à coder et du représentant à aligner
2. La deuxième concerne les segments plus longs. Nous avons constaté que le chemin se limite très souvent à la répétition du même vecteur sur pratiquement toutes les trames du segment (excepté la première et la dernière). Ainsi, un seul vecteur semble convenir pour coder la quasi-totalité du segment.

Cela nous amène à conclure que le codage des chemins de la DTW est un problème qui ne pourra véritablement être abordé que lorsque nous aurons mis au point une méthode efficace de constitution du dictionnaire de représentants.

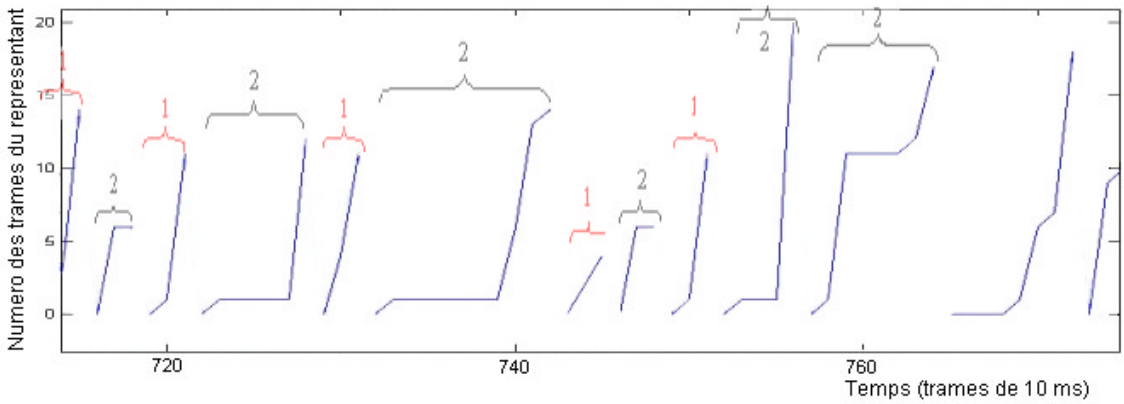

Fig.6 chemins de la DTW permettant d'aligner les représentant d'une portion de fichier de parole

Finalement, nous avons choisi de ne décrire les chemins que par des formes très générales. Nous avons alors constaté qu'il n'était pas vraiment nécessaire de coder les chemins si nous n'améliorions pas d'abord le choix des représentants. Etant donné la forte proportion de segments de petite taille, un alignement des segments suivant la diagonale plutôt que le chemin exact de la DTW n'introduit pratiquement pas de dégradation de qualité.

VII Conclusion et perspectives

Le tableau suivant présente en détail les débits obtenus pour trois fichiers. Le fichier A est tiré de la base d'apprentissage des HMM, la reconnaissance et la segmentation sont donc légèrement plus efficaces que sur le fichier B qui a été enregistré par la même locutrice mais qui n'a pas servi pour l'apprentissage. Cela se traduit par une augmentation des débits pour le codage du pitch et de l'information de voisement, l'efficacité de l'utilisation de l'information prosodique des représentants dépendant directement de la qualité de la segmentation.

Codage de l'enveloppe spectrale et de la prosodie en bits/s	Fichier A [1]	fichier B [2]	fichier C [3]
Enveloppe spectrale	185	179	232
Energie	80	79.5	103
Pitch	54	66	103
Voisement	25	25.5	32
Longueur des segments	47	56	50
Chemins de la DTW	0	0	0
Débit total de la prosodie	**206**	**227**	**288**

[1]fichier de parole appartenant à la base de données d'apprentissage (les fichiers font en moyenne 10 secondes)
[2]fichier de parole de la même locutrice n'appartenant pas à cette base de donnée
[3]fichier de parole d'un locuteur masculin inconnu.

Tab.1. Débits du codeur monolocuteur

Le système étant actuellement monolocuteur, nous pouvons comprendre facilement que cet effet est encore plus marqué lorsque nous codons le fichier C d'un locuteur masculin inconnu. Cet effet est de plus accompagné par une augmentation du nombre de segments par seconde – les HMM n'étant pas adaptés au locuteur – qui entraîne à la fois une augmentation du débit de l'enveloppe spectrale et de la prosodie. Si l'information spectrale est naturellement très mauvaise, la qualité du codage de la prosodie reste tout à fait intéressante avec un débit de 288 bits/s. Une adaptation au locuteur du système actuel fera baisser le nombre de segments, améliorera évidemment le codage de l'enveloppe spectrale et surtout diminuera considérablement le débit de la prosodie.

Le débit obtenu pour ce codeur de prosodie est donc actuellement d'environ 215 bits/s. L'énergie est codée en 80 bits/s, le codage des longueurs des segments prend 56 bits par seconde et descend à 47 bits/s en utilisant une combinaison de codes à longueur variable et une classification préalable des unités acoustiques, le voisement tient en 25 bits/s et le pitch en 54 bits/s. Nous obtenons donc au total un codeur monolocuteur fonctionnant à environ 400 bits/s. A un tel débit, les dégradations dues au codage de l'information prosodique des fichiers A et B du locuteur de la base de donnée, sont quasiment imperceptibles par rapport à la synthèse de ces fichiers de parole avec transmission complète et sans quantification de la prosodie.

Ce travail nous a permis d'identifier un certain nombre de limitations du système existant. Tout d'abord, il serait nécessaire d'améliorer le procédé d'analyse / synthèse paramétrique, qui pour le moment introduit de nombreux artefacts même en l'absence de tout codage des paramètres. Un suiveur de pitch plus robuste permettrait d'éviter des erreurs d'octave et des fluctuations injustifiées qui dégradent la qualité et augmentent inutilement le nombre de valeurs de pitch à transmettre. Un système d'analyse / synthèse à excitation mixte du type harmonique plus bruit permettrait également d'améliorer le naturel de la parole synthétisée.
Ensuite, dans le système actuel, les unités représentantes sont tout simplement les 8 unités les plus longues appartenant à la même classe acoustique. En améliorant le choix de ces unités représentantes (et éventuellement en autorisant un nombre variable par classe acoustique), il serait vraisemblablement possible d'améliorer à la fois le codage de l'enveloppe spectrale et le débit associé au codage de la prosodie. Des contraintes prosodiques pourraient également conditionner le choix des classes acoustiques et la segmentation lors de la reconnaissance par l'algorithme de Viterbi, afin par exemple d'éviter des transitions de voisement à l'intérieur des segments. Enfin, au niveau du codage, il semble possible d'améliorer la procédure de recherche de la meilleure unité représentante par DTW en introduisant des critères relatifs à l'évolution de l'énergie ou de la prosodie. Ainsi nous devrions à la fois améliorer la qualité et diminuer le débit associé à la transmission de la prosodie.
Enfin, le système est actuellement essentiellement monolocuteur : des techniques d'adaptation au locuteur devront être étudiées. Le problème de la résistance aux erreurs de transmission devra également être analysé.

Références

[1] J. Cernocký, G. Baudoin and G. Chollet. Segmental vocoder - going beyond the phonetic approach. In Proc. IEEE ICASSP 98, pages 605-608, Seattle, WA, May 1998.

[2] G. Baudoin, J. Cernocký, P. Gournay, G. Chollet, *Codage de la parole à bas et très bas débit*, Annales des Télécommunications, à paraître.

[3] Thèse de J. Cernocký, *Speech Processing Using Automatically Derived Segmental Units : Applications to very Low Rate Coding and Speaker Verification*, Université Paris XI Orsay, dec. 1998.

[4] Mouy, B., de La Noue, P., and Goudezeune, G. " NATO STANAG 4479: A standard for an 800 bps vocoder and channel coding in HF-ECCM system ", IEEE Int. Conf. on ASSP, Detroit, pp. 480-483, May 1995.