Vers une analyse acoustico-phonétique de la parole indépendante de la langue, basée sur ALISP *

Jan Černocký (1), Geneviève Baudoin (2), Gérard Chollet (3) et Dijana Petrovska-Delacrétaz (3)

[1] Institut de Radioélectronique, FEI VUT Brno, République Tchèque
[2] Département Signaux et Télécommunications, ESIEE Paris, France
[3] Département Signal et Images, ENST Paris, France

7 février 2001

Résumé : De nombreux systèmes de synthèse et de reconnaissance automatique de la parole utilisent des unités segmentales liées aux phones. Les phones sont les réalisations physiques des phonèmes correspondants. Ils sont donc, en général, définis a-priori et dépendants de la langue considérée. Nous présentons une alternative à cette approche : une détermination des unités de parole à l'aide des techniques ALISP (Traitement Automatique de la Parole, Indépendant de la Langue). ALISP permet de choisir l'inventaire des unités segmentales considérées à partir d'une analyse statistique de corpus de parole sans a-priori sur nos connaissances phonétiques et/ou phonologiques. Nous avons testé expérimentalement de telles unités dans un vocodeur à très bas débit : le débit moyen ainsi obtenu pour le codage des unités est de 120 bps. Nous présentons également les résultats de la comparaison d'une segmentation ALISP avec une segmentation acoustico-phonétique dans deux cas : mono et multilocuteur.

Abstract : Numerous systems for speech synthesis and automatic speech recognition make use of segmental units linked to phones, those phones being the physical realizations of corresponding phonemes. Such units are therefore defined a-priori and depend on the given language. Here, we present an alternative approach : determination of speech units using the ALISP (Automatic, Language Independent Speech Processing) techniques. ALISP allows to choose the inventory of units from a statistical analysis of the speech corpus, without any a-priori knowledge of phonetics and/or phonology. Experimentally, such units have been tested in a very low bit-rate speech coder : the resulting average rate is 120 bps. The results of a comparison of an ALISP segmentation with acoustic-phonetic segmentation are presented too in the case of one and multiple speakers.

1 Introduction

Les systèmes modernes de traitement automatique de la parole s'appuient sur sur des unités de type *sous-mot*. Dans les systèmes de reconnaissance vocale (à vocabulaire illimité), les unités constituent le niveau intermédiaire entre la description acoustique (ou paramétrique) et le niveau lexical. Dans la reconnaissance du

* Ce travail a été partiellement financé par le Ministère de l'Éducation de la République tchèque, sous le projet N^o VS97060.

locuteur, une pré-segmentation avec ces unités suivie de l'utilisation de plusieurs systèmes de décision permet d'obtenir de meilleurs résultats que les systèmes utilisant une modélisation "globale". Finalement dans le codage à très bas débit (very low bit-rate), la transmission de l'information sur chaque trame acoustique n'étant plus possible, ces unités déterminent l'information symbolique transmise dans le canal ou stockée.

Dans tous les domaines cités, ces unités de base doivent être modélisées par des modèles mathématiques. Le schéma général d'estimation de ces modèles (applicable aussi bien pour la reconnaissance, la synthèse que pour le codage) est donné sur la Figure 1. En synthèse, par exemple, l'entrée du modèle est un texte. Celui-ci est converti en signal de parole qui est ensuite comparé avec ce même texte lu par un locuteur humain. Le modèle (classiquement des diphones) est ajusté de façon à minimiser la différence entre la parole synthétique et humaine. Le critère est ainsi clairement défini :

1. dans la **synthèse vocale**, la parole synthétique doit différer le moins possible du signal produit par un locuteur humain.
2. dans la **reconnaissance de parole**, la compréhension du signal par la machine devrait s'approcher de la compréhension humaine (nous n'allons pas parler ici du texte, la dictée n'étant pas la seule application de la reconnaissance de la parole).
3. dans le **codage**, le signal après la chaîne codage-décodage doit différer le moins possible du signal original.
4. dans la **vérification du locuteur**, le but est de mieux séparer les clients d'un système des non-clients (imposteurs).

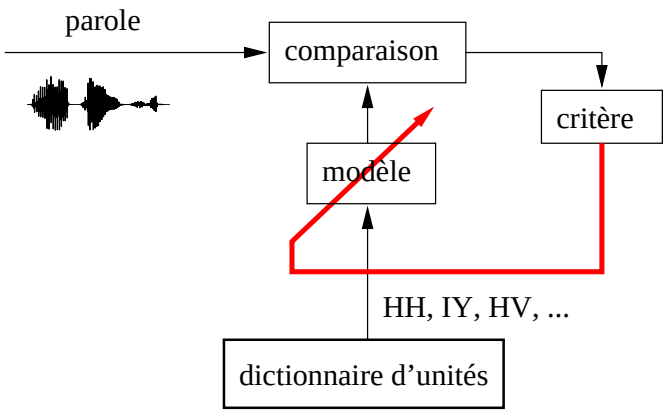

FIG. 1 – Estimation des modèles pour le traitement de la parole.

Classiquement, ces unités dérivent d'unités linguistiques telles que : les **phonèmes**, leurs dérivés (diphones, phonèmes en contexte, ...), syllabes, ou autres. Cependant, l'usage de ce type d'unités ne s'impose que dans le premier domaine et encore seulement à l'entrée d'un système de synthèse. Dans les autres domaines (2,3,4), ce choix se justifie historiquement, car dans les langues indo-européenes, les recherches en phonétique et acoustique de la parole sont classiquement très liées à l'orthographe (on pourrait se demander, quel serait l'état de ces sciences, si par exemple l'anglais était écrit en idéogrammes).

De plus, pour une application en traitement automatique de la parole, la définition d'un jeu d'unités et la détermination de leurs positions dans le signal de parole (un alignement) nécessite une très bonne expertise en phonétique et en linguistique. La transcriptions et/ou l'annotation manuelle de bases de données (BD ou corpus) sont des tâches très lourdes ; il est connu, que ces étapes sont les plus coûteuses et les plus sujettes aux erreurs humaines dans le procédé de création des corpus.

C'est pourquoi nous avons utilisé une approche alternative s'appuyant sur la détermination *automatique* des jeux d'unités et la transcriptions *automatique* des corpus. Les techniques regroupées sous un nom générique ALISP (Automatic Language Independent Speech Processing – Traitement Automatique de la Parole, Indépendant de la Langue) se basent sur les *données* et tentent de limiter au minimum les connaissances a-priori nécessaires. Recherchant un équilibre entre la précision de la description et son économie, ces techniques détectent des régularités dans le signal (ou sa paramétrisation) pour en faire émerger sa structure.

L'idée qu'il est possible d'apprendre des unités de base pour le traitement automatique de la parole en n'exploitant uniquement le signal, est fondée sur des expériences "naïves" (les bébés apprennent à parler sans savoir lire, en écoutant la voix de leurs proches, la même remarque s'applique aux adultes analphabètes), mais aussi sur des travaux de nombreux chercheurs dans des domaines parfois différents du traitement automatique de la parole : Kohonnen [12, 13] a développé une théorie des mémoires associatives, capables d'apprentissage non supervisé et de généralisation sur les données, Kruskal et Sankoff [14] ont publié de nombreux travaux sur l'apprentissage des séquences d'ADN, et Atal [2] a défini un hyper-espace des segments acoustiques, où il tente de mesurer l'information portée par le langage parlé. Enfin, les travaux de de Marcken [5] portent sur l'acquisition non-supervisée du lexique de la parole continue. Ses algorithmes sont basés sur l'encodage optimal des séquences de symboles au sens de longueur de description minimale (Minumum Description Length) et utilisent une représentation hiérarchique du langage.

L'utilisation de ces unités est directe dans les domaines, où la représentation symbolique ne constitue qu'un niveau intermédiaire entre deux signaux (le codage), ou auxiliaire (pré-segmentation dans la vérification). Au cas, où l'on exige une transcription linguistique (reconnaissance vocale), des techniques de mise en correspondance (mapping) des unités ALISP et des phonèmes doivent être élaborées. Mieux encore, il est possible de remplacer des dictionnaires de prononciation classiques par leurs homologues constitués à partir d'unités déterminées automatiquement.

Cet article est structuré de la façon suivante : dans la section 2, nous présentons les outils ALISP qui servent à déterminer des unités dans un corpus des signaux de parole. La section 3 est consacrée aux expériences en codage de la parole à très bas débit (la vérification la plus simple de notre approche). La section 4 présente les résultats d'une comparaison d'une segmentation ALISP avec une segmentation acoustico-phonétique fine et une segmentation en classes phonétique larges (les deux étant obtenues par un système de reconnaissance). La section 5 conclut notre article.

2 Outils ALISP – justifications théoriques et solutions techniques

Sur un corpus de parole donné, la détermination des unités s'effectue en deux étapes principales : dans la première, nous définissons le jeu d'unités et nous recherchons une segmentation initiale du corpus. Dans la deuxième, ces unités sont modélisées par des modèles stochastiques. Le système est ainsi *appris* et peut traiter un signal de parole inconnu.

Nous appelons les techniques utilisées pour cette extraction et cette modélisation des "outils" (voir la chaîne de traitement Figure 2). Certains parmi eux sont utilisés largement en traitement de la parole (paramétrisation, modèles de Markov cachés), les autres (décomposition temporelle, multigrammes) sont plus spécifiques aux approches ALISP. Ces outils sont hautement modulaires, et la position de certains d'entre eux dans la chaîne de traitement peut changer (c'est le cas pour les multigrammes). Les sous-sections suivantes donnent une description plus détaillée de ces outils.

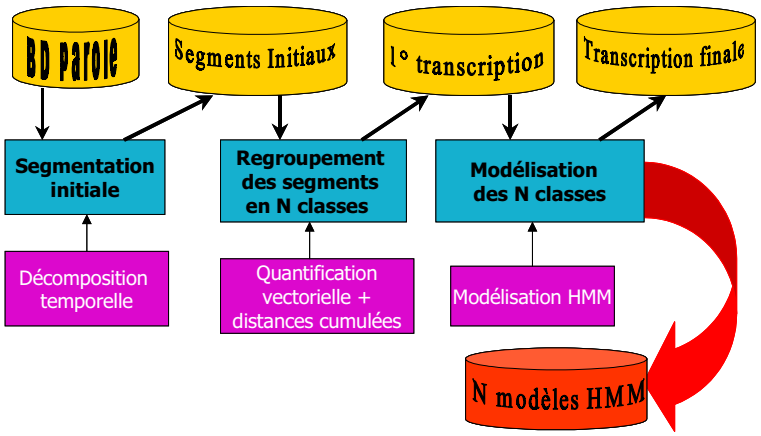

FIG. 2 – Outils utilisés dans la recherche des unités pour le traitement de la parole.

2.1 Paramétrisation

L'une des premières étapes dans tout traitement automatique de la parole est sa **paramétrisation**. À partir d'un signal numérisé, nous devons extraire un nombre limité de paramètres, décrivant le signal, et convenable pour le traitement automatique de la parole 3. Ici, nous nous limitons aux paramètres d'un filtre numérique représentant le conduit vocal selon la Figure 4. Ces paramètres, dits de *prédiction linéaire LPC* [15], sont convertis en paramètres LPC-cepstraux [15], moins corrélés que les coefficients de la prédiction linéaire. Ces paramètres modélisent l'enveloppe spectrale du signal de parole. (Fig. 5), d'où une possible difficulté en appliquant nos méthodes sur les langues tonales. Il est cependant connu, que la fréquence fondamentale influence aussi cette enveloppe spectrale.

Les paramètres sont classiquement extraits sur des trames de longueur fixe. Cette limitation par rapport à des approches comme les ondelettes par exemple, est compensée par une notion de variabilité temporelle dans les autres outils décrits plus loin.

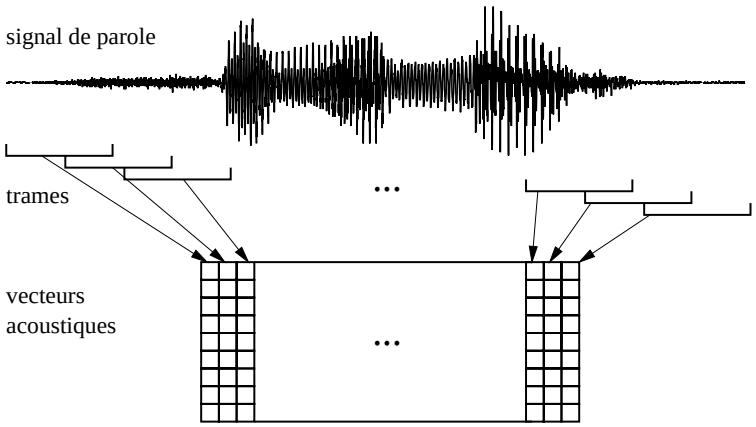

FIG. 3 – Signal de parole, découpage en "trames", détermination d'un nombre limité de paramètres caractérisant chaque trame.

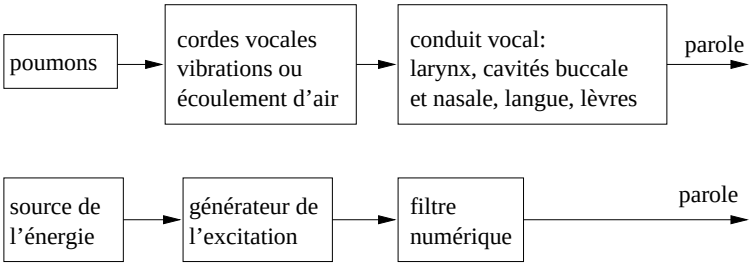

FIG. 4 – Modèle de production de la parole utilisé pour l'extraction des paramètres.

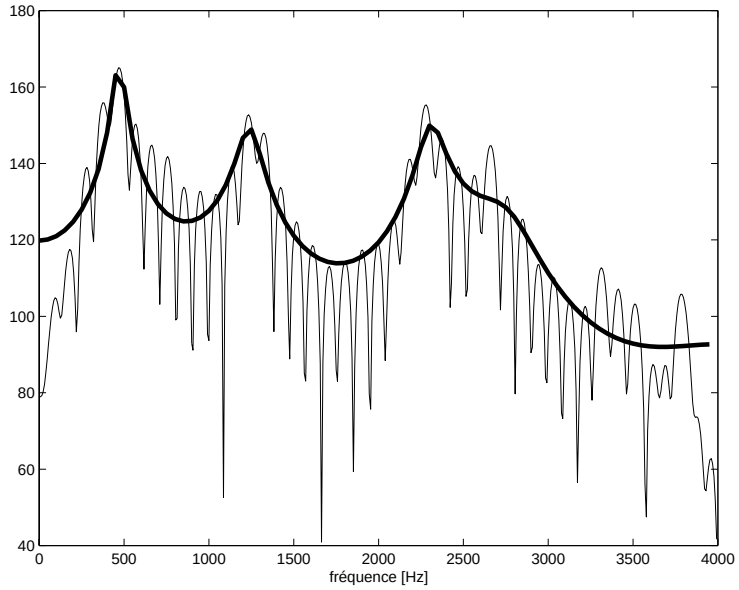

FIG. 5 – Spectre d'un segment voisé de parole avec son enveloppe.

2.2 Décomposition temporelle

On applique la *décomposition temporelle* sur les vecteurs de coefficients LPC-cepstraux. Nous avons choisi cette méthode parmi les nombreuses techniques de segmentation de la parole (voir [19] pour un résumé), car non seulement elle utilise un critère de stabilité spectrale pour déterminer un segment, mais elle prend aussi en compte leurs transitions – celles-ci sont représentées par un recouvrement des fonctions d'interpolation. La décomposition temporelle, introduite par Atal [3] et perfectionnée par Bimbot [4], approche une matrice de paramètres par des vecteurs-cibles et des fonctions d'interpolation. Techniquement, la recherche des cibles et des fonctions d'interpolation de la décomposition temporelle se fait par une *décomposition en valeurs singulières* à court terme d'une sous-matrice $\mathbf{Y}$ de la matrice des coefficients cepstraux $\mathbf{X}$:

$$\mathbf{Y}^T = \mathbf{U}^T \mathbf{D} \mathbf{V}.$$

On assemble ensuite les lignes de la matrice $\mathbf{U}$ pour trouver une fonction d'interpolation concentrée sur une fenêtre rectangulaire. La ré-estimation de la fonction d'interpolation et l'adaptation de la fenêtre sont itérées pour obtenir une compacité maximale de la fonction d'interpolation. Le post-traitement des fonctions d'interpolation contient un lissage, une dé-corrélation, et une normalisation. Dans l'étape suivante, le calcul des

cibles est effectué en utilisant la pseudo-inverse de la matrice $\mathbf{\Phi}$. Enfin, les cibles et fonctions d'interpolation sont affinées localement.

Les fonctions d'interpolation, déterminant ainsi des parties quasi-stationnaires du signal, définissent une première *segmentation* de la parole. La Figure 6 montre un exemple de la décomposition temporelle.

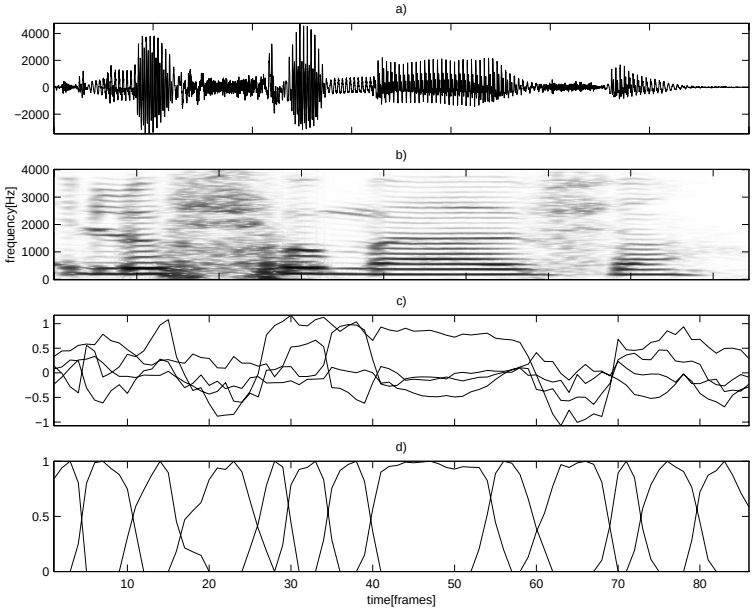

FIG. 6 – Exemple d'une décomposition temporelle - "le chômage". a) le signal, b) le spectrogramme, c) les trajectoires des 4 premiers coefficients LPC-cepstraux, d) les fonctions d'interpolation de la décomposition temporelle.

2.3 Quantification vectorielle

Les segments trouvés subissent une classification non-supervisée, réalisée ici par une *quantification vectorielle*. Il existe plusieurs méthodes [10], de recherche de classes en fonction de la proximité des vecteurs de paramètres dans un espace à P dimensions : à côté de la quantification vectorielle, ce sont par exemple les *Modèles de Markov cachés ergodiques*, où l'on attribue les vecteurs aux états en fonction d'une vraisemblance, ou les *Self-Organizing Maps* de Kohonnen, où la proximité dans un espace à grande dimension se traduit par une proximité des classes dans une espace à petite dimension (typiquement 2 – une surface). Le but de toutes ces méthodes est de réunir dans une même classe les vecteurs qui se ressemblent, et de mettre dans les classes différentes les vecteurs distincts. Mathématiquement parlant, il nous faut minimiser les distances intra-classe, tout en maximisant les distances inter-classe.

La quantification vectorielle est une réponse simple à ce problème. Les vecteurs sont représentés par un dictionnaire de vecteurs-codes (nous allons utiliser le terme anglais *codebook* dans la suite, pour ne pas confondre ce dictionnaire avec le dictionnaire des unités ALISP) : $\mathbf{Y} = \{\mathbf{y}_i; 1 \leq i \leq L\}$, où L est le nombre de classes. Ce codebook doit être *appris* sur une base de données en minimisant une distance moyenne globale entre les vecteurs d'apprentissage et les vecteurs-codes. Cet apprentissage est mis en œuvre par l'algorithme Linde–Buzo–Gray [10] avec des éclatements successifs du codebook : $L = 1, 2, 4, \ldots$ L'ensemble d'apprentissage est constitué des vecteurs cepstraux originaux situés aux centres de gravité des fonctions d'interpolation.

Une fois le codebook appris, nous pouvons procéder à une *quantification* : dans cette étape, on attribue à chaque événement de la décomposition temporelle le numéro (étiquette) de la classe qui lui est la plus proche. Pour cette quantification, nous avons utilisé tous les vecteurs d'un segment prédéterminé par la décomposition temporelle en utilisant une distance cumulée .

La décomposition temporelle avec la quantification vectorielle effectuent ainsi une *transcription initiale* (bornes temporelles et labels) de la base de données de parole.

2.4 Multigrammes

Il se peut que nous ayons besoin d'unités plus longues que celles déterminées par une combinaison décomposition temporelle+quantification vectorielle. Bien que nous travaillions avec des unités déterminées automatiquement, nous pouvons nous approcher ainsi des techniques syllabiques ou diphoniques utilisées dans les traitements classiques. Ce séquencement a de nombreux avantages : en codage par exemple, nous pouvons ainsi limiter le débit binaire (le dictionnaire d'unités devient plus grand, mais le nombre d'unités à transmettre par seconde decroît) et nous pouvons de plus, en limitant ainsi le nombre de transitions entre unités, atténuer les effets indésirables dûs à la concaténation de segments courts. On appelle *"multigramme"* une séquence formée d'un nombre variable de symboles, et n-multigrammes les multigrammes, dont la longueur est limitée à n. La technique utilisée pour ce séquencement est appelée décomposition en multigrammes [6]. Cette méthode, dont nous connaissons plusieurs variantes – discrètes ou continues – permet de détecter des *séquences caractéristiques* d'unités dans le corpus d'apprentissage.

Nous supposons que les événements de la décomposition temporelle ont déjà été étiquetés par la quantification vectorielle (nous avons donc une chaîne de symboles – Figure 7). Pour un dictionnaire de multigrammes $\{x_i\}$ donné, la segmentation d'une chaîne d'observations discrètes et sa transcription en multigrammes se fait en maximisant la vraisemblance de la segmentation et de l'étiquetage :

$$(S^\star, X^\star) = \arg \max_{\forall (S,X)} L(O, S, X | \{x_i\}), \tag{1}$$

où O est la chaîne d'observations, S est sa segmentation et X l'attribution des multigrammes. Pour les multigrammes discrets, le dictionnaire contient les différentes séquences appelées multigrammes x_i ainsi que leurs probabilités π_i. Nous pouvons écrire : $s_j \equiv x_i$; $L(O, X|\{x_i\}) = P(x_{i_1})P(x_{i_2}) \dots P(x_{i_q})$.

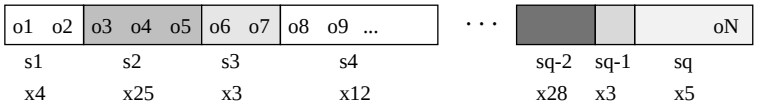

FIG. 7 – Séquencement des symboles par les multigrammes.

Le dictionnaire de multigrammes n'est pas connu a-priori et doit être appris sur une base de données de symboles. Cet apprentissage commence par une *initialisation*. on initialise les valeurs des probabilités π_i de toutes les séquences possibles de longueur 1 à n par le nombre d'occurrences de ces séquences dans la base de données d'apprentissage. Après cette initialisation, on itère plusieurs étapes de segmentation au sens du maximum de vraisemblance (Éq. 1). À l'étape n, on effectue la segmentation en utilisant le dictionnaire déterminé à l'étape $n-1$, puis on met à jour les probabilités π_i des multigrammes à partir de la nouvelle segmentation. Durant ces itérations, le dictionnaires est élagué des multigrammes rares en imposant un nombre d'occurrences minimal.

On peut utiliser la méthode des multigrammes à 2 niveaux de la chaîne de traitement. On peut l'utiliser sur :

– **Des événements de la décomposition temporelle quantifiés par quantification vectorielle**. Les multigrammes servent ici à initialiser des HMMs (voir la sous-section suivante) avec des nombres d'états variables.
– **Les symboles générés par une segmentation par les HMMs**. Les multigrammes aident ici à la création d'unités plus longues.

Les multigrammes constituent ainsi un module dont la position peut varier dans le schéma de la Figure 2.

2.5 Modèles de Markov cachés

Dans la deuxième étape de traitement, les unités trouvées par la combinaison décomposition temporelle+quantification vectorielle ou décomposition temporelle+quantification vectorielle+multigramme sont *modélisées* par les *Modèles de Markov Cachés (HMM)*. Cependant, ce formalisme, utilisé largement en reconnaissance de parole, ne sert pas seulement à produire des modèles, mais contribue lui-même à un affinement du jeu d'unités par des itérations de segmentation du corpus (un alignement des HMM avec les données) et de ré-estimation des paramètres des modèles.

La théorie des HMM [16, 21] est assez complexe et ne peut pas être traitée ici en détail. La reconnaissance de parole à l'aide des HMM est basée sur la maximisation de la vraisemblance de l'observation et des modèles :

$$\arg \max_{\{M_1^N\}} L(\mathbf{O}|M_1^N)L(M_1^N),$$

où $\mathbf{O}$ est une chaîne d'observations (vectorielles cette fois-ci), et M_1^N une séquence de modèles. La vraisemblance $L(\mathbf{O}|M_1^N)$ dite "acoustique" quantifie la correspondance entre les données et les modèles, quant à la vraisemblance $L(M_1^N)$ (modèle de langage), elle donne une plausibilité a-priori de la séquence de modèles M_1^N.

Un choix important est celui de *l'architecture* des HMM. Nous avons choisi l'architecture la plus simple gauche-droite (Figure 8). Le *nombre de modèles* est déterminé par la taille L du codebook de quantification vectorielle ou par la taille Z du dictionnaire des multigramme. Le nombre d'états-émetteurs des HMM est défini comme $2i + 1$, où i est le nombre des unités dans un multigramme. Au cas, où l'on ne travaille pas avec les multigrammes, ce nombre est $2 \times 1 + 1 = 3$. Dans la plupart de nos travaux, la notion du modèle de langage n'a pas été utilisée et nous avons attribué la même probabilité a-priori à tous les modèles.

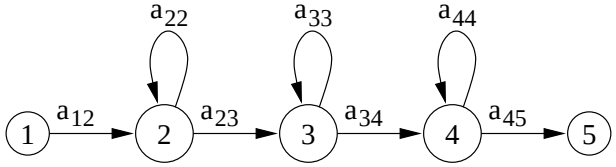

FIG. 8 – Modèle de Markov caché avec une architecture gauche-droite et 3 états émetteurs.

L'apprentissage des HMM se fait sur le même corpus que celui utilisé pour apprendre la décomposition temporelle et quantification vectorielle. *L'initialisation* des HMM prend en compte les transcriptions initiales T^0 obtenues par la combinaison décomposition temporelle+quantification vectorielle ou décomposition temporelle+quantification vectorielle+multigramme. Les modèles sont appris sans contexte et en contexte (apprentissage itéré) [21] pour aboutir à un jeu de paramètres initiaux Λ^0 :

$$\Lambda^0 = \{\lambda_i^0\} = \arg \max_{\forall \Lambda} L(O, \Lambda|T^0). \quad (2)$$

On répète ensuite, les étapes de segmentation à l'aide des modèles préalablement appris et de ré-estimation des paramètres de ces modèles :

- **Segmentation :** $T^{m+1} = \arg\max_{\{M_1^N\}} L(\mathbf{O}, M_1^N | \Lambda^m, LM^m)$.
- **Ré-estimation** des paramètres HMM : $\Lambda^{m+1} = \arg\max_{\{\Lambda\}} L(\mathbf{O}, \Lambda | T^{m+1})$.
- **Terminaison :** on arrête si l'augmentation de la vraisemblance n'est plus significative, ou si le nombre d'itérations est plus grand qu'un seuil. Sinon, retour à la segmentation.

Nous avons trouvé que l'utilisation de cette technique d'affinement améliore la cohérence des modèles avec les données (au sens d'une augmentation de la vraisemblance) et aussi la cohérence des segments acoustiques dans des différentes classes (la ressemblance des segments dans une classe devient meilleure).

Les techniques utilisées fournissent donc 3 types de résultats : un *dictionnaire d'unités*, déterminé sur le corpus d'apprentissage, une *transcription* du corpus d'apprentissage utilisant ces unités et un *jeu de modèles HMM*.

3 Expériences – codage de parole à très bas débit

Le codage à très bas débit à l'aide des unités ALISP nous a servi de première vérification de nos approches. En codage, où le passage au niveau lexical n'est pas indispensable, le critère pour évaluer la qualité des unités créées est simple : la parole à la sortie du décodeur doit différer le moins possible de la parole originale, et le débit binaire nécessaire pour transmettre l'information sur les unités (et une information auxiliaire, comme nous allons le voir plus loin) doit rester bas.

L'approche du codage de la parole à l'aide des phonèmes – *"codage phonétique"* – n'est pas nouvelle (les travaux de Ismail et Ponting [11], Ribeiro et Trancoso [17, 18], et autres), mais dans notre approche nous avons créé les unités sans aucune supervision.

Une fois le dictionnaire des unités créé, nous pouvons les détecter dans la parole à l'entrée du codeur, *transcrire* la parole à l'aide de ces unités, et envoyer leurs indices dans le dictionnaire au décodeur. Nous devons également transmettre une information supplémentaire sur la prosodie (fréquence fondamentale, énergie, longueur des segments). Au décodeur, la parole de sortie s'obtient par une *synthèse vocale*. D'abord, nous dérivons l'information sur des *unités de synthèse* à partir des unités de codage. Ensuite, nous recherchons des *représentants* dans un dictionnaire, créé dans la phase d'apprentissage. L'ensemble de paramètres du représentant choisi contrôle le synthétiseur en association avec l'information prosodique (voir Figure 9).

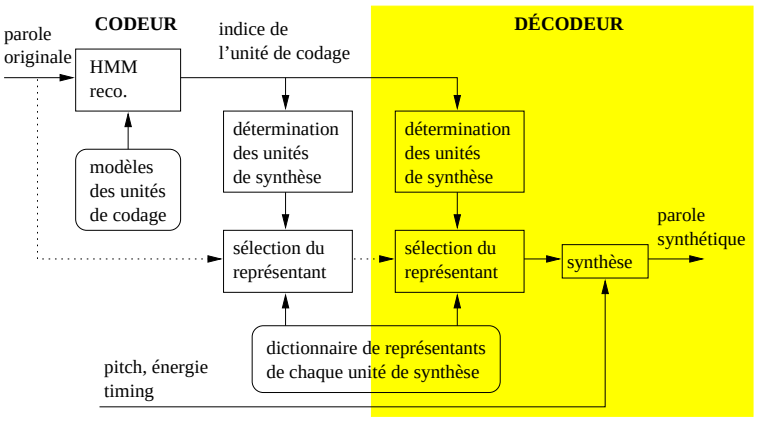

FIG. 9 – Codage et décodage de la parole : unités de codage, unités de synthèse et les représentants.

Dans l'évaluation du *débit binaire* nécessaire pour la transmission de l'information sur les unités, nous n'avons pas considéré les probabilités a-priori des unités (codage entropique [9]), mais nous avons calculé le nombre de bits nécessaire pour la transmission de chaque unité M_i par $\log_2 Z$, où Z est la taille du dictionnaire. Le débit binaire moyen est ainsi défini :

$$R_u = \frac{\log_2 Z \sum_{i=1}^{Z} c(M_i)}{T_f \sum_{i=1}^{Z} c(M_i) l(M_i)}, \tag{3}$$

où $c(M_i)$ est le nombre d'occurrences de M_i dans la chaîne encodée, et T_f est le décalage entre des trames acoustiques en secondes. La *qualité* de la parole après codage-décodage a été évaluée subjectivement par des tests informels.

Nous avons effectué les expériences en mode mono-locuteur, avec les données de deux corpus : Boston University Radio Speech Corpus (anglais américain) et Martin Ruzek (tchèque).

3.1 Boston University Radio Speech Corpus

Les données de ce corpus américain distribué par Linguistic Data Consortium[1] sont de qualité "Hi-Fi" (fréquence d'échantillonnage 16 kHz). Le corpus contient la parole de 7 présentateurs professionnels. Nous avons utilisé les données d'un locuteur masculin – 78 minutes et un locuteur féminin – 83 minutes. Selon la provenance des enregistrements, les données ont été divisées en un corpus d'apprentissage (celles enregistrées de la radio) et de test (données enregistrées au studio de Boston University).

Nous avons effectué une paramétrisation avec 16 coefficients LPC-cepstraux en trames de 20 ms (recouvrement 10 ms). La soustraction de la moyenne cepstrale (CMS) a été faite pour chaque appel. Nous avons ensuite appliqué la décomposition temporelle, ajustée afin de produire 15 cibles par seconde en moyenne. Sur les segments obtenus, nous avons appris un dictionnaire (codebook) de quantification vectorielle à 64 vecteurs-codes. Les HMM étaient appris directement sur les transcriptions décomposition temporelle+quantification vectorielle (sans pré-traitement par les multigrammes). Leur nombre réduit (64) a permis un affinement avec 5 itérations de segmentation et de ré-estimation. Nous avons vérifié que la vraisemblance d'alignement des données avec les modèles augmentait. Nous avons ensuite testé une application des multigrammes sur la dernière segmentation HMM, et nous avons obtenu des dictionnaires de séquences de longueur variable 1 à 6, de tailles 722 (pour le locuteur féminin) et 972 (pour le sujet masculin).

Pour le décodage, nous avons utilisé des unités de synthèse équivalentes à celles de codage, et nous avons disposé de 8 représentants pour chacune. Ici, nous avons testé une synthèse par prédiction linéaire et nous n'avons pas considéré le codage de la prosodie, les contours de F_0 et de l'énergie originaux étant introduits directement dans le synthétiseur. Les débits binaires (seulement pour le codage des unités et incluant les 3 bits nécessaires pour le codage du choix de représentant) obtenus sont donnés dans la Table suivante :

locuteur	féminin		masculin	
débit binaire sur l'ensemble de	apprentissage	test	apprentissage	test
HMM 6-ème génération	189.27	190.28	189.75	195.51
HMM 6-ème génération+multigrammes	135.91	145.09	141.86	156.02

En évaluant la qualité de la parole obtenue, nous l'avons jugée intelligible, avec une meilleure qualité pour les multigrammes (moins de distorsions sur les transitions).

[1] University of Pennsylvania, http ://www.ldc.upenn.edu/

3.2 Corpus de Martin Ruzek

Cette base de données tchèque a été créée en coopération entre l'Université Technique de Brno, l'Université Masaryk de Brno[2] et la Radio Tchèque, station Brno[3]. Nous avons numérisé à 11025 Hz deux bandes avec des textes lus par le célèbre acteur Martin Ruzek. Les longs paragraphes ont été éclatés en recherchant les minima de l'énergie, les fichiers ainsi obtenus ayant une longueur de 6 à 18 secondes. Nous avons rejeté les fichiers avec des bruits de fond (musique, et autres). Ce corpus a été divisé en une partie réservée à l'apprentissage (7/8) et une partie réservée aux tests (1/8).

Nous avons paramétrisé les données dans les trames de 220 échantillons avec un recouvrement de 110 échantillons (20 et 10 ms environ). Nous avons utilisé 12 coefficients cepstraux. Nous avons également calculé la fréquence fondamentale, par une méthode FFT-cepstrale sur des trames plus longues (500 échantillons). Le reste des traitement a été similaire aux expériences précédentes, avec environ 15 cibles de décomposition temporelle par seconde, un codebook de 64 vecteurs-codes, la même architecture de HMM et le même procédé de ré-estimation. Celle-ci a duré environ 8 heures sur un Pentium 233 MMX sous Linux.

Le codage a été réalisé de façon similaire aux expériences sur Boston University corpus , mais nous n'avons pas utilisé le prolongement des unités par les multigrammes. Les débits ainsi obtenus étaient de 173.26 bps sur le corpus d'apprentissage et de 175.44 bps sur le corpus de test.

La parole après codage-décodage est intelligible, mais n'est pas naturelle et souffre d'artéfacts audibles. Il faut cependant prendre en compte, qu'il n'y a pas de lissage sur les frontières des unités, et que la synthèse utilisée (une simple synthèse LPC) est assez primitive. L'usage de technique plus avancées : HNM (Harmonic Noise Model) ou PSOLA (Pitch-Synchronous Overlap and Add) devrait améliorer considérablement la qualité de la parole, tout en conservant le bas débit de transmission.

3.3 Bref corpus

Por evaluer nos algorithmes dans un cas independant du locutuer, nous avons utilise une partie des locuteur masculins de la base de donnee Bref, un corpus grand vocabulaire de textes francais lus, dans un environement pas bruite. Les donnes textuelles sont choisies dans le journal "Le monde", et reparties parmi 120 locuteurs. Nous avons choisi 40 locuteurs masculins pour nos experiences multilocuteur.

Ces donnees (echantillonnes a 16 kHz) ont subi les memes procedures que les donnees de Boston University. Le protocol experimental est identique a celui du Boston University corpus, mais il est applique aux 40 heures de parole provenant de 40 locuteurs differents. Pour les experiences de codage a tres bas debit, sans utilise la techniques des multigrammes, nous avons obtenus des debits de 133 bps pour les cas multilocuteur. La qualite de la parole apres codage-decodage est intelligible (de meme ordre d'intelligibilite que celle obtenue pour le cas monolocutuer). Pour ce travail, nous nous sommes surtout interesses a la correspondance des unites obtenues automatiquement (que nous nommons unites ALISP) avec des unites phonetiques.

[2]Nous remercions Ludek Bartek pour son support technique et sa patience pendant les enregistrements.

[3]Nous tenons à remercier M. Jaroslav Vojacek, son directeur commercial, de nous avoir permis d'utiliser ces enregistrements pour nos recherches.

4 Étude des correspondances entre une segmentation ALISP et une segmentation acoustico-phonétique

4.1 Cas monolocuteur

Pour le corpus de Boston University, nous avons pu comparer la segmentation obtenue sur les données du locuteur féminin avec une segmentation acoustico-phonétique. Cette dernière a été obtenue par un système de reconnaissance de phonèmes et d'unités sub-phoniques (nous allons appeler les deux classes "unités phonétiques") à Boston University, et est jointe au corpus. Pour quantifier cette correspondance, nous avons mesuré les recouvrements des unités ALISP avec les unités phonétiques, et avons évalué une matrice de confusion $\mathbf{X}$ de taille $n_p \times n_a$ à l'aide des recouvrements relatifs :

$$x_{i,j} = \frac{\sum_{k=1}^{c(p_i)} r(p_{i_k}, a_j)}{c(p_i)}.$$

où $c(p_i)$ est le nombre d'occurrences du phonème p_i dans le corpus, et $r(p_{i_k}, a_j)$ est un recouvrement relatif de k-ème occurrence de l'unité phonétique p_i avec l'unité ALISP a_j. Le recouvrement relatif se calcule à partir du recouvrement absolu (cf. Figure 10) par une simple normalisation par la longueur de l'unité phonétique : $r(p_{i_k}, a_j) = R(p_{i_k}, a_j)/L(p_{i_k})$. La valeur $x_{i,j}$ quantifie alors la correspondance de i-ème unité phonétique avec j-ème unité ALISP sur tout l'ensemble de données étudiées. Un example de segmentation en unités phonétiques et unités ALISP du mot "wanted" est montré sur la Figure 11.

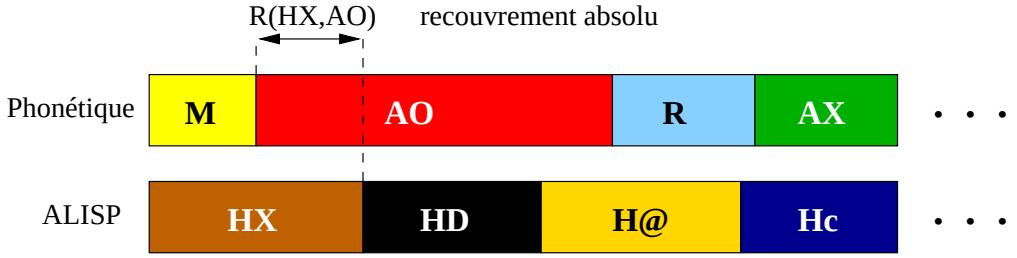

FIG. 10 – Recouvrement des unités ALISP avec des unités phonétiques.

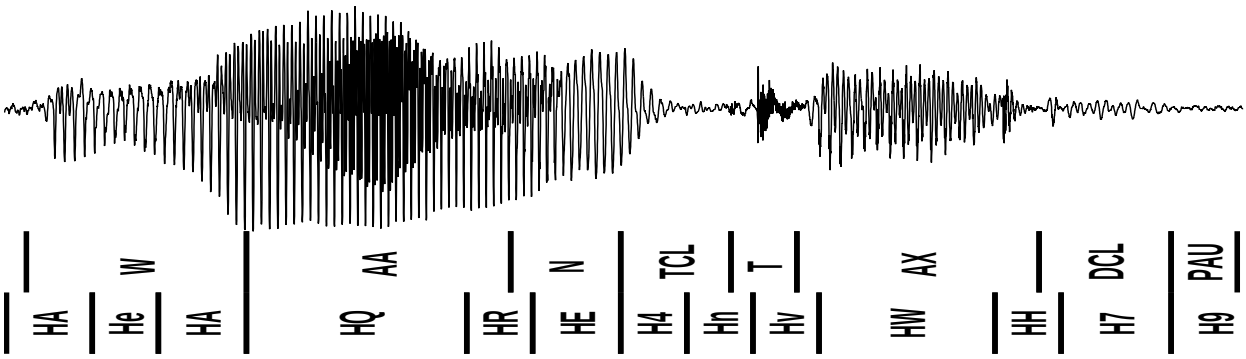

FIG. 11 – Mot "wanted" (locuteur féminin de Boston University corpus). Segmentation en unités phonétiques et unités ALISP.

Les unités ALISP utilisées dans cette expérience étaient celle de la sous-section 3.1 (sans le post-traitement par les multigrammes). Les unités phonétiques sont donnés dans la Table 1. Le corpus distingue les voyelles accentuées des non accentuées mais nous ne nous sommes pas servi de ce classement. L'alphabet des étiquettes phonétiques est similaire à "l'ARPABET" défini pour la base de données classique TIMIT ([8] donne la conversion en alphabet phonétique [1]).

La matrice de confusion résultante (après réarrangement des colonnes pour obtenir une forme pseudo-diagonale) se trouve sur la Figure 13. Il est clair, que cette matrice quantifie la variabilité des unités. Elle montre, que la correspondance est consistante, mais pas biunivoque. Nous pouvons par exemple observer, que l'unité ALISP HA correspond à toutes les clôtures, mais aussi à une pause, et que l'unité H$ présentant une corrélation prononcée à l'unité SH est aussi liée à son correspondant voisé ZH et aux affriquées JH et CH, qui lui sont très proches acoustiquement. Figure 12 présente une comparaison de la voyelle AA avec son correspondant ALISP HQ dans le domaine temporel et sur des spectrogrammes.

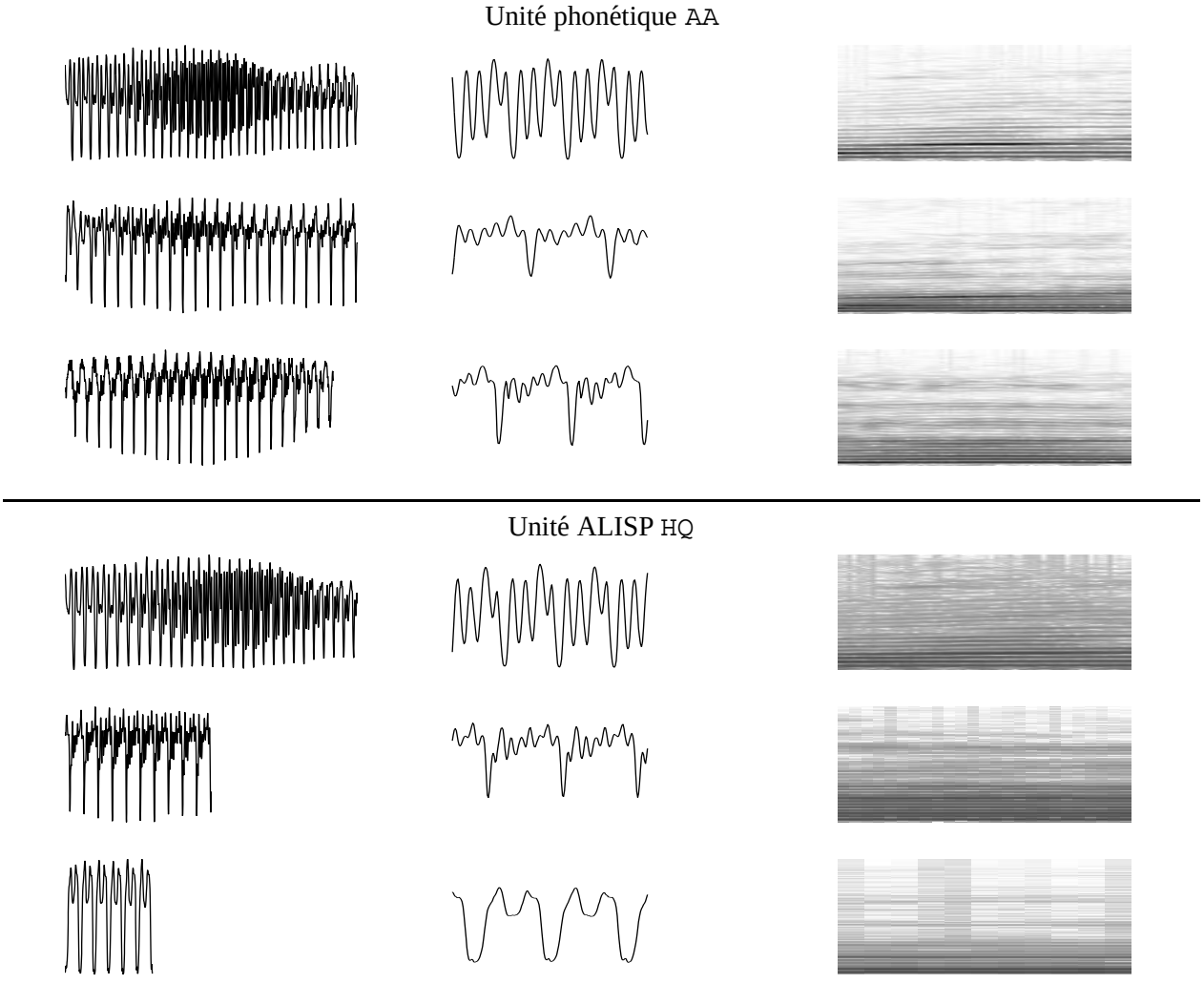

FIG. 12 – Comparaison de l'unité phonétique (3 apparitions) avec son correspondant ALISP (3 apparitions). Gauche : domaine temporel, centre : détail de 200 échantillons, droite : spectrogramme.

Nous avons également étudié la correspondance des unités ALISP avec des classes phonétique larges.

classe phonétique	abbrev.	phonèmes
occlusions	CLO	BCL, DCL, GCL, PCL, KCL, TCL
relâchement d'occlusion (explosion)	OCC	B, D, G, P, T, K, DX
affriquées	AFF	JH, CH
fricatives	FRI	S, SH, Z, ZH, F, TH, V, DH
nasales	NAS	M, N, NG, EM, EN, NX
consonnes vocaliques	DVG	L, R, W, Y, HH, HV, EL
voyelles	VOY	IY, IH, EH, EY, AE, AA, AW, AY, AH AO, OY, OW, UH, UW, ER, AX, AXR
autres	AUT	PAU, H#, brth

TAB. 1 – Le jeu d'unités phonétiques de Boston University, utilisé dans la comparaison monolocuteur.

Les labels des unités phonétiques (Figure 10 ont été remplacées par les labels des classes phonétiques CLO, AFF, FRI, NAS, DVG, VOY, AUT (cf. Table 1), et les recouvrements rélatifs ont été calculés de même façon que précédemment. La matrice de confusion se trouve sur la Figure 14. Nous pouvons observer une cohérence entre les classes des unités ALISP avec les classes phonétiques. Cette correspondance est la plus prononcée pour l'unité H@ liée aux affriquées. Pour mettre en correspondance plus claire les unités ALISP avec les classes phonétiques, nous pourrions effectuer un *clustering* des unités afin que les clusters correspondent le plus aux classes phonétiques.

Pour construire un système de reconnaissance de parole avec les unités ALISP, il nous faudrait trouver une correspondance plusieurs↔plusieurs entre plusieurs phonèmes et plusieurs unités ALISP (les travaux de Deligne [6] sur les multigrammes conjoints apportent des perspectives intéressantes), ou construire le dictionnaire de prononciations directement à l'aide de ces unités (Fukada [7] propose une composition des unités automatiquement apprises, dans les mots et les phonèmes).

4.2 Cas multilocuteur

Pour le corpus de Bref, nous avons compare la segmentation avec des unites ALISP, obenue sur les donnes de 40 locuterus masculins avec une segmetation acoustico-phonetique. La segmentation phonetique etant obtenue par un procedure automatique de phonetisation des textes, suivie d'un alignement Viterbi[4]

En utilisant la meme procedure de mise en correspondance entre les unites ALISP et les unites phonetique, que pour le cas du decrit ci-dessus. Le nombre d'unites ALISP dans ce cas multilocuteur est aussi 64. Il est clair que ce ne sont pas les memes unites ALISP que pour les cas monolocuiteur. L'ensemble de phonemes utilise pour le decodage phonetique est resume dans le tableau 2.

La matrice de confusion résultante (après réarrangement des colonnes pour obtenir une forme pseudo-diagonale) se trouve sur la Figure 15. Il faut remarquer que le nombre d'unites phonetiques utilises dans ce cas est 35. Tnadis que pour les experiences dans le cas monoocuteur, on a 57 unites phonetiques. On peut constater que matrice de confusion represente une plus grande variabilite. Ceci est du a la variabilite de la parole du aux differentes prononciations par differents locuteurs. Dans le cas monolocuteur c'est seulement la variabilite intra-locutuer qui intervient. Cette variabilite inter-locuteur est probablement a l'origine d'une

[4]Les transcrioptioons phonetiques proviennent du projet Sirocco (http ://www.enst.fr/ sirocco). Nous remercions tout particulierement Guillaume Gravier et Francois Yvon de nous avons mis a disposition ces transcriptions .

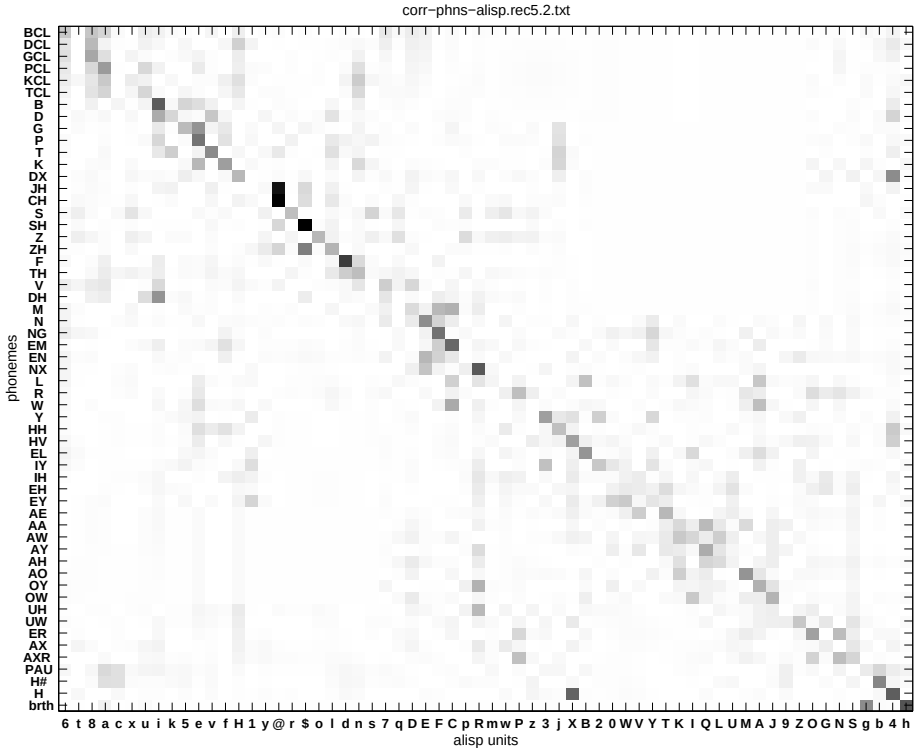

FIG. 13 – Correspondance de la segmentation ALISP avec une segmentation acoustico-phonétique pour le locuteur féminin de Boston University corpus. La couleur blanche correspond à une corrélation zéro, la noire à la valeur maximale de $x_{i,j}$=0.806.

correspondance plus floue entre les unites Alisp et les phones, dans le cas multilocuteur. On peut neanmoins observe une correlation prononcee entre les unites Alisp multilocuteur He et H8 et les les relachemetns d'occlusion (explosion) b, d et g. De meme les unites Alisp Hf et Hi correspondent aux relachemetns d'occlusion (explosion) p,t et k, et a l'occlusion cl. On observe aussi une correspondance prononcee entre les unites Alisp Hs et Ht et les fricatives S et Z, qui sont tres proches acoustiquement. La fricative f a son correspondant Alisp dans l'unite HI. Les nasales M et n sont auusi relativement bien representees par l'unite Alisp HM. Parmi les unites Alisp qui sont les plus confuses, on peut indiquer l'unite H2 (correspondasnt aux phonemes o , A, U , a , k, o, u et w), et l'unite Alisp H4 (correspondant a @, d, g, k et t). Un regroupement des phones en clases phonetiques larges donnnerait probablemant de meilleurs resultats.

5 Conclusions

Un des objectifs de l'Association Phonétique Internationale est de répertorier les unités segmentales (phones) utilisées dans les langues parlées. Un signal de parole quelconque peut alors être transcrit (par un phonéticien) comme une séquence de ces unités. Ce "codage" correspond en moyenne à un débit de l'ordre de 50 bps (correspondant à l'élocution d'environ 10 phones par seconde). Les efforts pour automatiser ce processus (dénommé "décodage acoustico-phonétique" par les spécialistes de la reconnaissance automatique de la parole) n'ont pas encore aboutis.

L'approche proposée dans cet article consiste a découvrir un ensemble d'unités à partir d'échantillons

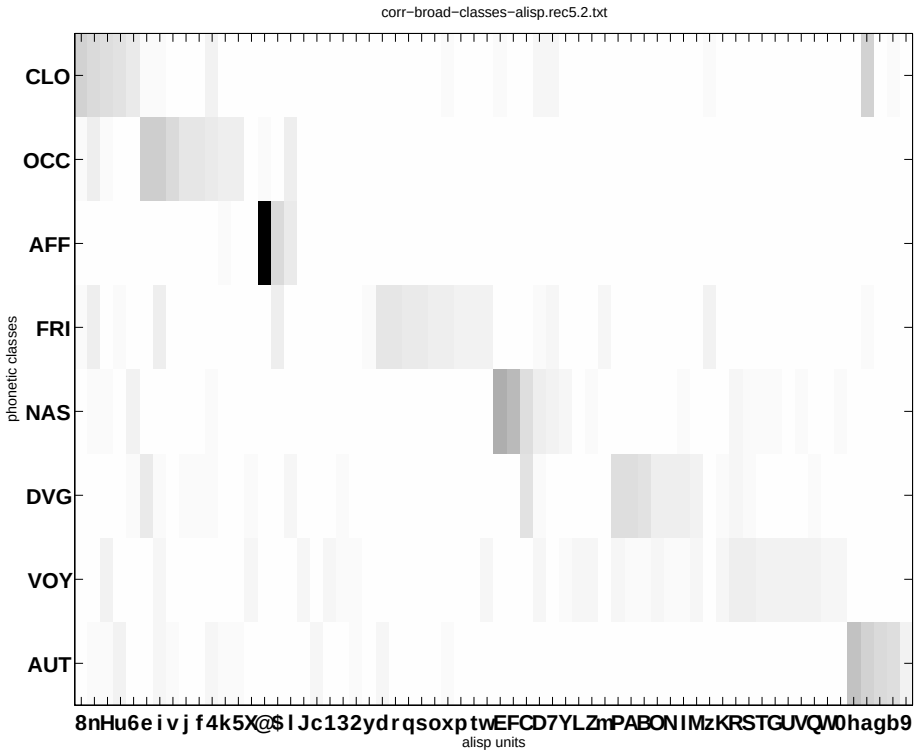

FIG. 14 – Correspondance de la segmentation ALISP avec des classes phonétiques larges pour le locuteur féminin de Boston University corpus. La couleur blanche correspond ‘a une corrélation zéro, la noire à la valeur maximale de $x_{i,j}$=0.7596.

de parole sans aucun a-priori sur la fonction linguistique des unités. L'ensemble des unités est déterminé automatiquement (de manière non-supervisée) sur un corpus d'apprentissage et de validation. Des outils (tels que la décomposition temporelle, la quantification vectorielle, les "multigrammes", la modélisation stochastique, ...) ont été adaptés pour atteindre cet objectif. Une validation a été réalisée sur plusieurs langues en codage de la parole à très bas débit.

Il est alors intéressant d'analyser le degré de correspondance entre une transcription à l'aide des unités ALISP et une transcription phonétique traditionnelle. Le fait que cette correspondance soit imparfaite devrait permettre d'améliorer le codage lexical (et morphologique) en intégrant des variantes découvertes automatiquement sur les nombreuses bases de données pour lesquelles une transcription orthographique est disponible (par exemple SpeechDat [20]).

Références

[1] International Phonetic Association (IPA) homepage. http ://www.arts.gla.ac.uk/IPA/ipa.html.

[2] B. Atal. Automatic speech recognition : a communication perspective. In *Proc ICASSP'99*, pages I–457, 1999.

[3] B. S. Atal. Efficient coding of LPC parameters by temporal decomposition. In *Proc. IEEE ICASSP 83*, pages 81–84, 1983.

classe phonétique	abbrev.	phonèmes
occlusions	CLO	cl, vcl
relâchement d'occlusion (explosion)	OCC	b, d, g, k, p, t
fricatives	FRI	S, Z, f, s, v, z
nasales	NAS	m, n
consonnes vocaliques	DVG	R, j, l, w
voyelles	VOY	@, o~, A, E, U~, O, a~ e, i, u, o, 2, 9, y
autres	AUT	#

TAB. 2 – Le jeu d'unités phonétiques de Bref utilisé dans la comparaison multilocuteur.

[4] F. Bimbot. An evaluation of temporal decomposition. Technical report, Acoustic research department AT&T Bell Labs, 1990.

[5] C. de Marcken. The unsupercised acquisition of a lexicon from continuous speech. Technical Report A.I.Memo No. 1558, C.B.C.L. Memo No. 129, Massachusetts Institute of Technology, Artificial Intelligence Lab. and Center for Biological and Computational Learning, Dpt. of Brain and Cognitive Sciences, November 1996.

[6] S. Deligne. *Modèles de séquences de longueurs variables : Application au traitement du langageécrit et de la parole*. PhD thesis, École nationale supérieure des télécommunications (ENST), Paris, 1996.

[7] T. Fukada, M. Bacchiani, K. Paliwal, and Y. Sagisaka. Speech recognition based on acoustically derived segment units. In *Proc. ICSLP 96*, pages 1077–1080, 1996.

[8] J.S. Garofolo, L.F.Lamel, W.M. Fisher, J.G. Fiscus, D.S. Pallett, and N.L. Dahlgren. DARPA–TIMIT acoustic–phonetic speech corpus. Technical Report NISTIR 4930, U.S. Department of Commerce, National Institute of Standards and Technology, Computer Systems Laboratory, February 1993.

[9] A. Gersho. Advances in speech and audio compression. *Proc. IEEE*, 82(6) :900–918, June 1994.

[10] A. Gersho. *Vector quantization and signal compression*. Kluwer Academic Publishers, 1996.

[11] M. Ismail and K. Ponting. Between recognition and synthesis – 300 bits/second speech coding. In *Proc. EUROSPEECH 97*, pages 441–444, Rhodes, Greece, 1997.

[12] T. Kohonen. *Self organization and associative memories*. Springer Verlag, Berlin, 1984.

[13] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Series in Information Sciences*. Springer Verlag, Berlin, 1997.

[14] J. B. Kruskal. An overview of sequence comparison. In D Sankoff and J. B. Kruskal, editors, *Time Warps, String Edits, and Macromolecules : The Theory and Practice of Sequence Comparison*, pages 1–44, Reading, Massachusetts, September 1983. Addison-Wesley Publishing Co.

[15] L. R. Rabiner and L. W. Schaeffer. *Digital processing of speech signals*. Prentice Hall, 1978.

[16] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, 77(2) :257–286, February 1989.

[17] C.M. Ribeiro and I.M. Trancoso. Application of speaker modification techniques to phonetic vocoding. In *Proc. ICSLP 96*, pages 306–309, Philadelphia, 1996.

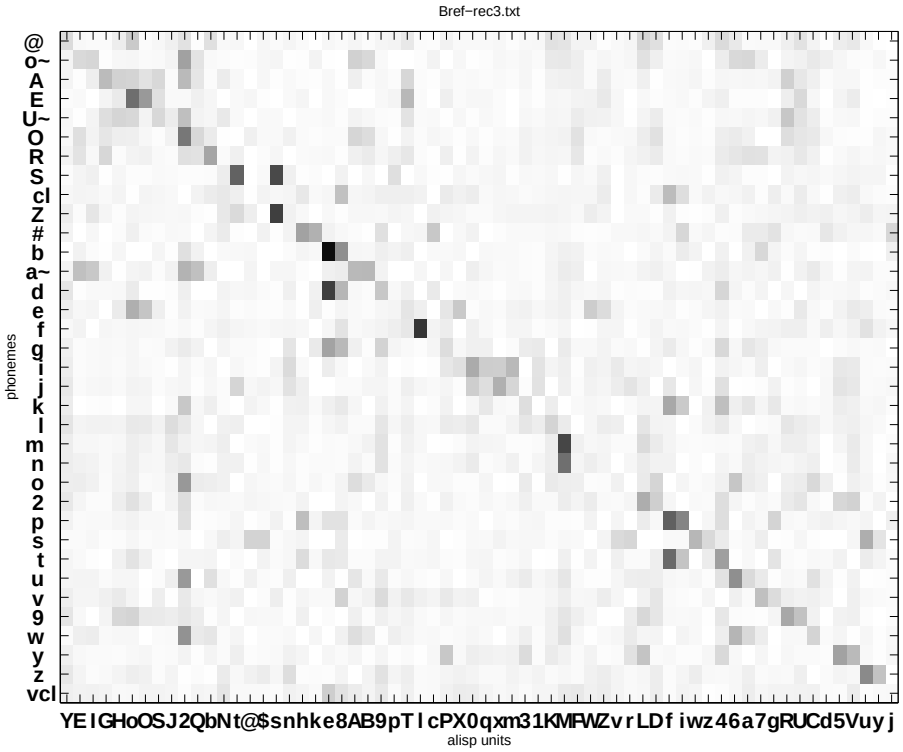

FIG. 15 – Correspondance de la segmentation ALISP avec une segmentation acoustico-phonétique pour les locuteurs masculins du corpus Bref. La couleur blanche correspond à une corrélation zéro, la noire à la valeur maximale de $x_{i,j}$=0.45

[18] C.M. Ribeiro and I.M. Trancoso. Phonetic vocoding with speaker adaptation. In *Proc. EUROSPEECH 97*, pages 1291–1294, Rhodes, Greece, 1997.

[19] J. Černocký. *Traitement de la parole s'appuyant sur des unités segmentales déterminées automatiquement : applications au codage à très bas débit et à la vérification du locuteur*. PhD thesis, Université Paris XI Orsay, December 1998.

[20] R. Winski. Definition of corpus, scripts and standards for fixed networks. Technical report, SpeechDat-II, January 1997. Deliverable SD 1.1.1., workpackage WP1, http ://www.speechdat.org.

[21] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland. *The HTK book*. Entropics Cambridge Research Lab., Cambridge, UK, 1996.