N°=D'ordre : 05/2009-M/IN

République Algérienne Démocratique et populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie
Houari Boumadienne
Faculté d'Electronique et d'Informatique

Mémoire
Présenté pour l'obtention du diplôme de MGISTER
En : Informatique
Spécialité : Systèmes Intelligents et Ingénierie du Logiciel

Par
Amrous Anissa Imene

THEME

Coopération de connaissances dans les modèles de Markov cachés pour la reconnaissance automatique de la parole.

Soutenu le 13/ 12 / 2009, devant le jury composé de :

Mme	M. Boukala	Professeur	USTHB	Présidente
Mr	M. Debyeche	Maître de conférences	USTHB	Directeur de Thèse
Mr	H. Azzoune	Maître de conférences	USTHB	Examinateur
Mr	A. Amrouche	Maître de conférences	USTHB	Examinateur

Résumé

Les systèmes de Reconnaissance Automatique de la Parole (RAP) à base de Modèles de Markov cachées (HMM : Hidden Markov Model, en anglais) utilisent généralement des paramètres cepstraux dits paramètres standards comme modélisation acoustique du signal de parole. Les paramètres cepstraux les plus performants actuellement sont les coefficients MFCCs (Mel Frequency Cepstral Coefficients), les coefficients LPCC (Linear Predictive Cepstrum coefficients) et les coefficients PLP (Perceptual Linear Predictive). Cependant, ces paramètres restent très sensibles aux variations du signal dans les milieux réels. La variabilité est causée par l'environnement (présence de bruit), elle conduit à différentes types de disparités entre observations et modèles acoustiques, ce qui engendre des faibles taux de reconnaissance dans les conditions réelles (bruitées). La sensibilité des paramètres standards à la variabilité du signal à motiver plusieurs chercheurs à utiliser de nouveaux paramètres et de nouveaux paradigmes pour rendre les modèles acoustiques plus robustes. L'objet de notre mémoire de magister est d'intégrer des sources d'informations auxiliaires dans les systèmes de RAP utilisant les HMMs comme moteur de décodage et ceci afin de les rendre plus robustes aux conditions réelles. Les sources auxiliaires proposées dans ce travail sont en relation avec les fondements de la phonation et de la perception humaine. Ces informations sont portées par les paramètres pitch (fréquence fondamental), l'énergie et les fréquences des trois premiers formants. L'incorporation de ces informations auxiliaires dans le système RAP mis en oeuvre a été réalisée par deux types de stratégies de fusion. Dans la première stratégie dite à Identification Directe (ID) ou fusion de paramètres, les deux types de paramètres (standards et auxiliaires) sont concaténés dans le même vecteur pour former une seule observation à l'entrée du système de RAP. Alors que dans la deuxième stratégie dite à Identification Séparée (IS) ou fusion des scores, chaque type de paramètres est modélisé par un sous système de reconnaissance indépendant, les sorties des deux sous systèmes indépendants sont fusionnées en utilisant un réseau de neurones artificiels de type Perceptron Multi Couches (PMC). Les expériences de validation réalisées en mode indépendant du locuteur sur les bases de données ARADIGITS (mots isolés) et TIMIT (parole continue) en milieu bruité à différents niveaux RSB (Rapport Signal/Bruit) ont montré une amélioration des taux de reconnaissance (jusqu'a 7 % de gain pour un RSB de 5 décibels) dans le cas de la fusion de type ID. Ceci motive l'utilisation des paramètres auxiliaires pour la reconnaissance de la parole en environnement réel.

Summary

The state-of-the-art Automatic Speech Recognition (ASR) systems based on Hidden Markov Models (HMMs) use usually cepstral-based features (standard features) as acoustic observation. The most powerful features currently used are the MFCCs (Mel-Frequency Cepstral Coefficients), the LPCC (Linear Predictive Cepstrum Coefficients) and the PLP (Perceptual Linear Predictive). However, these features are very sensitive to speech signal variability under real-life conditions. The speech signal variability is mostly due to environmental factor (presence of noise), it leads to different kinds of mismatch between acoustic features and acoustic models. This causes a reduction on the recognition rate under real-life conditions. The sensitivity of standard features to noise motivates many authors to look for new parameters to make the acoustic models more robust. The main focus of this work is on integrating auxiliary knowledge sources into standard ASR systems so as to make the acoustic models more robust to the variabilities in the speech signal under real-life conditions. The auxiliary knowledge sources that have been investigated in the present work are pitch frequency, energy and the first three formant frequencies. Two main strategies to integrate the auxiliary features into HMM-based ASR system have been studied. In the first strategy called "Direct Integration strategy" the fusion process is based on the concatenation of the auxiliary features with the standard cepstral features in the same acoustic vector, while in the second strategy named "Separate Integration strategy" each type of feature (standard and auxiliary) is modelled by a separate recognition sub-system and the decision is done on the level of the probabilities scores by a MLP *(*Multi Layer Perceptron*)* neuronal network . Very interesting improvements are obtained in noisy environments for different SNR (Signal to Noise Ration). With DI strategy, the improvement reaches a gain of 7% for 5 decibel SNR. These motivate the integration of auxiliary features for speech recognition in real conditions (noisy environment).

Liste des abréviations

ACP	Analyse en Composantes Principales.
AR	Auto-Régressif.
DTW	Dynamic Time Warping (alignement temporel dynamique).
DAP	Décodage Acoustico-Phonétique.
HMM	Hidden Markov Models (modèles de Markov cachés).
HTK	HMM Toolkit (boîte à outils pour modèles de Markov cachés).
GMM	Gaussian Mixture Model.
QV	Quantification Vectorielle.
LPC	Linear Predictive Coding.
MLE	Maximum Likelihood Estimation.
MLP	Perceptrons Multi-Couches (Perceptron Multi-Couches).
MFCC	Mel Frequency Cepstral Coefficient (coefficient cepstral en échelle Mel).
PLP	Perceptual Linear Prediction (prédiction linéaire perceptive).
PMC	Perceptrons Multi-Couches.
RAP	Reconnaissance Automatique de la Parole.
RSB	Rapport Signal sur Bruit.
RNA	Réseaux de Neurones Artificiels.
TCDI	Transformée en Cosinus Discrète Inverse.
TFD	Transformée de Fourier Discrète.

Table des matières

Table des Figures

Liste des tableaux

Introduction générale

Introduction générale

Les systèmes de Reconnaissance Automatique de la Parole (RAP) utilisent généralement des paramètres cepstraux dits paramètres standards comme modélisation acoustique du signal de parole. Les paramètres cepstraux les plus performants actuellement sont les coefficients MFCCs (Mel Frequency Cepstral Coefficients), les coefficients LPCC (Linear Predictive Cepstrum Coefficients) et les coefficients PLP (Perceptual Linear Predictive) avec leurs différentes variantes [Lévy-03]. Cependant, ces paramètres restent très sensibles aux variations du signal dans les milieux réels [Baud-93] [Mary-08] [Hass-02]. La variabilité est causée par l'environnement ou le locuteur et conduit à différents types de disparités entre observations et modèles acoustiques, ce qui engendre des faibles taux de reconnaissance dans les conditions réelles (bruitées).

L'objet ce mémoire est de chercher à améliorer la qualité de la modélisation acoustique, en intégrant des sources d'informations auxiliaires dans les systèmes standards de RAP, afin de les rendre plus robustes à la variabilité du signal de parole.

Cette connaissance additionnelle peut être portée par des paramètres tels que le pitch, les formants, l'énergie, la vitesse d'élocution, etc. La fusion de ces paramètres avec les paramètres standards peut permettre de construire de meilleurs modèles acoustiques, en les rendant moins sensibles à la variabilité du signal de parole.

Cette fusion, donc l'intégration de sources d'information auxiliaires nécessite la résolution de deux problèmes :
1 Quelles types de sources auxiliaires devraient être employées ?
2. Comment devraient-elles être intégrées dans le système RAP standard ?

Dans ce mémoire, nous avons utilisé comme paramètres auxiliaires : la fréquence fondamentale, l'énergie et les fréquences des trois premiers formants. Ces paramètres auxiliaires sont intégrés dans le système de RAP par deux stratégies de fusion inspirées des méthodes de fusion appliquées dans le domaine de la reconnaissance audio-visuelle. Dans la première stratégie dite « à identification direct ou fusion de paramètres», les deux types de paramètres (standards et auxiliaires) sont concaténés dans le même vecteur pour former une seule observation acoustique à l'entrée du système de RAP. La deuxième stratégie dite « à identification séparée ou fusion des scores» consiste à modéliser chaque type de paramètres par un sous système de reconnaissance indépendant, et de fusionner les scores probabilistes à la sortie des deux sous systèmes.

Ce mémoire est organisé comme suit :

Le premier chapitre présente une introduction à la reconnaissance de la parole. Les différentes caractéristiques et la structure générale d'un système de RAP, ainsi que les paramètres les plus efficaces pour représenter le signal de parole et les méthodes de reconnaissance les plus utilisées actuellement.

Dans le deuxième chapitre nous présentons le formalisme des modèles de Markov cachés, leur théorie, leurs principes pour la modélisation de la parole ainsi que les algorithmes d'apprentissage et de décodage permettant la construction d'un système de RAP. Et ceci pour les deux modes de reconnaissance : reconnaissance de mots isolés et reconnaissance de la parole continue.

Le chapitre trois est le chapitre noyau de cette thèse, il présente les paramètres auxiliaires que nous proposons comme vecteur de la connaissance additionnelle, leurs méthodes d'extraction et les différentes stratégies d'intégration dans un système de RAP à base des modèles de Markov cachés.

Dans le chapitre quatre nous présentons le contexte expérimental et les résultats obtenus avec les différents systèmes mis au point ainsi que l'évaluation objective de ces systèmes.

Nous terminons ce manuscrit par une conclusion générale en mettant en évidence le travail réalisé dans ce mémoire. A la lumière des résultats obtenus nous donnons aussi les perspectives futures envisagées.

Reconnaissance Automatique de la Parole

1.1 Introduction

Le but de la Reconnaissance Automatique de la Parole (RAP) consiste à extraire, à l'aide d'un ordinateur, l'information lexicale contenue dans un signal de parole. Depuis plus de deux décennies, des recherches intensives dans ce domaine ont été réalisées par de nombreux laboratoires internationaux. Des progrès importants ont été accomplis grâce au développement d'algorithmes de reconnaissance puissants ainsi qu'aux avancées en sciences cognitives, technologie de l'information, l'intelligence artificielle, le traitement du signal, l'algorithmique, etc.
Ainsi, différents systèmes de reconnaissance de la parole ont été développés, couvrant des domaines aussi vastes que la reconnaissance de quelques mots clés sur lignes téléphoniques, la dictée vocale, la commande et contrôle sur PC, et allant jusqu'aux systèmes de compréhension du langage naturel pour des applications limitées.

Dans ce chapitre nous allons présenter les différentes caractéristiques d'un système de reconnaissance de la parole, la structure générale de ce dernier, les méthodes d'analyse du signal pour une paramétrisation efficiente et enfin les approches de reconnaissance en insistant sur celles les plus utilisées actuellement.

1.2 Caractéristiques d'un système de RAP

La reconnaissance automatique de la parole est un processus qui convertit le signal acoustique de parole en un ensemble de mots ou de phrases. Les systèmes de RAP peuvent être caractérisés selon plusieurs critères [Nguy-02].

1.2.1 Le mode de prononciation

Il existe trois modes de prononciation distincts :

- Prononciation mots isolés : chaque mot est prononcé isolément, une pause de durée marquée sépare les mots.
- Prononciation mots connectés : le système reconnaît des séquences de quelques mots sans pause volontaire entre eux (exemple : reconnaissance de chiffres connectés).
- Parole continue : les mots sont prononcés naturellement sous forme de phrases aux énoncés plus ou moins longs.

1.2.2 Le mode de reconnaissance

Les systèmes de reconnaissance de la parole se distinguent suivant trois types d'utilisations différentes :

- Dépendant du locuteur: le système est adapté à la voix d'un locuteur particulier.
- Multi locuteurs : le système est adapté aux vois de plusieurs locuteurs.
- Indépendant du locuteur : le système n'est pas adapté à la voix d'un locuteur particulier et peut être utilisé par un locuteur quelconque (tout locuteur).

1.2.3 La taille du vocabulaire

Le vocabulaire d'un système de RAP est l'ensemble des mots que le système est capable de reconnaître. Il est évident que la performance du système diminue quant la taille du vocabulaire augmente [Deb-07]. Suivant la taille du vocabulaire, les systèmes de RAP sont classés en trois catégories :

- Système de RAP à petit vocabulaire (entre 10 à 100 mots).
- Système de RAP à moyen vocabulaire (entre 100 à 1000 mots).
- Système de RAP à grand vocabulaire (plus de 1000 mots).

1.2.4 La syntaxe

Dans un système de RAP, la syntaxe spécifie les contraintes imposées sur les suites de mots prononcés. Elle peut simplifier la tâche du système de reconnaissance en lui fournissant des connaissances supplémentaires. La syntaxe peut être de type :

- Grammaire syntaxique qui impose des règles à suivre pour la construction des phrases à reconnaitre [Youn-89].

- Des modèles de type N-grammes qui veut dire que tout mot peut être suivi par une séquence de N-1 autres avec une probabilité fixée à l'avance [El-Bè-90]. En général le type Bi-grammes est le plus couramment utilisé en reconnaissance de la parole.

1.2.5 L'environnement

Les conditions environnementales (bruit, ligne téléphonique, qualité des microphones, etc.) peuvent influer sur la performance globale d'un système de reconnaissance. La plupart des systèmes de reconnaissance fonctionnent relativement bien dans un environnement calme mais les performances se dégradent notablement si les conditions environnementales sont très bruitées (réelles).

1.3 Architecture d'un système de RAP

L'architecture générale d'un système de RAP est donnée par la figure 1.1 suivante. Dans cette structure, on trouve deux modules qui sont l'extraction de paramètres (feature extraction) et la reconnaissance (ou classification).

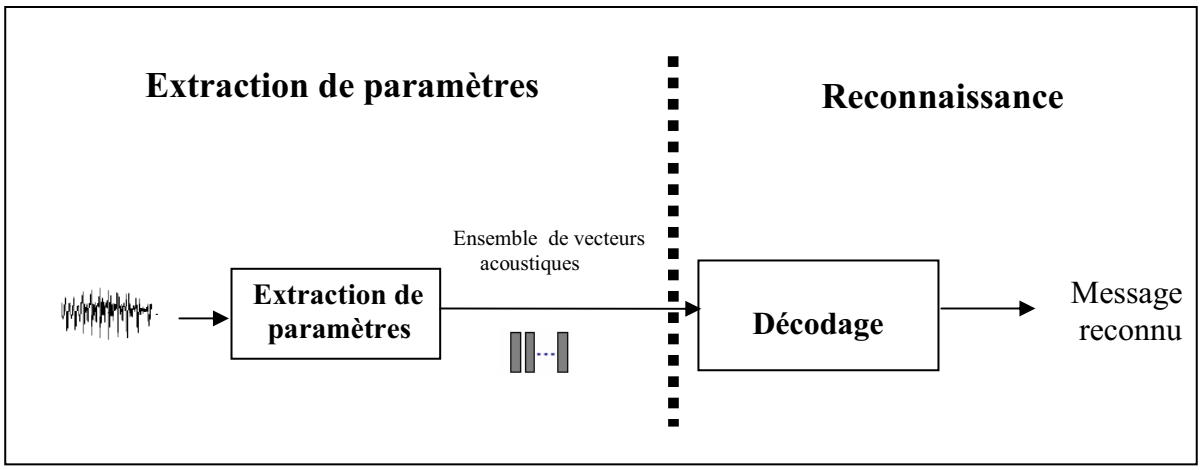


Figure 1.1 : Structure générale d'un système de RAP

1.3.1 Extraction des paramètres

Le signal de parole présente de la redondance et il est porteur de plusieurs types d'informations, ce qui justifie la recherche d'une représentation spécifiquement pertinente pour la reconnaissance. L'extraction des paramètres consiste à associer au signal de parole une série de vecteurs de paramètres acoustiques. Cette extraction débute par une mise en forme du signal de parole suivie par le calcul de coefficients acoustiques comme indiqué sur la figure 1.2 suivante :

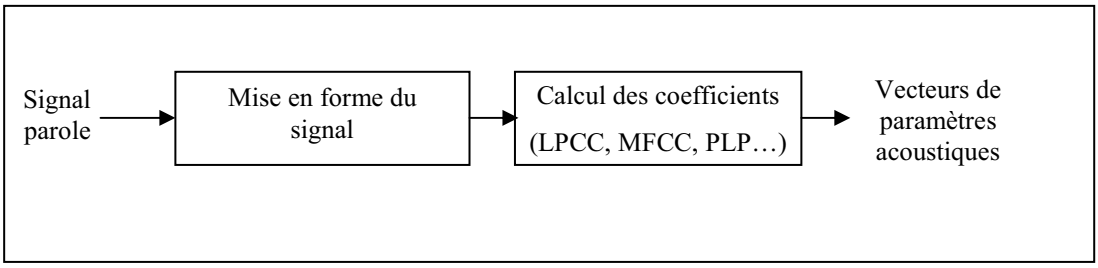


Figure 1.2 : Schémas synoptique de l'extraction de paramètres acoustiques.

1.3.1.1 Mise en forme du signal de parole

L'étape de mise en forme de signal de parole consiste à lui appliquer une série d'opérations de traitement du signal dont le but est de l'adapter au calcul de coefficients acoustiques. La figure 1.3 illustre ces différentes opérations.

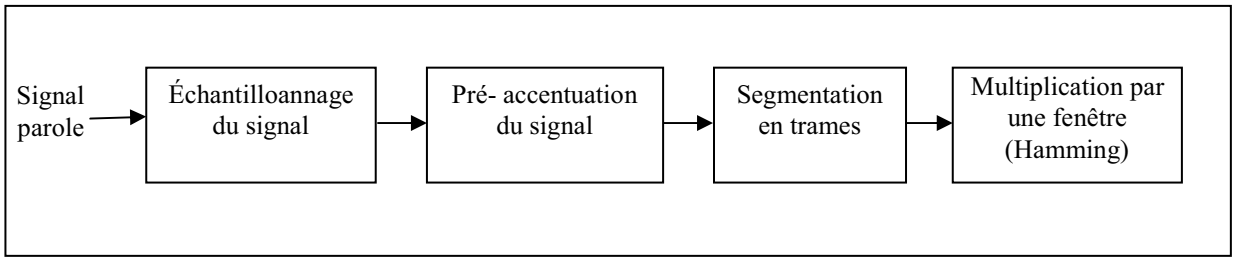


Figure 1.3 : Etapes de mise en forme du signal de parole.

- Echantillonnage : effectué à une fréquence f_e compatible avec les exigences du théorème de Shannon, la perte d'information entre le signal continu et le signal discret correspondant est nulle si et seulement si la fréquence d'échantillonnage est au moins supérieure ou égale au double de la fréquence la plus haute contenue dans ce signal.

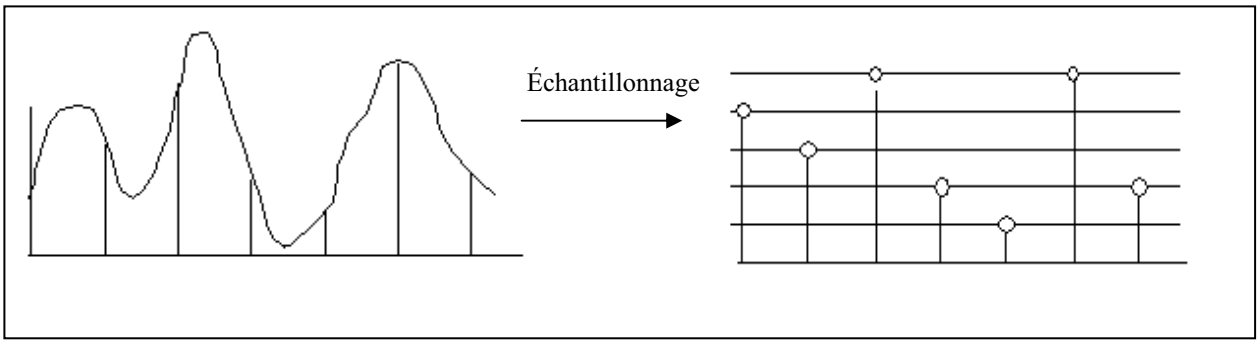


Figure 1.4: Exemple d'échantillonnage d'un signal de parole.

- **Pré- accentuation** : elle consiste à relever les hautes fréquences qui sont moins énergétiques que les basses fréquences, la pré- accentuation s'_n de l'échantillon sn à l'instant n est calculée par l'équation 1.1 (la valeur de α est généralement prise entre 0,9 et 1):

$$s_n' = s_n - \alpha . s_n - 1 \tag{1.1}$$

- **Segmentation en trame** : cette opération consiste à découper le flot de parole continue en trames pendant lesquelles le signal est supposé quasi-stationnaire. Chaque trame a habituellement une durée identique d'environ 20 à 30 ms, et le décalage entre deux trames consécutives est d'environ 15 à 10 ms (figure 1.5).

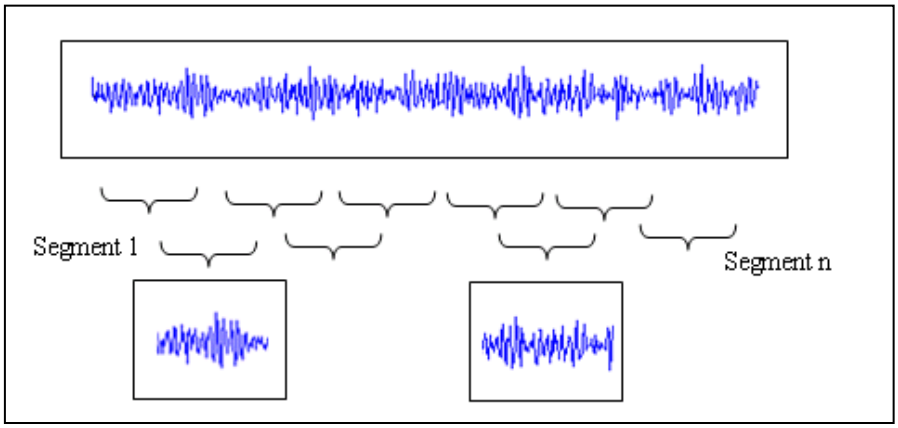


Figure 1.5 : Segmentation du signal parole en trames.

- Multiplication par une fenêtre : pour réduire les effets de bord produits par la segmentation, les trames sont alors multipliées par une fenêtre de pondération (équation 1.2). Un exemple de fenêtres de pondération utilisées en reconnaissance, est la fenêtre de Hamming dont la formule est donnée par l'équation 1.3.

$$s''_n = w_n s'_n \tag{1.2}$$

Avec

$$w_n = 0{,}54 - 0{,}46.\cos(2\pi \frac{n}{N-1}), \qquad 0 \le n \le N-1 \tag{1.3}$$

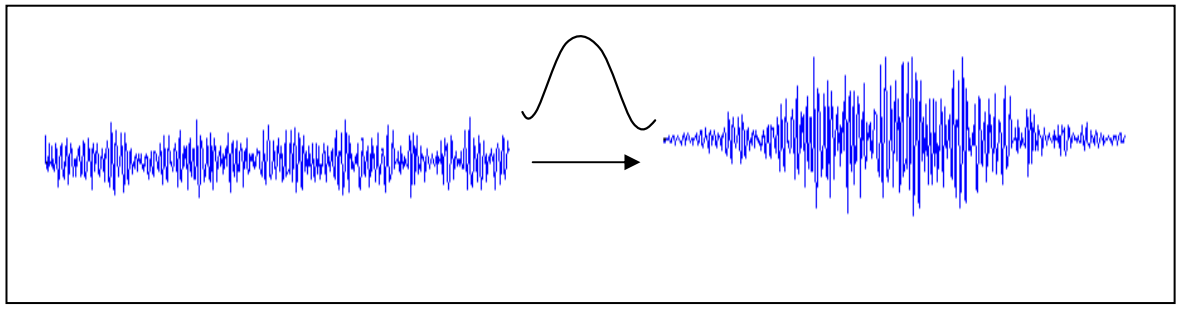


Figure 1.6: Pondération d'une trame du signal par la fenêtre de Hamming

1.3.1.2 Le calcul des coefficients

Après l'étape de mise en forme, le signal est prêt pour être paramétré. En reconnaissance de la parole, les paramètres extraits doivent être [jaco-95] :

- **pertinents** : extraits de mesures suffisamment fines, ils doivent être précis mais leur nombre doit rester raisonnable afin de ne pas avoir de coût de calcul trop important dans le module de décodage.

- **discriminants** : ils doivent donner une représentation caractéristique des sons de base et les rendre facilement séparables.

- **robustes** : ils ne doivent pas être trop sensibles à des variations de niveau sonore ou à un bruit de fond.

Dans la littérature il existe plusieurs types de coefficients. Dans ce qui suit nous décrivons les trois principales techniques de paramétrisation qui ont montré un grand intérêt dans les applications de la reconnaissance de la parole. Ces techniques sont l'analyse LPC (Linear Predictive Coding), les coefficients PLP (Perceptual Linear Predictive) et les coeffficients MFCC (Mel Frequency Cepstral Coefficients).

1.3.1.3 L'analyse LPC

La théorie du codage par prédiction linéaire appliquée à la parole figure parmi les techniques les plus utilisées en traitement automatique de la parole [Mark-76].

Principe de la prédiction linéaire
La prédiction linéaire est basée sur l'hypothèse que chaque échantillon du signal original *s(n)* peut être approché par une combinaison linéaire des *p* échantillons qui le précédent [Dekk-03] :

$$s(n) = \left[\sum_{i=1}^{P} - a_i s(n-i) \right] + e(n) = \hat{s}(n) + e(n) \tag{1.4}$$

Dans cette expression, $\hat{s}(n)$ est l'échantillon prédit, les coefficients *a(i)* sont appelés coefficients de prédiction d'ordre *p* et le terme e(n) est l'erreur de prédiction d'ordre *p*, définie par l'équation (1.5) suivante:

$$e(n) = s(n) - \hat{s}(n) = s(n) + \sum_{i=1}^{P} a(i) s(n-i) = \sum_{i=0}^{P} a(i) s(n-i) \tag{1.5}$$

Tel que $a_0 = 1$.

En cherchant à estimer les coefficients de prédiction a_i tout en minimisant l'erreur *e(n)*. La transformée en Z de l'équation (1.5) est écrite comme suit :

$$E(z) = S(z) + \sum_{i=1}^{P} a_i S(z) z^{-1} = S(z) \left[1 + \sum_{i=1}^{P} a_i z^{-i} \right] = S(z) A(z) \tag{1.6}$$

Où :

$$A(z) = 1 + \sum_{i=1}^{P} a_i z^{-i} = \sum_{i=0}^{P} a_i z^{-i} \tag{1.7}$$

L'équation (1.6) peut être écrite comme :

$$S(z) = E(z)\frac{1}{A(z)} = E(z)H(z) \tag{1.8}$$

Ceci montre que le signal parole peut être vu comme la sortie d'un filtre auto-régressif tout pôles (all-pole digital filter) dont la fonction de transfert est définie par :

$$H(z) = \frac{1}{A(z)} \tag{1.9}$$

Et dont l'entrée du filtre est l'erreur *E(z).*

Estimation des coefficients de prédiction

L'estimation des coefficients de prédiction est basée sur la minimisation de l'erreur quadratique moyenne de prédiction E. Pour une trame de longueur [n0, n1], l'erreur E est calculée comme suit [Boit-00] [Dekk-03] :

$$\mathrm{E} = \sum_{n=n_0}^{n_1} e^2(n) \tag{1.10}$$

La substitution dans l'équation (1.10) donne :

$$\mathrm{E} = \sum_{n=n_0}^{n_1}\left[\sum_{i=0}^{P} a_i s(n-i)\right]^2 = \sum_{n=n_0}^{n_1}\sum_{i=0}^{P}\sum_{j=0}^{P} a_i s(n-i)s(n-j)a_j = \sum_{i=0}^{P}\sum_{j=0}^{P} a_i c_{ij} a_j \tag{1.11}$$

Où

$$c_{ij} = \sum_{n=n_0}^{n_1} s(n-i)s(n-j) \tag{1.12}$$

Pour minimiser la fonction E, nous calculons sa dérivée partielle par rapport aux coefficients de prédictions a_i (c'est une méthode d'optimisation classique):

$$\frac{\partial E}{\partial a_k} = 2\sum_{i=0}^{p} a_i c_{ik} = 0 \qquad k = 0,1,....,p; \quad i = 0,1,....,p \tag{1.13}$$

Puisque $a_0 = 1$, l'équation (1.13) devient l'équation normale :

$$\sum_{i=1}^{P} a_i c_{ik} = -c_{0k} \qquad k = 1,2....,p \tag{1.14}$$

Ces équations normales, dites de Yule-walker, constituent un système linéaire de *P* équations à *P* inconnues. La résolution de ce système permettra d'obtenir les coefficients de prédiction a_i. Parmi les méthodes de résolution de ce système, on peut citer la méthode d'autocorrélation, la méthode de covariance, la méthode de burg [Keil-00].

Ordre de prédiction linéaire

En pratique l'ordre du filtre de prédiction (le nombre de coefficients a_i) est choisi généralement en fonction du nombre de formants R [ALL-91]. Il est donné par la relation :

$$p = 2R + 2\,. \tag{1.15}$$

Un ordre *p* entre 12 et 16 s'avère le plus représentatif dans l'analyse de la parole.

1.3.1.4 Les coefficients PLP (Perceptual Linear Predictive)

La méthode PLP est une méthode inspirée du principe de prédiction linéaire (LP, Linear Predictive). Elle combine ce principe à une représentation du signal qui suit l'échelle humaine d'audition. Son but est d'estimer les paramètres d'un filtre auto-régressif tout pôle, modélisant au mieux le spectre auditif [Deb-07]. Le processus de calcul des coefficients PLP peut être décrit par la figure 1.7 suivante :

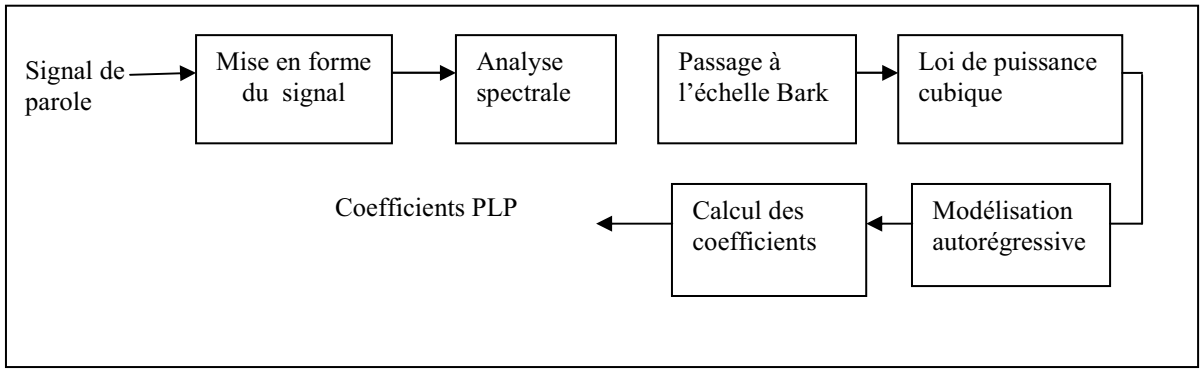


Figure 1.7: Processus de calcul des coefficients PLP.

Après la mise en forme du signal de parole, une analyse spectrale est effectuée pour obtenir le spectre du signal temporel en lui appliquant une TFR (Transformer de Fourier Rapide) et d'une TFR inverse comme suit (figure 1.8):

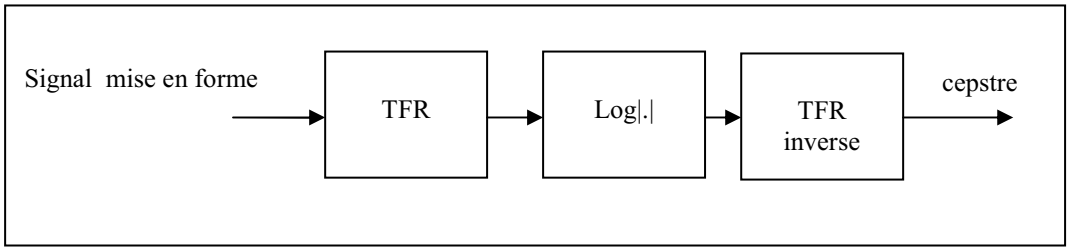


Figure 1.8 : Le calcul de spectre d'un signal de parole.

Ensuite, un passage de l'échelle de fréquence usuelle à l'échelle de Bark **(**équation 1.16**)** est effectué.

$$\Omega(\omega) = 6\ln\left(\frac{\omega}{1200\pi} + \left(\left(\frac{\omega}{1200\pi}\right)^2 + 1\right)^{0.5}\right) \qquad (1.16)$$

ω représente la fréquence angulaire exprimée en rd/s et Ω la fréquence de Bark.

Ce passage à l'échelle Bark, permet d'approximer la forme des filtres auditifs. Le spectre de puissance dans l'échelle de Bark est convolué avec le spectre de puissance de la courbe de bande critique en utilisant l'équation 1.17 suivante :

$$\Psi(\Omega) = \begin{cases} 0 & \Omega < -1.3 \\ 10^{2.5(\Omega+0.5)} & -1.3 \le \Omega \le -0.5 \\ 1 & -0.5 \le \Omega \le 0.5 \\ 10^{-1(\Omega-0.5)} & 0.5 \le \Omega \le 2.5 \\ 0 & \Omega \ge 2.5 \end{cases} \quad (1.17)$$

On essaye ensuite d'approximer la sensibilité de l'oreille humaine à différentes fréquences par l'intermédiaire d'une fonction de transfert *E(ω)*. Le spectre de puissance est multiplié par cette fonction de transfert.

$$E(\Omega) = E(\omega).\Theta(\Omega) \quad (1.18)$$

$$\Theta(\Omega_t) = \sum_{\Omega=-1.3}^{\Omega=2.3} P(\Omega - \Omega_t).\Psi(\Omega) \quad (1.19)$$

La non-linéarité entre l'intensité d'un son et sa force de perception par l'oreille est ensuite approximée par une loi de puissance :

$$\Phi(\Omega) = E(\Omega)^{0.33} \quad (1.20)$$

L'étape finale consiste en une modélisation auto-régressive classique du spectre du modèle auditif tout pôle, en calculant les coefficients auto-régressifs du filtre.

1.3.1.5 Les coefficients MFCCs (Mel-scale Frequency Cepstral Coefficients)

Les coefficients MFCC sont une extension des coefficients cepstraux par le passage de l'échelle fréquentielle linéaire à une échelle fréquentielle non linéaire dite l'échelle Mel [Deb-07]. La fréquence mel-échelle est définie par:

$$B(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (1.21)$$

Où *f* est la fréquence en Hz, *B(f)* est la fréquence mel-échelle de *f*.

L'intérêt de l'échelle Mel est d'être assez proche d'échelles issues d'études sur la perception sonore et sur les bandes passantes critiques de l'oreille [Barr-96].

Le calcul des paramètres MFCC se réalise de la façon suivante (figure 1.9):

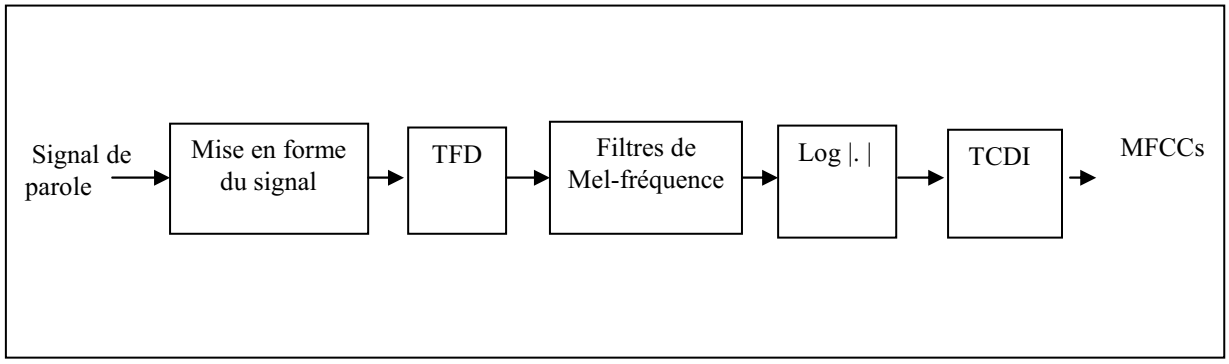


Figure 1.9: Calcul des MFCCs.

Après la mise en forme du signal, une TFD (Transformée de Fourier Discrète) est calculée pour faire passer le signal de parole dans le domaine spectral :
Pour un signal discret $\{s[n]\}$ avec $0 < n < N$, *où* N est le nombre d'échantillons d'une fenêtre analysée, F_s est la fréquence d'échantillonnage, la transformée de Fourier discrète $S[k]$ est obtenue par :

$$s[k] = \sum_{n=0}^{N-1} x[n] e^{-j 2 \Pi nk / N} \tag{1.22}$$

Le spectre du signal est multiplié avec des filtres triangulaires (figure 1.10) dont les bandes passantes sont équivalentes en domaine mel-fréquence. Les points frontières $B[m]$ des filtres en mel-fréquence sont calculés ainsi :

$$B[m] = B(f_l) + m \frac{B(f_h) - B(f_l)}{M+1} \qquad 0 \le m \le M+1 \tag{1.23}$$

Où M est le nombre de filtres, f_h est la fréquence la plus haute et f_l est la fréquence la plus basse pour le traitement du signal.

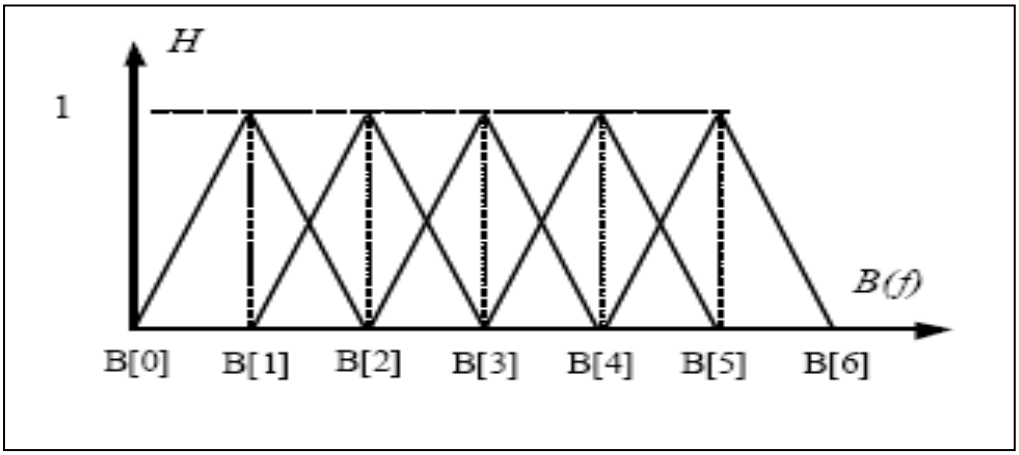


Figure 1.10: Les filtres triangulaires passe-bande en Mel-Fréquence (B(f)).

Dans le domaine fréquentiel, les points $f[m]$ discrets correspondants sont calculés par l'équation :

$$f[m] = \left(\frac{N}{F_s} \right) B^{-1} \left(B(f_l) + m \frac{B(f_h) - B(f_l)}{M+1} \right) \tag{1.24}$$

Où B^{-1} est la transformée de mel-fréquence en fréquence.

$$B^{-1}(m) = 700 * (10^{m/2595} - 1) . \tag{1.25}$$

Le coefficient $H_m[k]$ de chaque filtre est déterminé par le système suivant :

$$H_m[k]=\begin{cases} 0 & si \quad k \le f[m-1] \\ \dfrac{k-f[m-1]}{f[m]-f[m-1]} & si \quad f[m-1] \le f[m] \\ \dfrac{f[m+1]-k}{f[m+1]-f[m]} & si \quad f[m] \le k \le f[m+1] \\ 0 & si \quad k \ge f[m+1] \end{cases} \tag{1.26}$$

Pour un spectre lissé et stable à la sortie des filtres, le logarithme de spectre d'amplitude est calculé :

$$E[m]=\log\left[\sum_{k=0}^{N-1}|S[k]|^2 H_m[k]\right] \qquad 0 \le m \le M \tag{1.27}$$

Les coefficients cepstraux de mel-fréquence (MFCCs) seront obtenus par une TCDI (Transformée en Cosinus Discrète Inverse) qui permet d'obtenir des coefficients peu corrélés à partir des coefficients aux sorties des filtres :

$$c[n]=\sum_{m=0}^{M-1} E[m]\cos\left(\frac{\pi n(m+\frac{1}{2})}{M}\right) \qquad 0 \le n \le M \tag{1.28}$$

Une dizaine de coefficient MFCCs sont généralement considérés comme suffisants pour les expériences de reconnaissance de la parole [Barr-96].

1.3.1.6 Calcul des coefficients différentiels

Afin de prendre en compte la dynamique du signal, nous ajoutons aux paramètres acoustiques les coefficients différentiels (ou coefficients delta) du premier et du second ordre. Soit le vecteur acoustique à N composantes $C_t=\{c_t^1, c_t^2, ..., c_t^N\}$. Les coefficients delta de premier ordre sont alors estimés par :

$$\Delta C_t=\frac{\sum_{K=-L}^{L} KC_t}{\sum_{K=-L}^{L} K^2} \tag{1.29}$$

Les coefficients du second ordre sont calculés en itérant deux fois l'expression (1.29). La figure 1.11 schématise un exemple du module typique d'extraction de paramètres acoustiques pour un système de RAP.

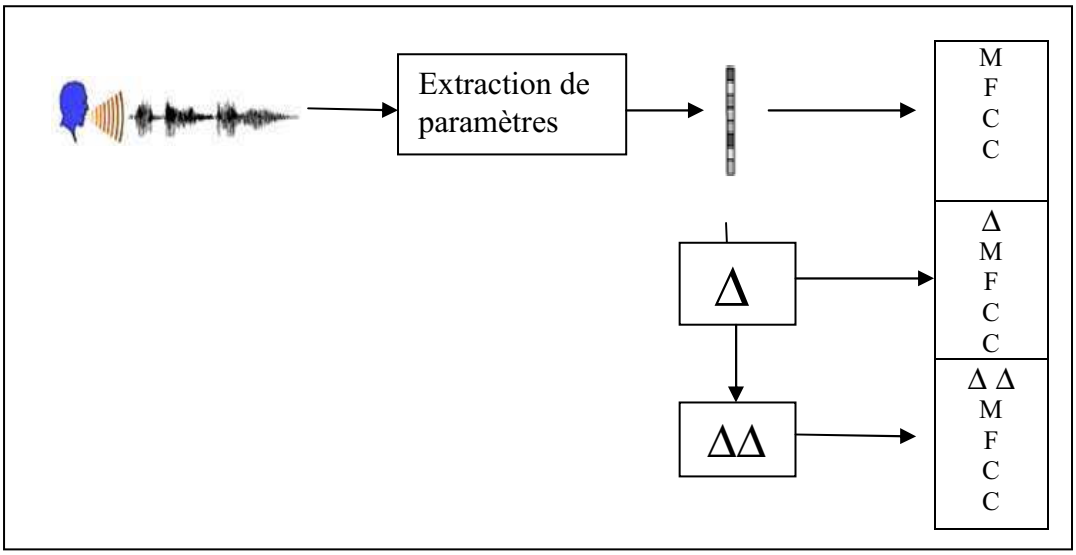


Figure 1.11 : Module typique d'extraction de paramètres standards (MFCC) pour la RAP.

Du point de vue de l'étude bibliographique [Barr-96] [Jaco-95] [Igou-98] [Youn-97], à l'heure actuelle, le choix des paramètres MFCCs semble être le plus satisfaisant pour représenter le signal de parole dans le cadre de la RAP. D'où nous avons choisi cette paramétrisation pour construire notre système de reconnaissance.

1.3.2 Approche de reconnaissance

Plusieurs approches de reconnaissance ont été utilisées. Nous pouvons citer l'approche basée sur la programmation dynamique connue sous l'acronyme DTW (Dynamic Time Warping) [Bell-57], l'approche à base de connaissance utilisant les systèmes experts [Klat-77], l'approche probabiliste à base de modèles de Markov cachés (Hidden Markov Models : HMM en anglais) et l'approche hybride combinant les modèles HMMs et les réseaux de neurones artificiels [Morg-95] [Hoch-94]. L'approche de reconnaissance qui s'appuie sur l'utilisation des modèles de Markov cachés reste à l'heure actuelle la plus utilisée. Elle a permis de nombreuses avancées dans le domaine de la reconnaissance de la parole continue et de la reconnaissance multi locuteurs. C'est cette approche qui sera utilisée par notre système de reconnaissance et qui sera donc suffisamment développée ultérieurement.

1.3.2.1 L'approche DTW

L'approche DTW est fondée sur un principe de comparaison du signal à reconnaitre à des formes de références. Elle a été utilisée avec succès pour la première fois en reconnaissance de la parole par [Sako-78]. Elle consiste principalement à calculer une mesure de distance dynamique entre la forme à reconnaitre (le signal de test) et les formes de références. Le signal identifié correspond à la forme de référence la plus proche en terme de distance calculée.

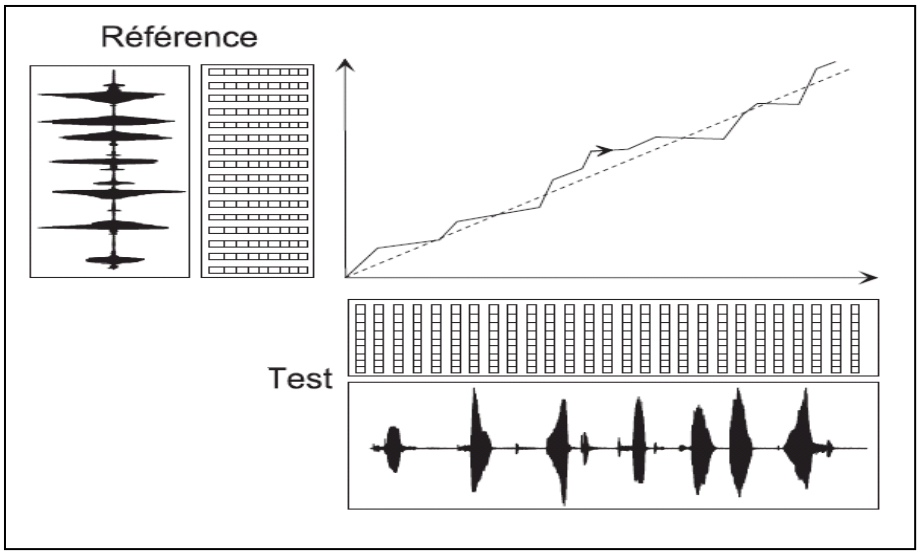


Figure 1.12: Calcul de la distance dynamique par DTW.

Soient $R=(r_1,....,r_I)$ une référence de mot et $T=(t_1,....,t_J)$ le mot à reconnaitre de longueurs respectives I et J. Nous appelons $d\ (r_i,t_j)$ la distance entre les vecteurs acoustique r_i et t_j. Plusieurs types de distances peuvent être utilisés en fonction des méthodes de paramétrisation retenues, nous pouvons citer à titre d'exemple :

- La distance euclidienne est plutôt utilisée dans les systèmes à base d'analyse cepstrale.

$$d\ (r_i,t_j)=\left(\sum_{k=0}^{P}|r_k-t_k|^2\right)^{\frac{1}{2}} \tag{1.30}$$

- La distance d'Itakura est plutôt utilisée dans le cadre d'une paramétrisation par prédiction linéaire.

$$d\ (r_i,t_j)=\log\left[\frac{r_i^t R_b r_i}{t_j^t R_b t_j}\right] \tag{1.31}$$

Où R_b représente la matrice des coefficients d'autocorrélation évalués sur le segment t_j.

Contraintes locales

Afin de tenir compte des réalités physiques du mécanisme de production de la parole, les déplacements entre les vecteurs de paramètres sont limités (contraintes locales). Les contraintes locales les plus courantes sont représentées dans la figure 1.13.

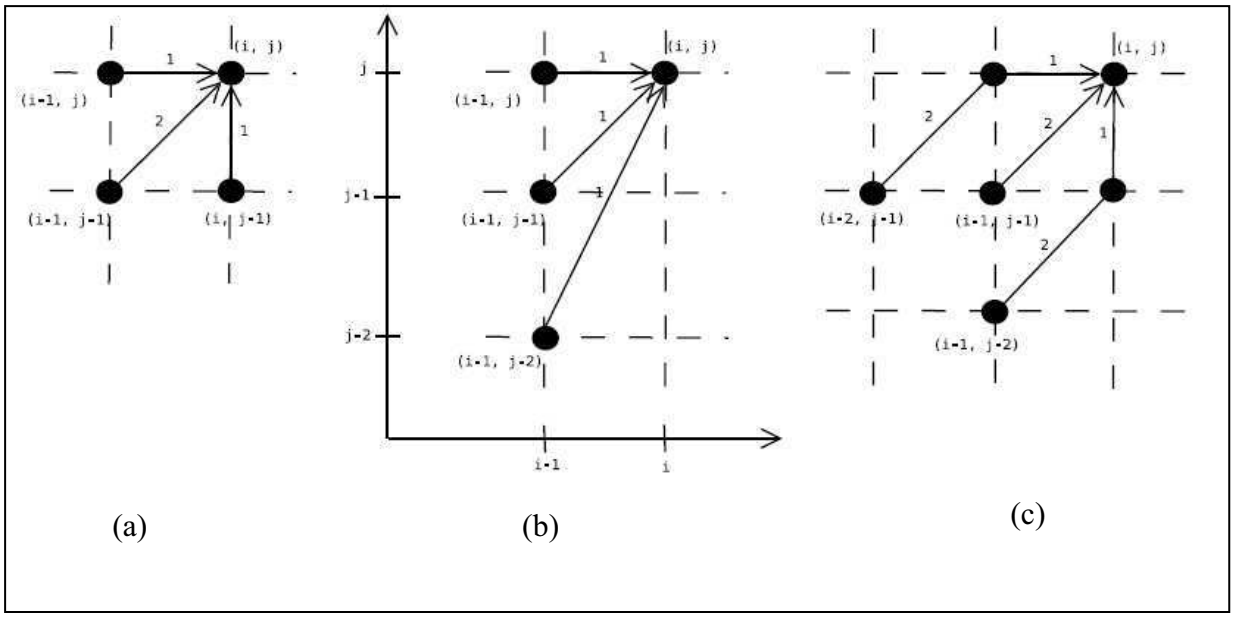


Figure 1.13 : Contraintes locales utilisées dans la DTW.

Le principe est donc de trouver le chemin d'alignement ayant un coût minimum. La distance cumulée *g(i,j)* est d´efinie (en fonction de la contrainte locale choisie - ici 1.13.a) par :

$$g(i,j) = \min \begin{bmatrix} g(i-1,j) + d(i,j) \\ g(i-1,j-1) + 2 * d(i,j) \\ g(i,j-1) + d(i,j) \end{bmatrix} \tag{1.32}$$

En normalisant par les longueurs de R et T, on obtient donc :

$$D(R,T) = \frac{g(I,J)}{I+J} \tag{1.33}$$

1.3.2.2 L'approche probabiliste

Cette approche est aujourd'hui classique et incontournable. Elle est à la base de la majorité des systèmes de RAP actuels. Elle a était proposée pour la première fois par F. Jelinek [Jeli-76] qui a proposé une formulation probabiliste simple du problème de la RAP.

1.3.2.2.1 Principe

La formulation probabiliste du problème de la RAP est la suivante :
Soient M_1 $M_{2....}$ M_R les R modèles probabilistes correspondent aux R phrases (ou mots) prononcées (w_1, $w_{2,....}$ w_R).
Soit : O= o_1 $o_{2....}$ o_T la suite d'observations acoustiques (vecteur acoustique, par exemple de type MFCC) du mot à reconnaitre.

La tâche de la RAP consiste à déterminer, parmi tous les modèles possibles (M_1 $M_{2....}$ M_R), le modèle M_{best} 1<=best<=R, le plus probable connaissant l'observation acoustique O :

$$M_{best} = \arg\max_{M} \quad P(M/O) \tag{1.34}$$

Grâce à la règle de Bayes, il est possible d'écrire :

$$P(M/O) = \frac{P(O/M)P(M)}{P(O)} \tag{1.35}$$

Avec :
P(*O/M*) : la probabilité d'émettre l'observations *O*, étant donnée la suite de modèles *M*, est estimée par la modélisation acoustique.
P(*M*) : la probabilité de la suite de mots *M*, est estimée par le modèle linguistique.
P(O) : représente la probabilité d'occurrence de la suite d'observations acoustiques O. Elle est indépendante de *M* et elle reste donc constante lorsque *M* varie.

Comme *P(O)* ne dépend pas de *M*, les deux équations précédentes peuvent être réduites à :

$$P(M/O) = P(O/M)P(M) \tag{1.36}$$

$$M_{best} = \arg\max_{M} \quad P(O/M)P(M) \tag{1.37}$$

L'approche probabiliste permet donc d'intégrer les niveaux acoustiques et linguistiques dans un seul processus de décision. Ces niveaux sont classiquement représentés par des modèles de Markov cachés. Les unités acoustiques modélisées peuvent être des mots ou des unités plus courtes telles que le phonème.

1.3.2.2.2 Structure d'un système de RAP probabiliste

Un système de reconnaissance de la parole contient un module d'extraction des paramètres acoustiques, un module de reconnaissance et trois sources d'informations : les modèles acoustiques (HMM), le lexique et le modèle de langage (figure 1.14).

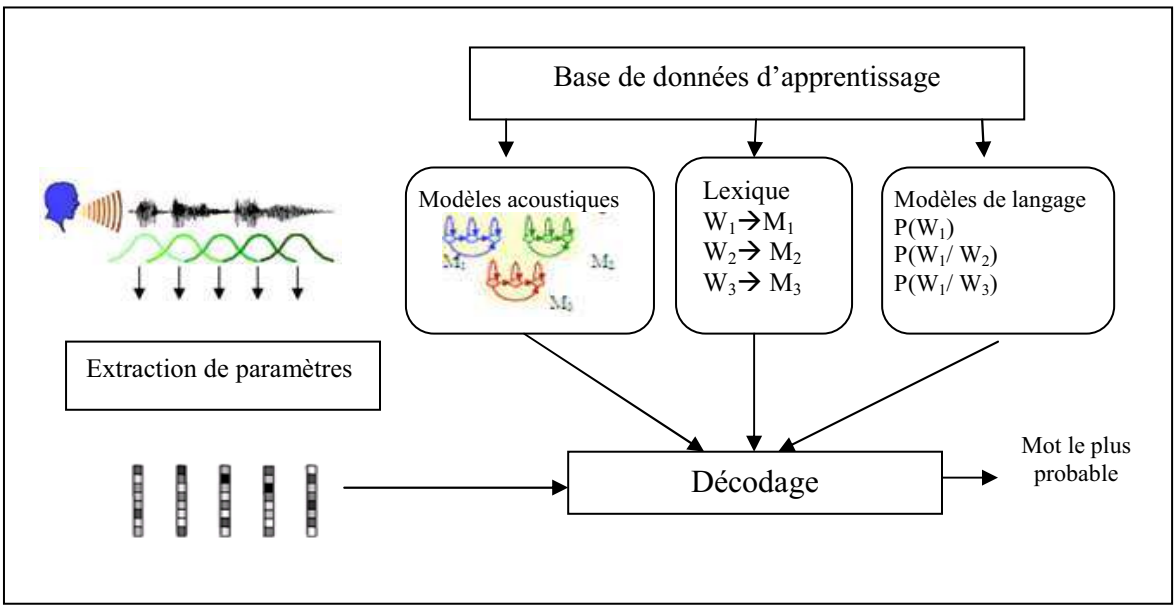


Figure 1.14: Structure d'un système de la RAP probabiliste.

- **Le module d'extraction des paramètres acoustiques :** permet de convertir le signal de parole sous la forme de vecteurs acoustiques qui représentent les informations pertinents pour les processus de reconnaissance.

- **Les modèles acoustiques** : représentent les unités lexicales qui peuvent être un mot, une syllabe, un phonème, etc.

- **Le lexique :** est utilisé pour faire le lien entre le vocabulaire et les modèles acoustiques. Pour un système de reconnaissance de mots isolés, le lexique fait habituellement la correspondance entre un modèle acoustique et un mot. Pour la reconnaissance de parole continue, le modèle acoustique étant habituellement un phonème, le lexique impose alors la combinaison des phonèmes pour créer un mot [Lame-96].

- **le modèle de langage :** contient les informations qui indiquent comment connecter les mots ensemble dans un arbre des mots possibles. Le modèle de langage peut contenir les syntaxes grammaticales ou des modèles de langage stochastiques comme les N-grammes.

- **Le décodage :** L'entrainement des modèles acoustiques est effectué à partir de la base de données d'apprentissage qui est constituée d'un ensemble de répétition pour chacun des mots du vocabulaire (chaque mot du vocabulaire est modélisé par un modèle acoustique). La probabilité de génération des mots est renforcée par le modèle de langage. Enfin, le décodage est effectué par choix du modèle acoustique le plus probable.

1.4 Conclusion

Dans ce chapitre, nous avons présenté les méthodes d'analyse du signal pertinentes pour représenter le signal de parole en vu de sa reconnaissance et les techniques de reconnaissance les plus utilisées actuellement. Parmi les méthodes développées, les méthodes construites à partir d'une modélisation probabiliste basée sur les modèles de Markov cachés. Cette modélisation est appliquée à une représentation spectrale du signal sous forme de vecteurs de paramètres. Le formalisme des modèles de Markov cachés, les algorithmes d'apprentissage et de décodage seront présentés plus en détails dans le chapitre suivant étant donné que c'est cette approche que nous avons utilisé dans ce travail.

Les Modèles de Markov Cachés

2.1 Introduction

Depuis leur introduction en traitement de la parole [Jeli-76], les modèles de Markov cachés (HMM : Hidden Markov Model, en anglais) ont pris une importance considérable, au point que la quasi-totalité des systèmes de RAP actuels utilisent cette modélisation.
Les modèles de Markov cachés supposent que le phénomène modélisé est un processus aléatoire et inobservable qui se manifeste par des émissions elles-mêmes aléatoires. Ces deux niveaux donnent à l'approche markovienne une flexibilité pour modéliser un phénomène aussi complexe que la parole.
De nombreuses présentations théoriques des HMM existent dans la littérature, nous reprenons en partie les notations de L. Rabiner [Rabi-89].

2.2 Définition

Théoriquement, un modèle de Markov caché est un double processus stochastique (Q_t, O_t) $1 \le t \le T$. La chaîne interne Q_t non observable, et la chaîne externe O_t observable, s'allient pour générer le processus stochastique. La chaîne interne change d'état en suivant une loi de transition. L'observateur ne peut voir que les sorties des fonctions aléatoires associées aux états et ne peut pas observer les états de la chaîne Q_t, d'où le terme de Modèles de Markov Cachés [Jaco-95].

Le processus (Q_t) $1 \le t \le T$ est une chaîne de Markov d'ordre 1(la connaissance du passé se résume à celle du dernier état occupé), il doit vérifier:

$$P(Q_{t+1} = q_j / Q_t = q_j, \ldots, Q_0 = q_0) = P(Q_{t+1} = q_j / Q_t = q_j)$$

$$= a_{ij} \qquad \text{pour tout } t \ge 1. \tag{2.1}$$

Le processus (O_t) $1 \le t \le T$, est un processus observable qui vérifie :

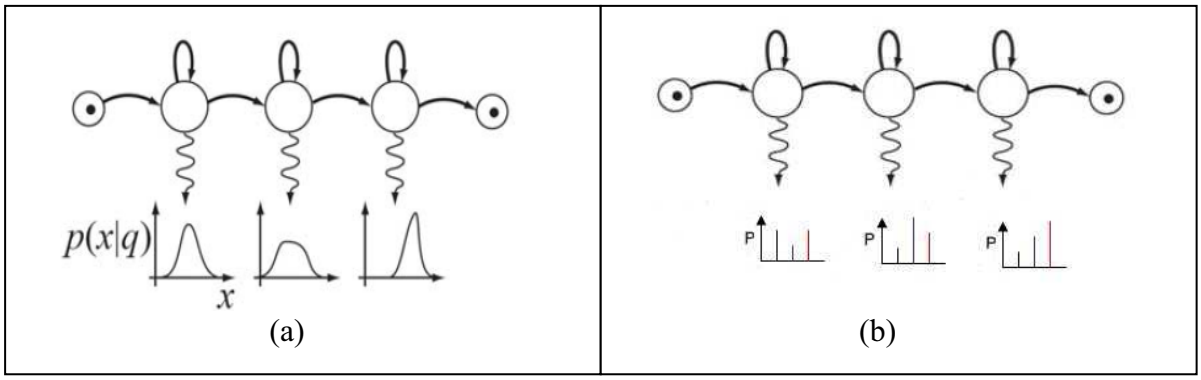
$$P(O_t = o_t / Q_t = q_i, \ldots, Q_1 = q_1, O_{t-1} = o_{t-1}, \ldots, O_1 = o_1) = P(O_t = o_t / Q_t = q_i)$$

$$= b_i(o_t) \tag{2.2}$$

La modélisation de la loi de probabilité $b_i(o_t)$ peut être de nature:

- **Discrète** : $b_i(o_t)$ est une distribution de probabilité discrète : une loi discrète est généralement représentée par les fréquences d'apparitions des observations discrètes (figure 2.1.a).

- **Continue** : $b_i(o_t)$ est une fonction de densité de probabilité définie sur R^d ($d = |o_t|$): les densités traditionnelles utilisées sont des densités gaussiennes, entièrement définies par le vecteur moyenne et la matrice de covariance, ou des densités de type multi-gaussiennes (sommes pondérées de densités gaussiennes) (figure 2.1.b).



(a) (b)

Figure 2.1: Types de modélisation de la loi de probabilité des observations :(a) modélisation de la loi continue (b) modélisation de la loi discrète.

Un Modèle de Markov Caché λ est alors défini par le triplet λ=(π, A, B) tel que :

- $\pi = \{ \pi_i \}$ le vecteur des probabilités initiales :

$$\pi_i = P(q_1 = i) \ , \quad 1 \le i \le N \tag{2.3}$$

- $A = \{a_{ij}\}$ la matrice des probabilité de transitions entre les états *i* et *j*:

$$a_{ij} = P(q_{t+1} = j \mid q_t = i). \ 1 \le i \le N, \ 1 \le j \le N \ , \quad 1 \le t \le T$$

$$\sum_{i=1}^{N} a_{ij} = 1 \qquad 1 \le j \le N \tag{2.4}$$

- $B = \{b_i (o_t)\}$ la matrice (ou la fonction de densité) de probabilité d'observation selon le type de modélisation d'observations :

$$b_i(o_t) = P(o_t / q_t = i) \tag{2.5}$$

2.3 Modélisation de la parole par un HMM

Pour simplifier les choses nous supposons qu'un modèle HMM modélise un mot du vocabulaire, dans un cas plus général, la modélisation d'un mot est construite par la concaténation de plusieurs modèles HMMs, où chaque modèle HMM modélise une unité acoustique de base telle que le phonème.

2.3.1 Principe de la modélisation

Un modèle HMM va modéliser un signal parole d'une telle façon que chaque segment supposé stationnaire ou pseudo-stationnaire de signal va correspondre à un état dans le modèle HMM. Chaque état HMM est caractérisé par une distribution de probabilité des différents vecteurs acoustiques associés au segment attribué à cet état. La transition d'un segment à un autre segment du signal est modélisée par la transition entre les états, laquelle est supposée être instantanée et caractérisée par la probabilité de transition de l'état (Figure 2.2).

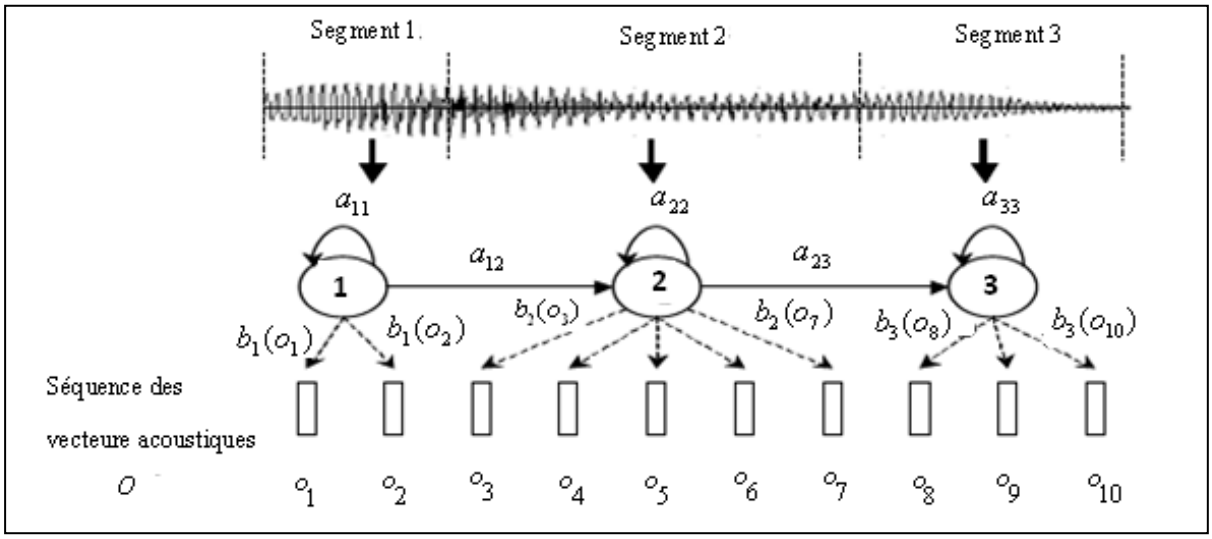


Figure 2.2: Un exemple de HMM à 3 états modélisant un signal contenant 10 vecteurs acoustiques.

2.3.2 Topologie des HMMs utilisés pour la parole

La parole est un phénomène dont la dimension temporelle ne peut être ignorée. Les HMM utilisés pour la représenter sont des modèles "gauche-droite" qui ne permettent pas de "retour en arrière" et qui démarrent toujours depuis l'état initial (*i=1*).
C'est-à-dire que leurs probabilités vérifient :

$$i > j \quad \Rightarrow a_{ij} = 0 \qquad 2 \le i \le N, \qquad 1 \le j \le N-1 \tag{2.6}$$

$$\pi_i = P(q_1 = i) = \begin{cases} 1 & pour \quad i = 1 \\ 0 & pour \quad 1 < i \le N \end{cases} \tag{2.7}$$

$$(\text{c-à-d } \pi = \{1 \quad 0 \quad .... \quad 0\})$$

Dans ce cadre, R. Bakis [Baki-76] a proposé un modèle type pour représenter un mot qui permet le bouclage sur l'état courant (progression acoustique stationnaire) et le passage à l'état suivant (progression acoustique "standard") (figure 2.3). Le nombre d'états du modèle est normalement proportionnel à la durée moyenne du mot. La pluparts des systèmes de reconnaissance utilisent des modèles à trois états.

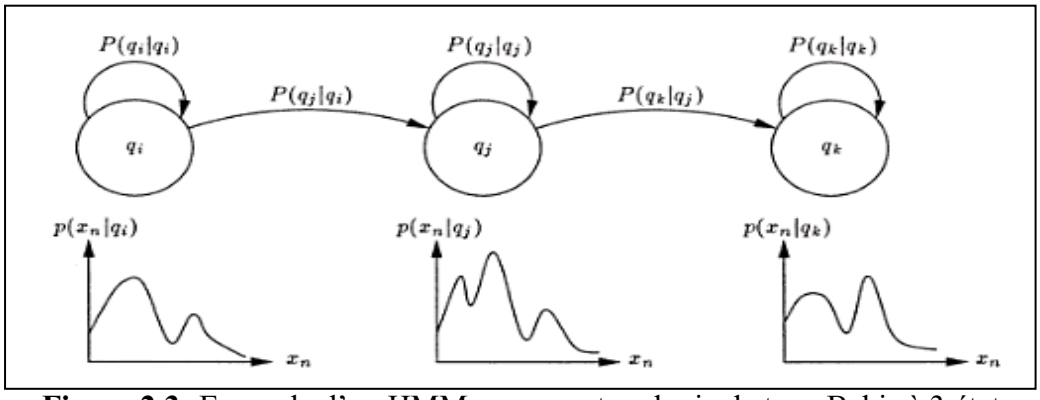


Figure 2.3: Exemple d'un HMM avec une topologie de type Bakis à 3 états.

Ce type de modèle est devenu générique dans le domaine de la RAP. Il est utilisé dans de nombreux systèmes pour modéliser les unités acoustiques de base.

2.3.3 Modélisation des observations acoustiques

Les observations émises lors des transitions représentent la succession des trames acoustiques au cours de la prononciation du mot. Ces observations peuvent être décrites par un nombre fini de symboles au moyen de la quantification vectorielle dans le cas des modèles discrets, ou de modéliser leurs probabilités d'émission par des densités de probabilité continues dans le cas des modèles continus.

2.3.3.1 Observations discrètes

L'espace de représentation du signal de parole est généralement un espace multidimensionnel continu E, et les vecteurs de coefficients calculés sur une trame de signal sont des points de cet espace. Il est possible de modéliser ces observations par des modèles de Markov à émissions discrètes au moyen de la quantification vectorielle (QV).
La QV permet le passage de l'espace E vers un espace discret, en partitionnant cet espace et en choisissant un représentant pour chaque classe. L'ensemble des représentants constitue un dictionnaire de M prototypes noté $V = \{v_k\}_{1 \le k \le M}$. Chaque vecteur $O \in E$ est quantifié par le prototype du dictionnaire dont il est le plus proche au sens d'une distance $d(O, \hat{O})$ définie dans l'espace E:

$$O \xrightarrow{QV} \hat{O} = v_{r'} \qquad avec \qquad r' = \arg\min_{1 \le r \le M} d(O, v_r) \tag{2.8}$$

Divers algorithmes ont été proposés pour la réalisation du dictionnaire des prototypes, basés le plus souvent sur une classification hiérarchique descendante ou sur les nuées dynamiques [Gray-84]. Cependant, la QV introduit des distorsions, et il est souvent préférable de travailler directement dans l'espace continu E.

2.3.3.2 Observations continues

Le principe de l'émission de symboles discrets peut se généraliser au cas continu. Les probabilités d'émission discrètes $b_j(k)$ sont alors remplacées par des densités de probabilité continues dans l'espace de représentation. Cette solution évite les distorsions introduites par la QV, mais pose le problème du choix des densités de probabilité et de la robustesse de leur estimation. L'utilisation d'une combinaison linéaire de gaussiennes dans l'espace R^d est fréquente:

$$b_j(O) = \sum_{k=1}^{M} c_{jk} N(O, \mu_{jk}, \sum_{jk}) = \sum_{k=1}^{M} c_{jk} b_{jk}(O) \qquad 1 \le j \le N \tag{2.9}$$

Avec M est le nombre de gaussiennes à l'état j. $N(O, \mu_{jk}, \sum_{jk})$ où $b_{jk}(O)$ est la distribution gaussienne du $k^{\text{ème}}$ mélange de l'état j. Cette distribution est définie par le vecteur moyen μ_{jk} et la matrice de covariance $\sum_{jk}$ à l'état j. c_{jk} est le coefficient de pondération du $k^{\text{ème}}$ mélange qui satisfait la contrainte stochastique :

$$\sum_{k=1}^{M} c_{jk} = 1 \tag{2.10}$$

$$c_{jk} \geq 0, \qquad 1 \leq j \leq N, \quad 1 \leq k \leq M$$

Nous rappelons que la densité de probabilité d'une loi normale de moyenne μ et de matrice de covariance Σ en dimension d est:

$$N(O, \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp^{-\frac{1}{2}(O-\mu)' \Sigma^{-1} (O-\mu)} \qquad (2.11)$$

L'hypothèse qui est souvent faite d'une indépendance entre les d dimensions de l'espace autorise l'utilisation de matrices de covariance diagonales, cela limite le nombre de paramètres à estimer et simplifie les calculs. D'autres types de densités de probabilité sont possibles, comme la sortie d'un réseau de neurones dans le cas de systèmes de reconnaissance hybride [Bour-90].

2.4 Un système de RAP à base d'HMMs

A partir d'une suite d'observations O supposées émises par un modèle, différents problèmes peuvent être posés:

• L'évaluation de la probabilité que la suite des observations ait été émise par un modèle HMM. Lorsque plusieurs modèles existent, cette évaluation permet le choix du modèle le plus probable.

• La recherche de la séquence cachée d'états la plus probable d'un modèle ayant produit les observations.

• L'apprentissage des paramètres d'un modèle. A partir d'un modèle initial et d'observations supposées émises par ce modèle, on cherche les probabilités de transition et d'émission maximisant la vraisemblance des observations.

Nous allons commencer par présenter la résolution de ces problèmes dans le cadre simplifié de la reconnaissance de mots isolés dans laquelle chaque modèle HMM représente un mot de vocabulaire. La reconnaissance de parole continue n'est qu'une généralisation de ces résolutions et elle va être abordée dans le paragraphe 2.6.

2.4.1 Reconnaissance d'une séquence d'observations

Supposons que les observations $O = (o_1 \ldots o_T)$ sont les trames acoustiques d'un mot inconnu. On cherche à trouver le mot qui a été prononcé parmi l'ensemble des R mots du vocabulaire, ceci revient à désigner parmi les modèles proposés lequel coïncide le mieux avec la séquence d'observations, soit d'après la règle de décision bayésienne (1.37):

$$M_{best} = \arg\max_{M} \quad P(O/M)P(M) \qquad 1 \leq best \leq R \qquad (2.12)$$

Chaque mot M est modélisé par une modèle HMM λ_M, d'où l'hypothèse suivante:

$$P(O/M) = P(O/\lambda_M) \qquad (2.13)$$

L'équation (2.12) peut alors être reformulée:

$$M_{best} = \arg\max_{M} \quad P(O/\lambda_M)P(M) \qquad 1 \le best \le R \tag{2.14}$$

Sa résolution nécessite l'estimation de la probabilité d'émission des observations $P(O/\lambda_M)$ pour chaque modèle HMM modélisant un mot du vocabulaire.

2.4.1.1 Probabilité d'émission des observations

La probabilité que la suite d'observations $O = (o_1 \ldots o_T)$ soit émise par le modèle λ peut être calculée comme la somme des probabilités conjointes de l'observation O et du chemin Q pour l'ensemble de tous les chemins possibles de longueur T:

$$P(O/\lambda) = \sum_{Q} P(O,Q/\lambda) \tag{2.15}$$

D'après les définitions du modèle, la probabilité de partir à l'instant t=1 de l'état d'indice $q1$=1 et de suivre le chemin Q= $(q_1 \ldots q_T)$ est le produit des probabilités de transition sur le chemin:

$$P(Q/\lambda) = \pi_{q1} a_{q1q2} a_{q2q3} \ldots . a_{qT-1qT} \tag{2.16}$$

Et la probabilité d'avoir émis les observations O en suivant ce chemin (supposant que les observations sont indépendantes) est:

$$P(O/\lambda) = \prod_{t=1}^{T} P(o_t / q_t, \lambda) = b_{q1}(o_1) b_{q2}(o2) \ldots . b_T(o_T) \tag{2.17}$$

D'où la probabilité conjointe du chemin et des observations:

$$P(O,Q/\lambda) = P(O/Q,\lambda)P(Q/\lambda) \tag{2.18}$$

La probabilité d'émission des observations est finalement:

$$P(O/\lambda) = \sum_{Q} P(O,Q/\lambda) = \sum_{Q} P(O/Q,\lambda).P(Q/\lambda) = \sum_{Q} \prod_{t=}^{T} a_{qt-1qt} b_{qt}(o_t) \tag{2.19}$$

Avec un modèle HMM à N états, le nombre de chemins possibles est de l'ordre de N^T, et l'ensemble des chemins devient très rapidement impossible à décrire. Il existe heureusement un algorithme rapide, dit "Forward-Pass" (estimation directe), qui permet de calculer récursivement cette quantité [Baum-72].

2.4.1.2 Estimation directe

Une variable intermédiaire est introduite pour le calcul de la probabilité d'émission. La variable directe $\alpha_t(i)$ est définie comme la probabilité que les observations jusqu'à l'instant t aient été émises par le modèle λ, et que l'état à cet instant soit l'état d'indice i:

$$\alpha_t(i) = P(o_1 \ldots . o_t, q_t = i/\lambda) \tag{2.20}$$

On résout alors $\alpha_t(i)$ par récurrence:

- **Initialisation**:

$$\alpha_1(i) = \pi_i b_i(O_1) \qquad 1 \le i \le N \tag{2.21}$$

- **Récurrence** : (figure 2.4)

Pour t allant de 1 à T,

Pour j allant de 1 à N

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1}) \tag{2.22}$$

Fin

Fin

- **Résultat final:**

$$P(O/\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{2.23}$$

Puisque par définition : $\alpha_T(i) = P(o_1 o_2 ... o_T, q_T = i \setminus \lambda)$ et donc P (O\λ) ce n'est que la somme des $\alpha_T(i)$.

Le nombre d'opération de calcul effectuer par cette procédure est d'ordre N^2T opérations, il est claire que c'est mieux de $2TN^T$ opérations effectuer dans le calcul directe.

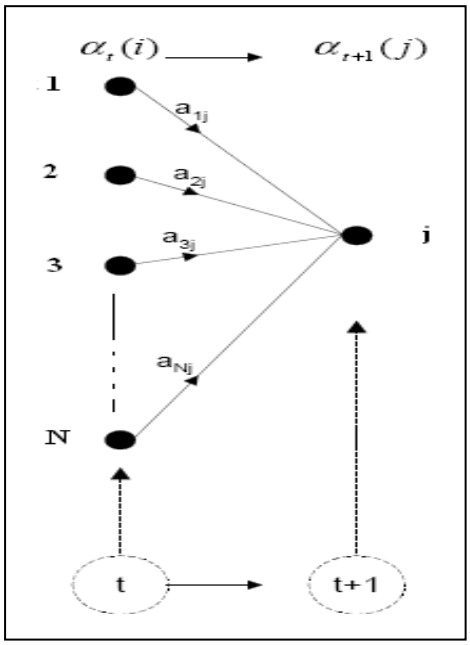


Figure 2.4: Progression de la procédure estimation directe.

2.4.1.3 Estimation Backward

L'estimation directe est suffisante pour obtenir la probabilité d'émission recherchée. Cependant, une estimation rétrograde dans le temps est aussi possible, avec la probabilité $\beta_t(i)$ que les observations après l'instant *t* soient émises en partant de l'état d'indice *i*:

$$\beta_t(i) = P(o_{t+1}....o_T / q_t = i, \quad \lambda) \tag{2.24}$$

- Initialisation:

$$\beta_T(i) = 1 \qquad 1 \le i \le N \tag{2.25}$$

- **Récurrence** : (figure 2.5)

Pour *t* allant de T à *1*,

  Pour *i* allant de 1 à *N*

$$\beta_T(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \tag{2.26}$$

  Fin

Fin

Résultat final :

$$P(O/\lambda) = \beta_1(i) \tag{2.27}$$

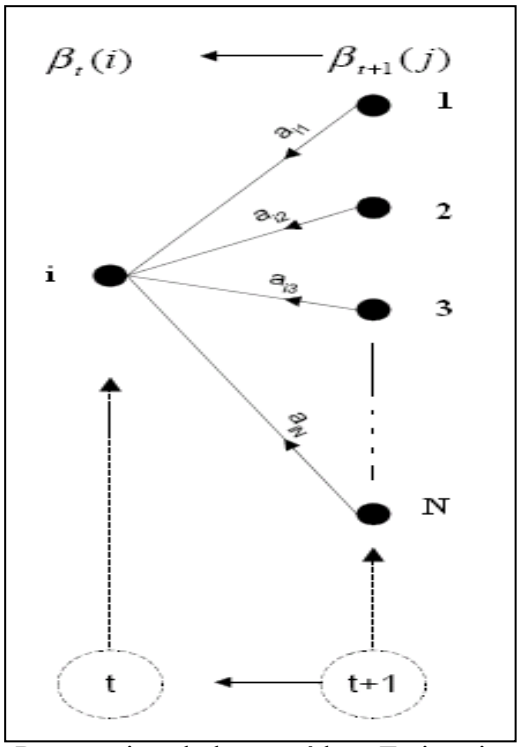


Figure 2.5: Progression de la procédure Estimation rétrograde.

2.4.2 Recherche des états cachés

La procédure d'estimation directe ou rétrograde fournit la probabilité d'émission des observations sur toutes les séquences d'états possibles, sans choisir un chemin particulier. Il est parfois utile de connaître la séquence d'états qui a émis les observations. L'algorithme de Viterbi [Vite-67] [Forn-73] cherche la séquence d'états cachés la plus probable et calcule la probabilité d'émission le long de ce chemin. La procédure peut être exploitée pour l'initialisation des modèles ainsi que pour le décodage de la parole continue.

On recherche la meilleure séquence d'états, $Q = \{q_1, q_2,\ldots,q_T\}$, connaissant la séquence d'observations $O = (o_1 \ldots o_T)$. On définit alors $\delta_t(i)$, comme la probabilité maximale que les observations observées jusqu'à l'instant t aient été émises par le modèle λ en suivant un chemin qui arrive à l'état d'indice i :

$$\delta_t(i) = \max_{q_1,q_2,\ldots,q_{t-1}} P(o_1, o_2,\ldots, o_t, q_1, q_2,\ldots, q_t = i \setminus \lambda) \tag{2.28}$$

Alors une récurrence similaire à celle suivie pour le calcul de la probabilité d'émission s'applique, à laquelle s'ajoute la mémorisation du meilleur chemin:

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] b_j(o_{t+1}) \tag{2.29}$$

Pour conserver la trace de l'indice de l'état qui maximise un $\delta_t(j)$ donné, on emploie le vecteur $\psi_t(j)$. On énonce alors la procédure complète comme suit :

-**Initialisation :**

$$\begin{cases} \delta_1(i) = \pi_i b_i(o_1) & 1 \le i \le N \\ \psi_1(i) = 0 \end{cases} \tag{2.30}$$

Récurrence :

Pour t allant de 2 à T,

Pour j allant de 1 à N,

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(o_t) \tag{2.31}$$

on conserve la mémoire du meilleur état précédent par:

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] \tag{2.32}$$

Fin

Fin

Résultat final :

$$P(O,Q \setminus \lambda) = \max_{1 \le i \le N}[\delta_T(i)] \tag{2.33}$$

construction du meilleur chemin $\tilde{Q} = (\tilde{q}_1, \tilde{q}_2, ..., \tilde{q}_T)$ avec:

$$\tilde{q}_T = \arg\max_{1 \le i \le N}[\delta_T(i)] \tag{2.34}$$

Puis, pour *t* allant de *T-1* à 1:

$$\tilde{q}_t = \psi_{t+1}(\tilde{q}_{t+1}) \tag{2.35}$$

2.4.3 Apprentissage d'un modèle

La reconnaissance d'un mot prononcé est rendue possible par l'évaluation de la probabilité d'émission des observations par tous les modèles de mots. Cela suppose l'existence d'un modèle pour chaque mot de vocabulaire, et l'apprentissage des paramètres de ces modèles. Ces paramètres sont les probabilités de transition entre états et les probabilités d'émission associées aux états.
Ainsi, connaissant une suite d'observations *O* émises par un modèle, il est possible de modifier les paramètres du modèle $\tilde{\lambda}$ de manière à rendre maximale la probabilité d'émission des observations *O* par le modèle. Il s'agit d'une estimation par le critère du maximum de vraisemblance (Maximum Likelihood Estimation ou MLE) qui est réalisée par l'algorithme de Baum-Welch [Baum-72] [Rabi-89].
Les formules de l'algorithme de Baum-Welch permettent une ré-estimation itérative des paramètres a_{ij} et $b_j(k)$ du modèle $\tilde{\lambda}_n$, le nouveau modèle $\tilde{\lambda}_{n+1}$ vérifie:

$$P(O \setminus \tilde{\lambda}_{n+1}) \ge P(O \setminus \tilde{\lambda}_n) \tag{2.37}$$

La convergence vers un optimum local est démontrée, mais les valeurs initiales des paramètres *A* et *B* sont cruciales pour assurer une convergence correcte et rapide le plus près possible du maximum global. L'algorithme de Viterbi réalisant le décodage peut servir à l'initialisation des modèles.

Pour décrire les procédures de réestimation, on va définir deux variables :

On appelle $\gamma_t(i)$ la probabilité d'être dans l'état *i* au temps *t*, sachant le modèle λ et la séquence d'observation *O :*

$$\gamma_t(i) = P(q_t = i \setminus O, \lambda) \tag{2.38}$$

et $\xi_t(i, j)$ la probabilité d'être dans l'état i au temps *t* et dans l'état *j* au temps *t*+1, sachant le modèle λ et la séquence d'observation *O :*

$$\xi_t(i, j) = P(q_t = i, q_{t+1} = j \setminus O, \lambda) \tag{2.39}$$

On peut développer ces probabilités en utilisant les variables des algorithmes forward et

Backward (figure 2.6) :

$$\xi_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O \setminus \lambda)} \tag{2.40}$$

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(j)}{P(O \setminus \lambda)} = \frac{\sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O \setminus \lambda)} = \sum_{j=1}^{N} \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O \setminus \lambda)} = \sum_{j=1}^{N} \xi_t(i,j) \tag{2.41}$$

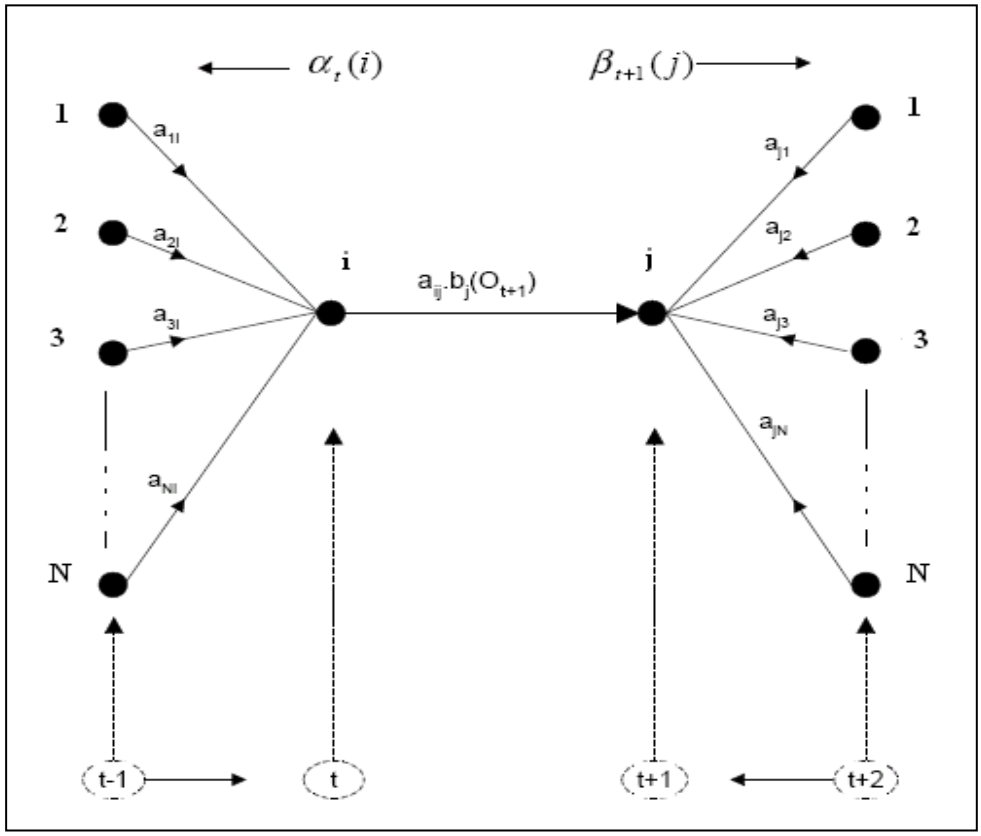


Figure 2.6 : Progression de la procédure de Baum-Welch

La somme de $\gamma_t(i)$ et $\xi_t(i,j)$ au cours de temps *t*, peut être interprétée comme :

$$\sum_{t=1}^{T-1} \gamma_t(i) = \sum_{t=1}^{T-1} P(q_t = i \setminus O, \lambda) = \text{nombre de transitions depuis l'état i.} \tag{2.42}$$

$$\sum_{t=1}^{T-1} \xi_t(i,j) = \sum_{t=1}^{T-1} P(q_t = i, q_{t+1} = j \setminus O, \lambda) = \text{nombre de transitions depuis l'état i vers l'état j.} \tag{2.43}$$

En utilisant ces sommes, les nouveaux paramètres réestimés $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$ pour un modèle $\lambda = (A, B, \pi)$ sont :

$\bar{\pi}_i$ =nombre de passage par l'état i au temps (t=1) = $\gamma_1(i)$ (2.44)

$$\bar{a}_{ij} = \frac{\text{nombre de transitions de l'état i vers l'état j}}{\text{nombre de transitions depuis l'état i}} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)} \quad (2.45)$$

a) Dans le cas d'une modélisation discrète des observations :

$$\bar{b}_j(k) = \frac{\text{nombre d'observations du symbole vk dans l'état j}}{\text{nombre de passages par l'état j}} = \frac{\sum_{\substack{t=1 \\ \text{avec } o_t = v_k}}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)} \quad (2.46)$$

b) Dans le cas d'une modélisation continue des observations :

Pour une loi de densité multi gaussiennes définit (la section 2. 3.3.2) par:

$$b_j(O) = \sum_{k=1}^{M} c_{jk} N(O, \mu_{jk}, \sum_{jk}) = \sum_{k=1}^{M} c_{jk} b_{jk}(O) \qquad 1 \le j \le N \quad (2.47)$$

Le coefficient de pondération c_{jk}, le vecteur de moyenne μ_{jk} et la matrice de covariance Σ_{jk} de la densité de probabilité associée au $k^{\text{ème}}$ mélange de l'état j sont recalculés comme suit:

$$\bar{c}_{jk} = \frac{\sum_{t=1}^{T}\gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{k=1}^{M}\gamma_t(j,k)} \quad (2.48)$$

$$\bar{\mu}_{jk} = \frac{\sum_{t=1}^{T}\gamma_t(j,k).o_t}{\sum_{t=1}^{T}\gamma_t(j,k)} \quad (2.49)$$

$$\bar{\Sigma}_{jk} = \frac{\sum_{t=1}^{T}\gamma_t(j,k).(o_t - \bar{\mu}_{jk})(o_t - \bar{\mu}_{jk})'}{\sum_{t=1}^{T}\gamma_t(j,k)} \quad (2.50)$$

Où $\gamma_t(j,k)$ est une généralisation de terme $\gamma_t(j)$, il est définit comme la probabilité d'être à l'état j au temps t pour le $k^{\text{ème}}$ mélange de gaussienne :

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jk}N(o_t,\mu_{jk},\Sigma_{jk})}{\sum_{k=1}^{M}c_{jk}N(o_t,\mu_{jk},\Sigma_{jk})}\right] \quad (2.51)$$

Il a été démontré par Baum et ses collègues [Baum-68] que $P(O|\bar{\lambda}) \geq P(O|\lambda)$.

2.5 Mise en œuvre des HMMs

Après avoir présenté les principes théoriques des HMMs, nous décrivons maintenant deux aspects importants pour leur mise en œuvre :

2.5.1 Changement d'échelle

L'une des particularités des probabilités acoustiques est leur faible valeur numérique. Ce qui implique que la valeur de la probabilité calculée lors de la phase de décodage est très faible au point de sortir des possibilités de codage numérique habituelles des ordinateurs.

La méthode employée pour résoudre ce problème est de transposer l'ensemble des calculs dans le domaine logarithmique. Les propriétés du domaine logarithmique permettent de transformer la multiplication de probabilités en addition, ce qui a pour effet une diminution des temps de calcul :
L'équation (2.17) peut alors s'écrire :

$$P(O/\lambda) = \prod_{t=1}^{T} P(o_t/q_t,\lambda) = \prod_{t=1}^{T} b_{qt}(o_t) \Rightarrow \qquad \log P(O/\lambda) = \sum_{t=1}^{T} \log b_{qt}(o_t) \tag{2.52}$$

Ceci évite également d'avoir à calculer l'exponentielle dans la fonction de densité gaussienne :

$$P(O/\lambda) = \frac{1}{(2\pi)^{d/2}\left|\sum\right|1/2} \exp^{-\frac{1}{2}(O-\mu)^t \Sigma^{-1}(O-\mu)} \tag{2.53}$$

$$\Rightarrow \log P(O/\lambda) = -\frac{d}{2}\log(2\pi) - \frac{1}{2}\log(\det(\Sigma)) - \frac{1}{2}(O-\mu)^T \Sigma^{-1}(O-\mu) \tag{2.54}$$

De plus comme le log est une fonction croissante monotoniquement, l'inégalité des vraisemblances (probabilités) reste valable pour le log-vraisemblance :

$$P(O/\lambda_1) > P(O/\lambda_2) \Rightarrow LogP(O/\lambda_1) > LogP(O/\lambda_2) \tag{2.55}$$

2.5.2 Initialisation des modèles

La convergence des modèles vers un maximum le plus proche possible du maximum global au cours de leur ré-estimation par les équations de Baum-Welch nécessite une bonne initialisation des probabilités d'émission.

Dans la littérature [Rabi-89], les expériences montrent que le choix aléatoire de l'initialisation produit des résultats satisfaisants pour les probabilités initiales et pour la matrice des probabilités de transition. Ceci n'est pas vrai pour la matrice des probabilités d'observation. De plus, les expériences montrent qu'une bonne initialisation est très utile dans le cas des HMM discrets, et qu'elle est essentielle dans le cas continu. Pour palier à ce problème d'initialisation, l'algorithme de Viterbi est utilisé pour trouver la meilleure suite d'états dans le modèle et associer chaque trame émise à un état particulier.

2.6 Reconnaissance de la parole continue

Pour les systèmes de reconnaissance de la parole continue à grand vocabulaire, il n'est pas réaliste d'apprendre un modèle pour chaque mot, et l'utilisation d'unités acoustiques sub-lexicales est nécessaire. De plus, la démarche proposée pour l'identification de mots isolés qui consiste à tester chacun des mots possibles n'est plus adaptée à la reconnaissance de la parole continue, car le nombre de phrases possibles par enchaînement de mots, et donc le nombre de modèles à évaluer, est virtuellement infini. La reconnaissance est donc réalisée avec une variante de l'algorithme de Viterbi déjà présenté pour le décodage, en respectant la syntaxe définie par un réseau de mots. La procédure d'apprentissage est elle aussi modifiée pour permettre l'apprentissage des modèles sur des phrases non segmentées.

2.6.1 Modèle acoustique

La modélisation par mots est très efficace pour des applications de reconnaissance de mots isolés ou en vocabulaire limité [Rabi-89]. Pour la reconnaissance de la parole continue à grand vocabulaire, il est impossible d'apprendre un modèle pour chacun des mots du vocabulaire. La modélisation d'unités acoustiques de taille plus courte que le mot devient nécessaire. Ces unités (dites sub-lexicales) peuvent être des phonèmes, des diphones (deux phonèmes successifs), des triphones (trois phonèmes successifs) ou encore des syllabes.

Dans la pratique, le phonème semble un choix naturel pour l'unité de base, car il a l'avantage de ne nécessiter qu'un nombre très réduit de modèles, au plus quelques dizaines, mais la variabilité de la coarticulation n'est pas prise en compte, les systèmes de reconnaissance les plus performants utilisent d'autres types de modélisation telles que les diphones ou les triphones, mais ceci augmente le nombre de modèles, de l'ordre de plusieurs milliers.

Les modèles de mots alors sont construits par concaténation de modèles sub-lexicaux. Nous nous limitons par la suite à la description de modèles de phonèmes.

Pour la modélisation d'une unité acoustique courte comme un phonème un seul état suffirait à modéliser un son stationnaire, mais en raison des phénomènes de coarticulation, le début et la fin d'une réalisation phonétique peuvent présenter des caractéristiques acoustiques différentes du centre supposé plus stationnaire. Le choix le plus adéquat est un modèle gauche-droit à trois états émetteurs [Barr-96] [Igou-98] [Youn-05].

2.6.2 Modèle de langage

Une reconnaissance acoustique même parfaite ne suffit pas pour obtenir une transcription correcte de la phrase: une suite particulière de 9 phonèmes peut être transcrite en français en 32000 suites de mots différentes orthographiquement correctes, quelques unes seulement sont des phrases syntaxiquement correctes [Barr-96]. Il est donc indispensable d'introduire dans les systèmes de RAP des contraintes du langage.

Le modèle de langage a pour objectif de capturer les contraintes du langage naturel afin de guider le décodage acoustique. Ces contraintes peuvent prendre différentes formes : grammaire à syntaxe fixe, modèle probabiliste, grammaire probabiliste. Nous nous intéressons dans notre travail aux modèles de langage probabilistes qui sont aujourd'hui les plus utilisés dans les systèmes de RAP.

2.6.2.1 Modèle de langage probabiliste

Les modèles de langage probabilistes ont pour objet d'attribuer une probabilité à une séquence de mots. De manière générale, la probabilité de la séquence de mots *W* de taille *k* est exprimée comme le produit des probabilités conditionnelles d'un mot sachant tous les mots précédents :

$$P(W) = P(w_1)\prod_{i=2}^{k} P(w_i / w_{i-1},....,w_{i+k+1}) = P(w_1)\prod_{i=2}^{k} P(w_i / h_i) \qquad (2.56)$$

Où h_i est l'historique de longueur *k* du mot w_i : $h_i = w_{i-1},....,w_{i+k+1}$
Le modèle de langage est estimé sur de grands corpus de textes pour avoir un maximum de couverture lexicale. Des données telles que des textes de journaux, de dépêches électroniques ou de transcriptions de documents audio sont utilisées.

Le modèle de type n-grammes est le modèle probabiliste le plus généralement utilisé. Pour ce genre de modèle, l'historique d'un mot est représenté par les n − 1 mots qui le précèdent.
Dans la pratique, la valeur de *n* ne dépasse pas 3 : on parle alors de modèle tri-grammes. (uni-gramme pour n = 1, bi-gramme pour n = 2).

Soit $f(w_{i-n+1},....,w_i)$ la fréquence de la suite de mots $w_{i-n+1},....,w_i$ dans un corpus d'apprentissage, un modèle n-gramme estime la probabilité d'un mot w_i conditionné par son historique $h_i = w_{i-n+1},....,w_{i-1}$ comme :

$$P(w_i / h_i) = \frac{f(h_i, w_i)}{f(h_i)} \qquad si \qquad f(h_i) > 0 \qquad (2.57)$$

2.6.3 Apprentissage en parole continue

La ré-estimation d'un modèle par l'algorithme de Baum-Welch optimise (maximise) la probabilité d'émission d'un ensemble de séquences acoustiques par un modèle. Lorsque les modèles représentent des unités sub-lexicales (phonèmes), il n'est pas possible de prononcer ces unités de manière isolée, et il faut les isoler par une segmentation des phrases prononcées. Or, la segmentation manuelle est un travail fastidieux qui doit être réalisée par un expert, on ne peut espérer obtenir de cette manière les très nombreuses prononciations d'une unité nécessaire à l'estimation robuste du modèle. Une ré-estimation des modèles avec des phrases non segmentées est heureusement possible.

2.6.3.1 Modèles connectés

Connaissant la transcription phonétique d'une phrase d'apprentissage, un modèle composite est construit en mettant les modèles correspondants en séquences. Par exemple, si la phrase est transcrite par la suite phonétique $\Phi = (\varphi_1 \ldots \varphi_T)$, alors un modèle de la phrase $\Lambda = (\lambda_1 \ldots \lambda_T)$ est construit, où λ_t modélise le phonème φ_t. Ce modèle composite est construit en reliant le dernier état du modèle λ_t à l'état initial du modèle λ_{t+1}. Le modèle Λ peut être ré-estimé sur la phrase complète avec la procédure standard de Baum-Welch, et il n'est donc pas nécessaire de disposer d'une segmentation de la phrase.

Tous les modèles sont ré-estimés en utilisant une simple transcription lexicale de la phrase, et de constituer le modèle de la phrase par concaténation de modèles de mots, eux-mêmes décrits par des modèles phonétiques.

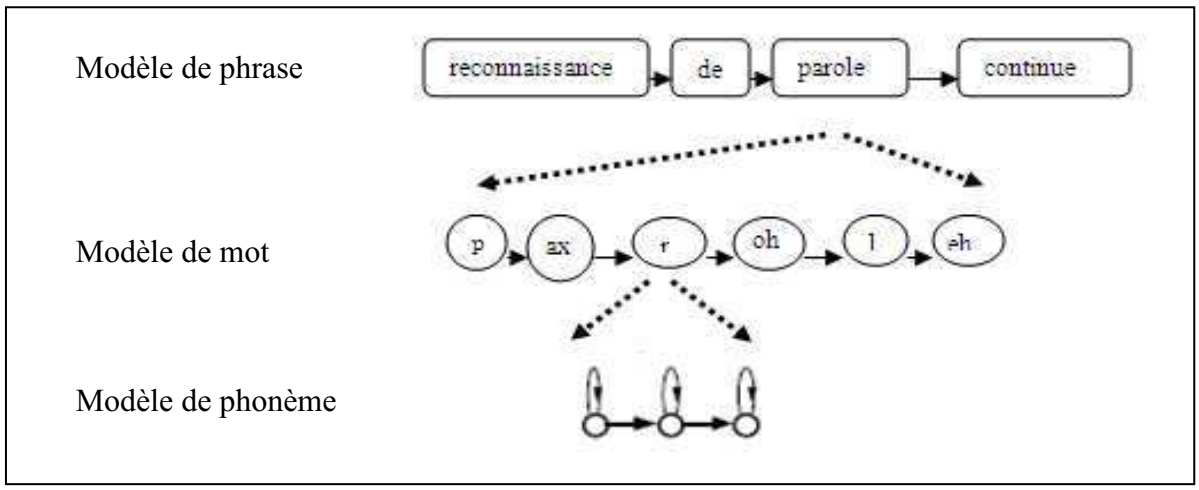


Figure 2.7: Modélisation d'une phrase à partir de modèles phonétiques.

2.6.4 Décodage de la parole continue

En reconnaissance de la parole continue, la suite de mots recherchée est celle qui maximise l'équation 1. 37 rappelée ici:

$$M_{best} = \arg\max_{M} \quad P(O/M)P(M) \tag{2.58}$$

Pour résoudre cette équation, il n'est pas possible de construire un modèle pour chacune des phrases pouvant être prononcées puis de comparer tous ces modèles avec la phrase à identifier. L'alternative consiste à construire un modèle unique (un réseau de modèles) pouvant émettre toutes les phrases syntaxiquement correctes du langage. Le décodage de la phrase prononcée est déduit du meilleur chemin dans ce modèle obtenu par une variante de l'algorithme de Viterbi qui est l'algorithme du passage de jeton [Youn-97].

2.6.4.1 Réseau de modèles

La reconnaissance de la parole continue utilise un seul réseau des modèles correspondant aux mots du vocabulaire. Dans le cas le plus simple, le modèle composite (réseau de modèles) est constitué des modèles de mots mis en parallèle, avec un bouclage de l'état final des modèles vers l'état initial de n'importe quel autre modèle pour permettre l'enchaînement de plusieurs mots (Figure 2.8), les modèles de mots peuvent eux-mêmes être des modèles construit à partir de modèles sub-lexicaux. En pratique le réseau de modèles contient des états qui n'émettent pas d'observations, mais servent uniquement à simplifier la représentation des transitions d'un modèle à un autre (par exemple l'état initial et l'état final du réseau de la figure 2.8).

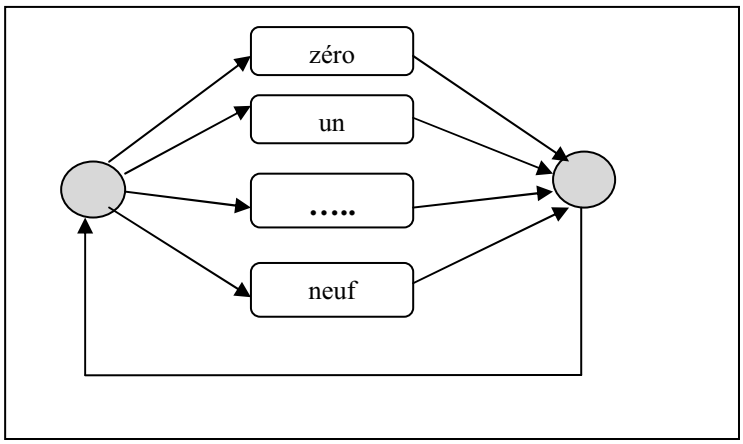


Figure 2.8: Exemple de réseau de modèles autorisant l'émission d'une suite quelconque de chiffres.

La suite d'états optimale trouvée par l'algorithme de Viterbi dans ce réseau fournit un décodage de la phrase en mots (ou en phonèmes). Cependant, l'algorithme recherche la meilleure suite d'états dans le réseau et non pas la meilleure suite de modèles.

2.6.4.2 Algorithme du passage de jeton

Des variantes de l'algorithme de Viterbi ont été proposées pour le décodage de la parole continue. L'algorithme du "Token Passing" ou "passage de jeton" est le plus utilisé [Youn-97]. Il s'applique sur un réseau de modèles sub-lexicaux.
Le jeton est un objet placé dans un état qui contient la probabilité du meilleur chemin arrivant dans cet état à l'instant courant, de plus, il conserve la mémoire de ses déplacements:

Initialisation :
Les jetons des états initiaux sont initialisés à 0
Algorithme :
Pour chaque instant *t = 1 à T* faire

Pour chaque état *i* faire
- Passer une copie du jeton dans l'état *i* vars tous les états connectés *j*.
- incrémenter sa valeur par le coût de la transition et de l'émission :$\log(a_{ij})+\log(b_j(o_t))$

Fin

- Effacer le jeton original

Pour chaque état *i* faire
Conservé le meilleur jeton arrivant à cette état
Fin

Fin
Résultat
Le meilleur jeton parmi tous les états finaux est récupéré.

Le décodage de la phrase en mots ou en phonèmes se déduit de la suite des modèles ayant émis les observations sur le chemin optimal.
L'intégration des contraintes linguistiques est possible dans cet algorithme en associant les probabilités d'un modèle de langage aux transitions entre modèles sub-lexicaux (ces transitions ont été conçues initialement équiprobable).

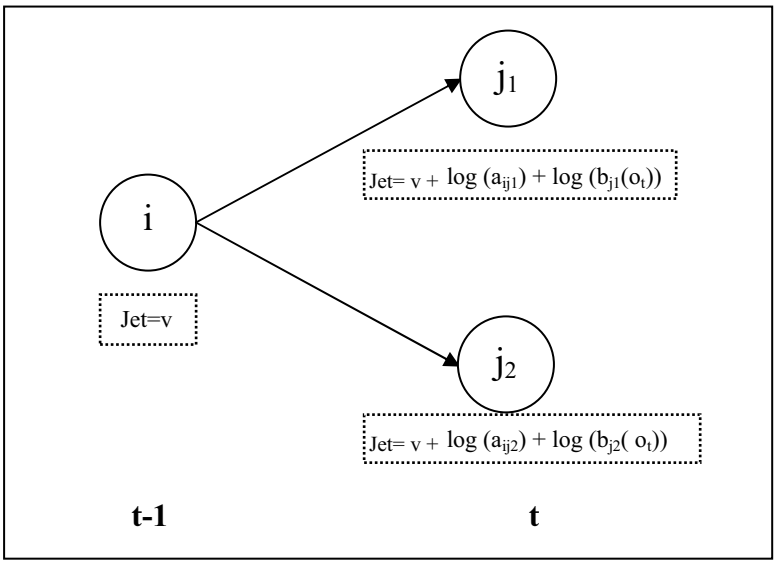


Figure 2.10: Algorithme de base du modèle de propagation de jeton.

2.7 Paramètres d'évaluation

Les performances des systèmes de RAP sont évaluées en comparant le résultat de la reconnaissance obtenue sur un nombre de phrases de test avec l'étiquetage de référence de ces phrases. La précision de cette évaluation dépend du nombre de tests réalisés.

Habituellement, les taux de reconnaissance sont représentés par le pourcentage d'identification (percent correct en anglais) et le pourcentage de reconnaissance (percent accuracy en anglais).

Le pourcentage d'identification (Ident) correspond à l'équation suivante :

$$\text{Ident} = \frac{N - O - S}{N} \times 100 \tag{2.59}$$

Le pourcentage de reconnaissance (Reco) correspond à l'équation suivante : pourcentage de mots ou de phrase reconnus correctement.

$$\text{Reco} = \frac{N - O - S - I}{N} \times 100 \tag{2.60}$$

Avec :
N : le nombre total d'unités.
O : le nombre d'omissions (le nombre d'unités non détectés).
S : le nombre de substitutions (le nombre d'unités pour lesquels le système a commis une erreur).
I : le nombre d'insertions (le nombre d'unités reconnus alors qu'aucune unité n'a été prononcé).

Comme le montre l'équation 2.59, le pourcentage d'identification ne prend pas en compte le nombre d'insertions. C'est le pourcentage de reconnaissance qui le prend en compte et pour

cela il est considéré comme le paramètre le plus indicatif pour évaluer les performances d'un système de RAP.

Ces deux équations d'évaluation peuvent être résumées dans une simple équation (2.61) pour le mode de reconnaissance de mots isolés, étant donné que le nombre d'omissions (O) et le nombre d'insertion (I) sont nuls.

$$\text{Ident} = \text{Reco} = \frac{N - S}{N} \times 100 \tag{2.61}$$

2.8 Conclusion

Dans ce chapitre nous avons présenté le concept des modèles de Markov cachés (HMMs), leur formalisme ainsi que leur emploi dans le domaine de la reconnaissance de la parole. L'utilisation des HMMs pour la reconnaissance de la parole pose en général trois problèmes à résoudre : l'initialisation, l'apprentissage et le décodage. C'est ainsi que nous avons étudié les différents algorithmes utilisés pour pallier à ces problèmes pour les deux modes de reconnaissance : mots isolés et parole continue. Dans le chapitre suivant, nous introduirons la notion des paramètres auxiliaires et nous étudierons les stratégies d'intégration de ces paramètres dans un système de RAP à base des HMMs.

Extraction et fusion des paramètres

3.1 Introduction

Les systèmes de reconnaissance de la parole actuels utilisent en entrées des paramètres dérivés de l'enveloppe spectrale de signal dits paramètres standards tels que les MFCCs, les LPCCs, les PLPs (section 1.3).
Ces paramètres standards sont très sensibles à la variabilité du signal de parole qui peut être causée par l'environnement ou la personne [Baud-93] [Mary-08] [Ezza-02]. Ceci engendre généralement différents types de disparités entre observations et modèles acoustiques [Doss-05]. Dans ce chapitre nous allons présenter les paramètres auxiliaires utilisés dans notre travail, leurs extractions ainsi que les stratégies d'intégration de ces paramètres en tant que sources d'informations complémentaires dans les systèmes de RAP. Ceci, afin de rendre ces systèmes plus robustes aux conditions réelles d'utilisation.

3.2 Signal de parole et variabilités

3.2.1 Vue générale de l'appareil phonatoire

La figure 3.1 nous montre l'appareil phonatoire humain et les éléments qui le définissent. La commande de ces différents éléments physiologiques s'effectue à partir du cerveau lui-même.

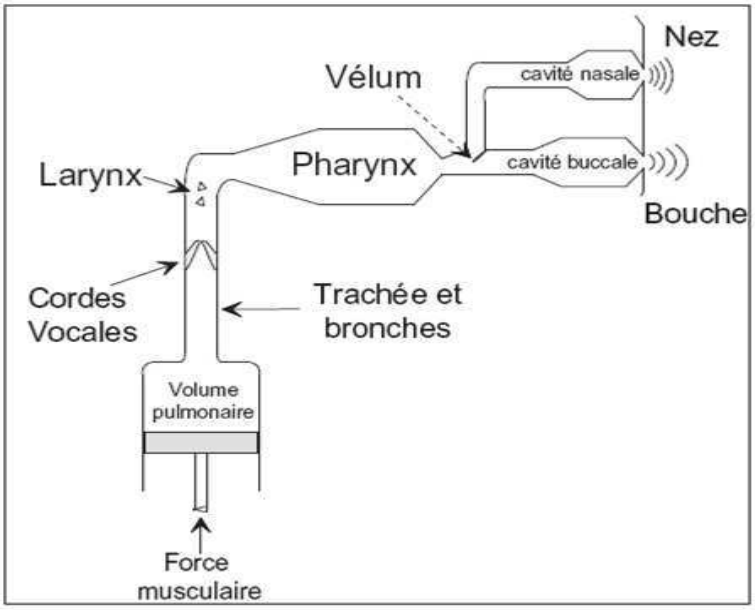


Figure 3.1: Les différentes parties constituant le conduit vocal [Rab-93].

L'appareil respiratoire fournit l'énergie nécessaire à la production des sons, en poussant l'air à travers le **larynx** où se trouvent les cordes vocales. Les **cordes vocales** sont en fait deux **muscles** symétriques placés en travers du larynx. Ces muscles peuvent fermer complètement le larynx et, en s'écartant progressivement, déterminer une ouverture triangulaire appelée **glotte**. L'air y passe librement pendant la respiration et la voix chuchotée, ainsi que pendant la phonation des sons non-voisés (ou **sourds**). Les sons voisés (ou **sonores**) résultent au contraire d'une vibration périodique des cordes vocales. Le larynx est d'abord complètement fermé, ce qui accroît la pression en amont des cordes vocales, et les force à s'ouvrir, ce qui fait tomber la pression, et permet aux cordes vocales de se refermer.

3.2.2 Variabilité du signal de parole

Lorsque on observe la représentation temporelle du signal de parole (figures 3.2 ; 3.3 ; 3.4), on peut constater de larges différences d'amplitude et de durées lors de la prononciation d'un même mot, d'un environnement à un autre, d'un locuteur à l'autre, mais aussi d'une prononciation à l'autre émanant du même locuteur [Buni-97].

3.2.2.1 Variabilité intra-locuteur

La variabilité intra-locuteur identifie les différences dans le signal produit par une même personne. Plusieurs critères peuvent être responsables de ces différences tels que la fatigue, la maladie affectant les organes phonatoire, l'état émotionnel du locuteur, etc.

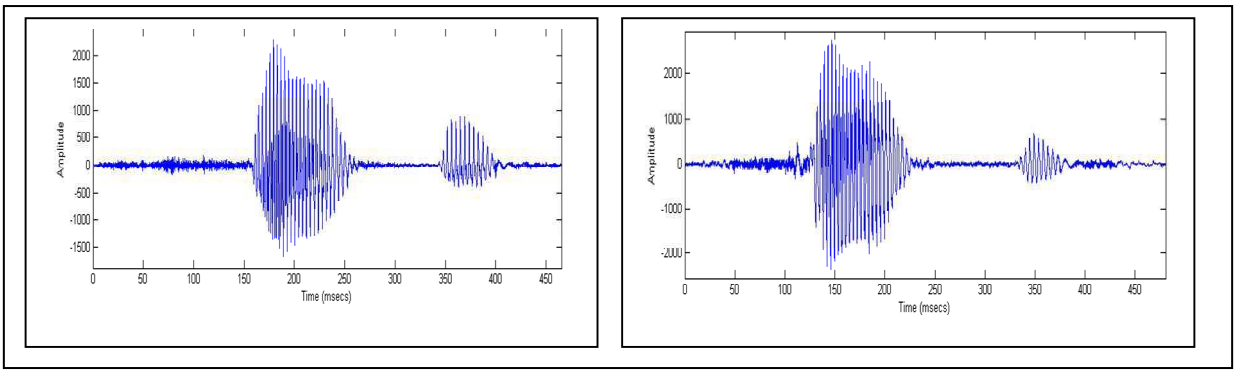


Figure 3.2 : Variabilité intra-locuteur pour le chiffre arabe « siffer ».

3.2.2.1 Variabilité inter-locuteurs

Des différences acoustiques importantes apparaissent dans un même mot prononcé par des locuteurs différents. Ces différences sont liés à l'âge, l'accent régional, le sexe, etc.

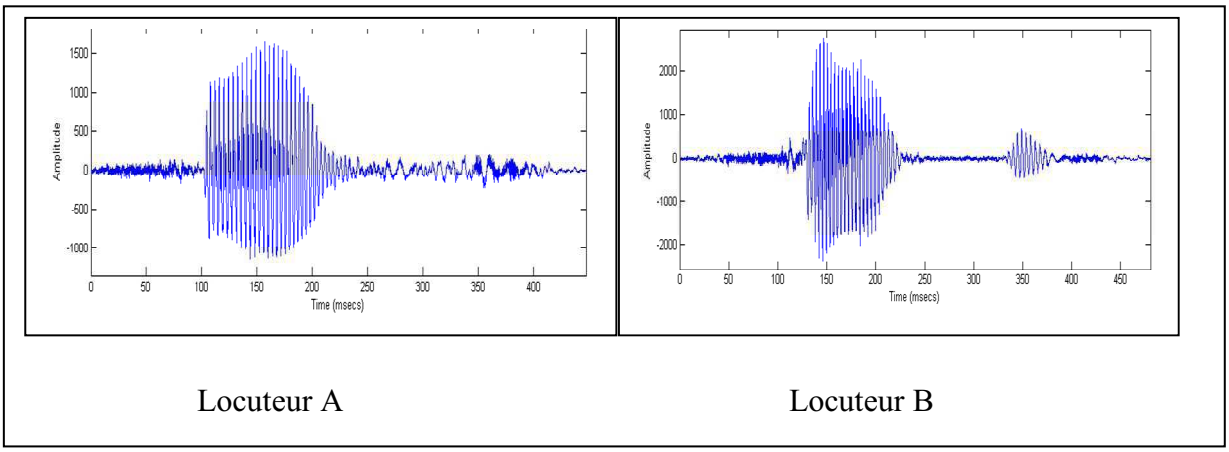


Figure 3.3: Variabilité inter-locuteurs pour le chiffre arabe « siffer ».

3.2.2.3Variabilité due à l'environnement

La variabilité due à l'environnement peut provoquer une véritable différence entre deux signaux de paroles sans que le locuteur ait modifié son mode d'élocution. Parmi les facteurs de variabilité liés à l'environnement on peut citer : la présence de bruit, la qualité de l'équipement (la ligne téléphonique et /ou du microphone), le canal de transmission, la position de ce dernier par rapport au locuteur, etc.

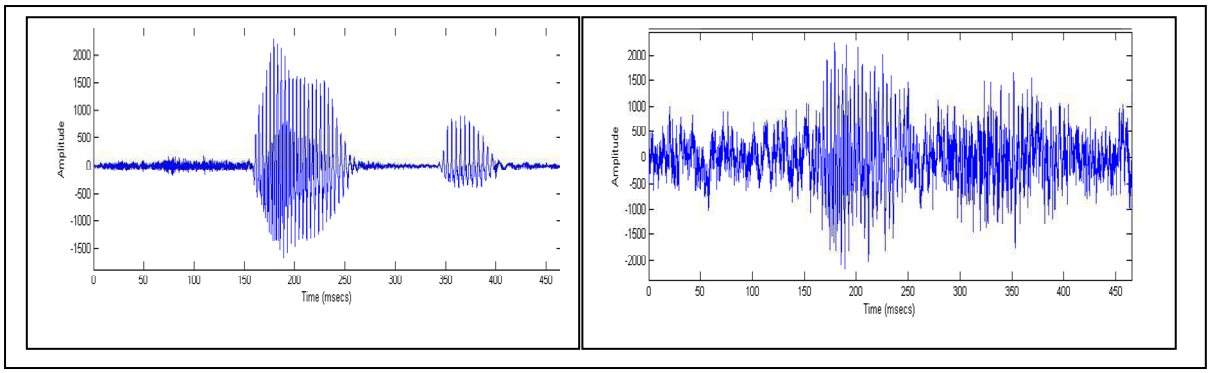


Figure 3.4: Variabilité due à un environnement bruité avec un niveau RSB (Rapport Signal/Bruit) = 0 dB d'un bruit d'usine pour le chiffre arabe « siffer ».

La transformation de ces signaux de parole en temps/fréquences/énergies (spectrogramme de la figure 3.5) nous permet de constater l'existence des zones fréquentielles de forme similaires tels que : la fréquence fondamentale et les formants (figure 3.6). Ce sont ces propriétés du signal que nous voulons exploiter pour rendre la modélisation acoustique du signal de parole plus robuste.

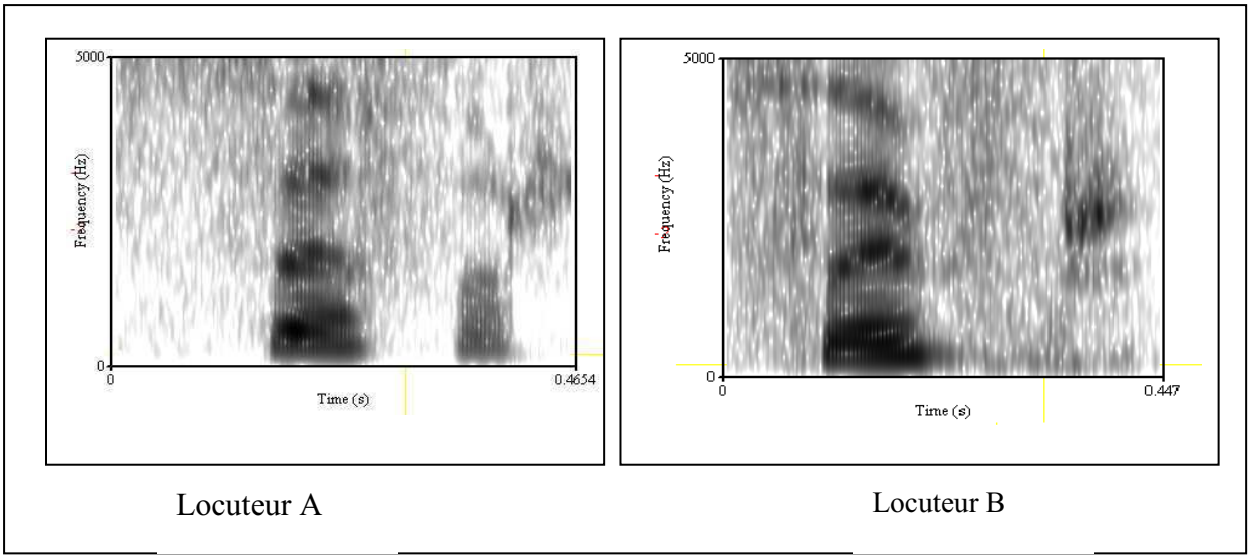


Figure 3.5: Spectrogrammes de chiffre arabe « siffer » pour les deux locuteurs A et B.

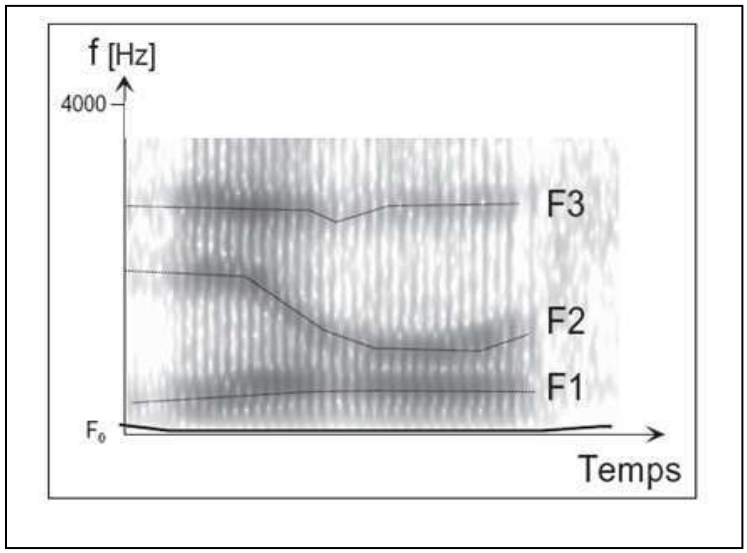


Figure 3.6: Fréquence fondamentale (ligne foncée) et la fréquence des trois premiers formants.

3.3 Les paramètres auxiliaires et méthodes d'extraction

La sensibilité des paramètres MFCC à la variabilité du signal à motiver plusieurs auteurs à la recherche de nouveaux paramètres pour rendre les modèles acoustiques plus robustes. On peut citer les travaux de Stephenson dans [Step-03] et Doss [Doss-05] qui utilisent dans un cadre de réseaux bayésiens (les réseaux bayésiens sont une variante des HMMs) comme paramètres auxiliaires : le pitch, l'énergie et la vitesse d'˝elocution de la parole. Par ailleurs, plusieurs travaux dans le domaine de la reconnaissance audio-visuelle cherchent à intégrer l'information visuelle dans le système de reconnaissance acoustique [Delé-96] [Rogo-99] [baki-08].

Nous avons choisi dans notre travail d'utiliser comme paramètres auxiliaires trois types de paramètres extraits directement à partir de signal de parole :

- La fréquence fondamentale (the fundamental frequency; pitch).
- l'intensité de la voix dite aussi l'énergie (energy)
- Les fréquences des trois premiers formants (formants).

Ces paramètres sont des caractéristiques fondamentales du signal de la parole qui résistent aux problèmes de la variabilité. La variation dans le temps de ces paramètres véhicule divers indices caractéristiques de l'individu que ce soit au niveau de son état physique (âge, sexe, physiologie), de son état émotionnel ou de son accent régional. De plus, ils sont moins sensibles au bruit par rapport aux coefficients cepstraux (MFCCs) [Mary-08].

3.3.1 La fréquence fondamentale

La fréquence fondamentale (F0) correspond à la fréquence de vibration des cordes vocales sous l'effet du passage de l'air à travers la glotte si l'on se place dans le domaine acoustique, ou à la hauteur de la voix par rapport au domaine perceptif. La plage de variation moyenne de cette fréquence varie d'un locuteur à l'autre en fonction principalement de son âge et de son sexe (de 80 à 200 Hz pour une voix masculine, de 150 à 450 Hz pour une voix féminine et de 200 à 600 Hz pour une voix d'enfant) [Roua-2005] [Lang-95]. Les algorithmes d'extraction de F0 utilisent une représentation temporelle ou spectrale du signal.

Les méthodes temporelles exploitent la similarité du signal d'une période à l'autre (Figure 3.7) pour identifier la période fondamentale. La fréquence fondamentale est alors l'inverse de la période fondamentale T0 :

$$F0 = 1/T0 \tag{3.1}$$

Dans le domaine fréquentiel, les algorithmes utilisent généralement les harmoniques de la fréquence fondamentale pour trouver F0. Cette propriété du signal peut être visualisée sur un spectrogramme du signal (Figure 3.6).

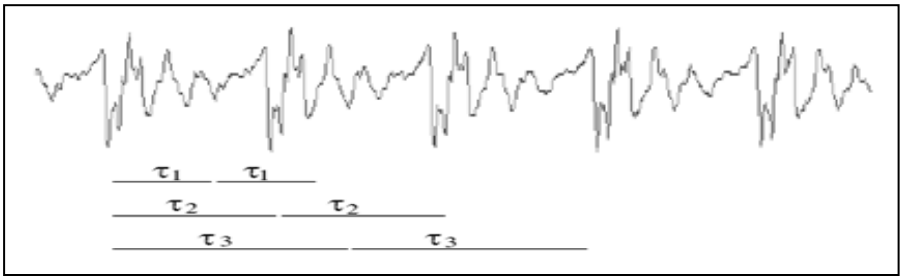


Figure 3.7 : La similarité du signal d'une période à l'autre

Il existe plusieurs algorithmes pour l'estimation de la fréquence fondamentale [Noll-67]. Un de ces algorithme qui est considéré comme simple et qui donne des résultats acceptables est l'algorithme basé sur la fonction d'autocorrélation du signal de parole dans le domaine temporel [Rabi-77].

3. 3.1.2 Méthode basée sur la fonction autocorrélation

Un algorithme d'extraction de la fréquence fondamentale peut en général se décomposer en trois phases successives :
1. un prétraitement du signal de parole (de la trame).
2. l'extraction du fondamental.
3. un post-traitement visant à corriger les erreurs.

Le prétraitement vise à optimiser les caractéristiques du signal en vue de l'extraction en utilisant, par exemple, un filtrage passe-bas ou un filtrage non linéaire.

La deuxième phase consiste à extraire la fréquence fondamentale et dépend donc de l'algorithme utilisé (pour la méthode d'autocorrélation l'estimation de la fréquence fondamentale est liée à la période pour laquelle la fonction d'autocorrélation est maximale)

La phase de post-traitement a pour but de diminuer les erreurs éventuelles et à lisser le contour du pitch en tenant compte des résultats antérieurs.

a) Définition de la fonction d'autocorrelation :
D'après Rabiner [Rabi-77], la fonction d'autocorrélation r_s du signal temporel *s(n)* est donnée par la relation:

$$r_s(m) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} s(n)s(n+m) \tag{3.2}$$

Si le signale *s(n)* est périodique, sa fonction d'autocorrélation est aussi périodique. Sa période est égale à celle du signal *s(n)* :

$$\forall n \ : \ s(n) = s(n+p) \qquad \Rightarrow r_s(m) = r_s(m+p) \tag{3.3}$$

Pour le signal de la parole qui est non stationnaire à long terme, la fonction d'autocorrélation est redéfinie pour une trame *w* de langueur *N*:

$$r_l(m) = \frac{1}{N} \sum_{n=0}^{N-m-1} [s(n+l)w'n)][s(n+l+m)w(n+m)] \qquad 0 \le m \le M_0 - 1 \tag{3.4}$$

M_0 : est le nombre de points de la fonction d'autocorrélation.

l : est l'indice du début de la trame.

b) Application de la fonction d'autocorrélation pour l'estimation du pitch

Nous résumons la méthode d'extraction de pitch basée sur la méthode d'autocorrélation introduite par L.Rabinner [Rabi-77] en trois étapes importantes :

1- Filtrage passe bas

L'utilisation d'un filtre passe- bas (aux alentours de 900 Hertz) permet l'élimination partielle des effets des formants d'ordre supérieure à 2 sur la fonction d'autocorrélation.

2- Dispositif non linéaire

Cette étape consiste à introduire le signal dans un dispositif non linéaire appelé 3-level center. Ce dispositif non linéaire rend le calcul de la fonction d'autocorrélation plus simple et le spectre de puissance du signal plus plat pour mieux détecter le pic correspondant au pitch. La relation entre l'entrée et la sortie du dispositif est donnée par l'équation suivante :

$$Y(n) = \begin{cases} 1 & si \quad s(n) \ge C_L \\ 0 & si \quad |s(n)| \ge C_L \\ -1 & si \quad s(n) \le -C_L \end{cases} \tag{3.5}$$

Où C_L est le seuil d'écrêtage (center clipped signal). Il est défini par la relation suivante :

$$C_L = 0.64 \times M \tag{3.6}$$

Tel que *M* représente le plus petit des maximums absolus Max1 et Max3 calculés respectivement sur le premier et le dernier tiers de la fenêtre d'analyse.
La figure suivante montre le graphe correspondant au dispositif non linéaire :

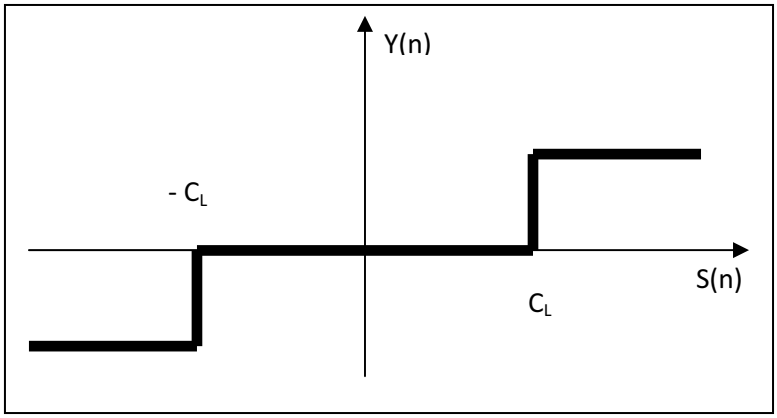


Figure 3.8 : graphe de la fonction 3-level center clipping.

Introduire le signal dans ce dispositif revient à coder chaque échantillon *s(n)* sur trois niveaux (-1, 0, 1). De ce fait la fonction d'autocorrélation du signal codé est particulièrement simple à calculer.

3- Calcul de la fonction d'utocorrélation :

Une fois le signal *s* est codé, on procède au calcul de sa fonction d'autocorrélation r_s puis à la recherche du maximum de cette fonction. La décision de voisement est déterminée en comparant R(0) au plus grand pic $R_{max}(k)$:

$$\begin{cases} \dfrac{R_{\max}(k)}{R(0)} \le S & \text{, la séquence est voisée.} \\ \dfrac{R_{\max}(k)}{R(0)} > S & \text{, la séquence est non voisée.} \end{cases} \tag{3.7}$$

Tel que S est un seuil à déterminer expérimentalement (exemple 30%).

La fréquence fondamentale est calculée pour les trames voisées comme suit :

$$F_0 = \frac{f_e}{K_{\max}} \tag{3.8}$$

Où :

f_e : est la fréquence d'échantillonnage du signal.

$K_{\max}$: est l'abscisse de $R_{\max}(k)$

3.3.2 L´énergie

L'énergie ou l'intensité du signal correspond à la variation de l'amplitude de signal de parole causée par une force plus ou moins forte provenant du pharynx et provoquant une variation de la pression de l'air sous la glotte. Ce descripteur permet de fournir une mesure de la force sonore de la voix (faible ou forte).

L'énergie à court terme d'un signal échantillonné sur une fenêtre de longueur T, $(s_t)_{t=1,T}$, est définie par :

$$E = \frac{1}{T}\sum_{t=1}^{T} s_t^2 \tag{3.9}$$

Pour respecter l'échelle perceptive, elle est généralement exprimée en décibels :

$$E_{db} = 10 \times \log_{10}(\frac{1}{T}\sum_{t=1}^{T} s_t^2) \tag{3.10}$$

Pour éliminer la variabilité de ce paramètre liée aux conditions d'enregistrement, cette énergie est normalisée par rapport au maximum d'énergie observé sur le signal global.

3.3.3 Les fréquences formantiques

Lorsque les cordes vocales entrent en vibration, elles fournissent un signal dont le résonateur (le conduit vocal) va amplifier certaines composantes. On obtient alors des formants qui sont un facteur fondamental dans la caractérisation du conduit vocal [Clav-2007].
Le nombre de formants, selon les caractéristiques du résonateur (volume, forme et ouverture), est variable d'un seul à théoriquement une infinité. Néanmoins, du point de vue perceptif, seuls quelques-uns d'entre eux jouent un rôle central au niveau de la parole. Par exemple, on peut caractériser toute voyelle en ne prenant en compte que ses trois premiers formants. Pour une réalisation de la voyelle [i] par exemple, les trois premiers formants sont généralement situés respectivement à 300, 2200 et 3000 Hz.

En fait, un formant ne peut jamais être ramené à une fréquence fixe (sinon de manière conventionnelle, en effectuant une moyenne par exemple, comme pour la voyelle [i] ci-dessus). Il s'agit plutôt d'une bande de fréquences qui sera d'autant plus large que le système est amorti. Ces régions formantiques apparaissent très clairement sur les spectrogrammes (figure 3.6 précédente).

3.3.3.1 Estimation des formants

La détermination précise des formants utilise plus couramment l'analyse LPC (section I.3.1.3), les formants sont alors estimés à partir d'une recherche des maximas du spectre du modèle LPC [Dekk-03].

Ces maximas correspondent également aux racines complexes du polynôme défini par l'équation (I.7), rappelée ici :

$$A(z) = 1 + \sum_{i=1}^{P} a_i z^{-i} = 0 \tag{3.11}$$

En effet, à une racine $Z_i = e^{sT}$ proche du cercle unité correspond à un formant F_i de bande passante B_i où $s = -\pi B \pm j2\pi F$.

Si :

$$Z_i = \mathrm{Re}(Z) + j\,\mathrm{Im}(Z) \tag{3.12}$$

Telles que *Re* et *Im* correspondent respectivement la partie réelle et imaginaire de la racine complexe de polynôme $A(z)$. L'expression de la bande passante *B* et de la fréquence *F* sont données par les relations suivantes :

$$B = -\frac{f_e}{\pi} Ln(Z) \qquad \text{(Hertz)} \tag{3.13}$$

$$F = \frac{f_e}{2\pi} Tan^{-1}\left(\frac{\mathrm{Im}(Z)}{\mathrm{Re}(Z)}\right) \qquad \text{(Hertz)} \tag{3.14}$$

Où f_e est la fréquence d'échantillonnage.

Après cette présentation des paramètres auxiliaires avec leurs méthodes d'extraction, nous allons procéder dans la section qui suit à la description des stratégies de la fusion de ces paramètres auxiliaires au sien d'un système de RAP standard.

3.4 Stratégies de la fusion

La fusion des paramètres peut prendre différentes formes selon le niveau ou elle est effectuée. En s'inspirant des méthodes de fusion appliquées entre autre dans le domaine de la reconnaissance audio-visuelle [Delé-96] [Rogo-99] [Baki-08], nous avons étudié deux principales stratégies d'intégration des paramètres auxiliaires dans un système de la RAP. Dans la première stratégie dite « à identification direct ou fusion de paramètres», les deux types de paramètres (standards et auxiliaires) sont concaténés dans le même vecteur acoustique pour former une seule observation acoustique à l'entrée d'un système de RAP. Une deuxième stratégie dite «à identification séparée ou fusion des scores» consiste à modéliser chaque type de paramètres par un sous système de reconnaissance indépendant, et de fusionner les scores (les probabilités) à la sorties des deux sous systèmes.

3.4.1 Identification Directe (ID)

C'est la manière la plus naturelle pour intégrer les informations auxiliaires dans un système de la RAP standard. Le processus de fusion consiste à concaténer les paramètres standards et auxiliaires dans le même vecteur acoustique à l'entrée de système de reconnaissance, comme le montre la figure 3.9. Cette stratégie d'intégration est dite par « identification directe », puisque les deux types de paramètres sont intégrés directement dans le système de reconnaissance sans aucun prétraitement et avant tout processus de classification (reconnaissance).

Dans cette stratégie les paramètres auxiliaires sont considérés comme s'il s'agissait de paramètres cepstraux supplémentaires, sans tenir compte de leurs natures hétérogènes (deux prétraitements différents pour le même signal acoustique). Ainsi les deux types de paramètres sont traités conjointement dans un même modèle de Markov (figure 3.10).

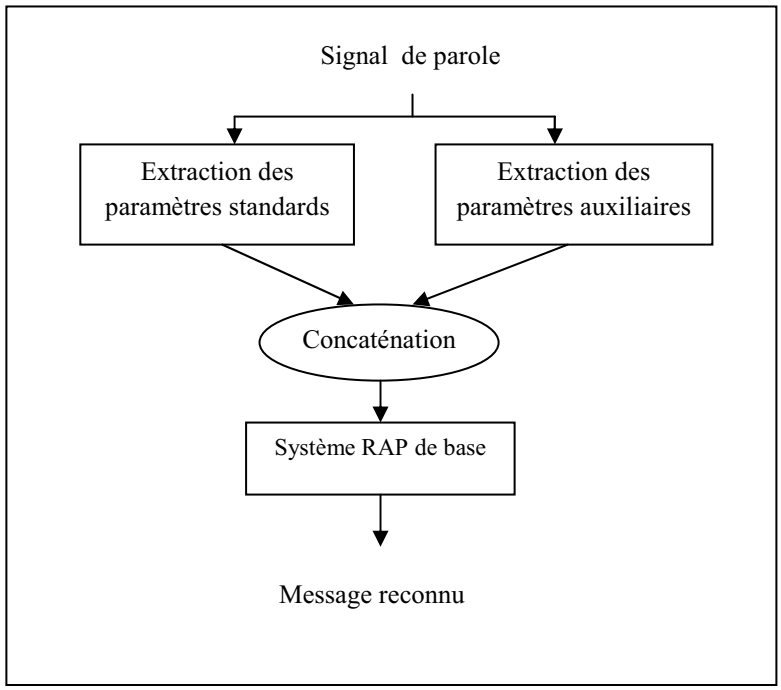


Figure 3.9 : Fusion des données auxiliaires dans un système RAP par identification direct.

Le vecteur d'observation du système ID est donc constitué des paramètres standards et des paramètres auxiliaires fusionnés. Nous supposons toujours l'indépendance statistique entre les deux types de paramètres, ce qui nous permet d'utiliser des matrices de covariances diagonales pour modéliser le nouveau flux acoustique standard-auxiliaire.

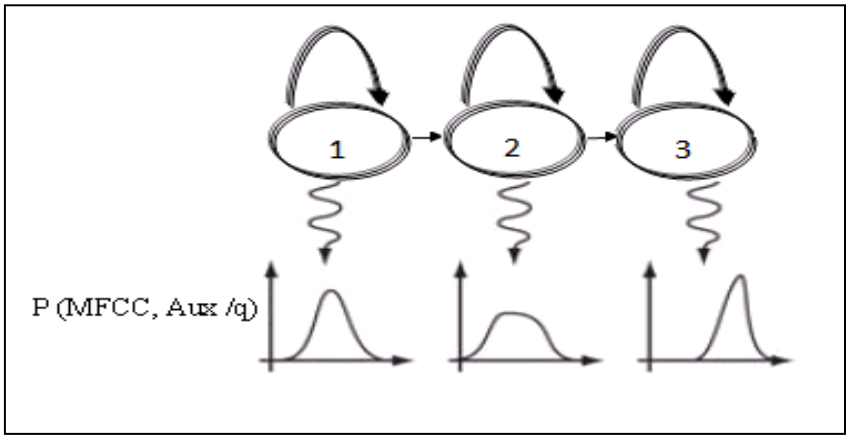


Figure 3.10: Modélisation des paramètres standards et auxiliaires dans un même modèle HMM.

3.4.2 Identification Séparée (IS)

Dans cette stratégie, chaque type de paramètres (standard et auxiliaire) est modélisé par un sous système de reconnaissance séparé. Les résultats (scores probabilistes) en sortie de chacun de ces sous systèmes de reconnaissance sont fusionnés dans un module de décision qui produit le résultat final.

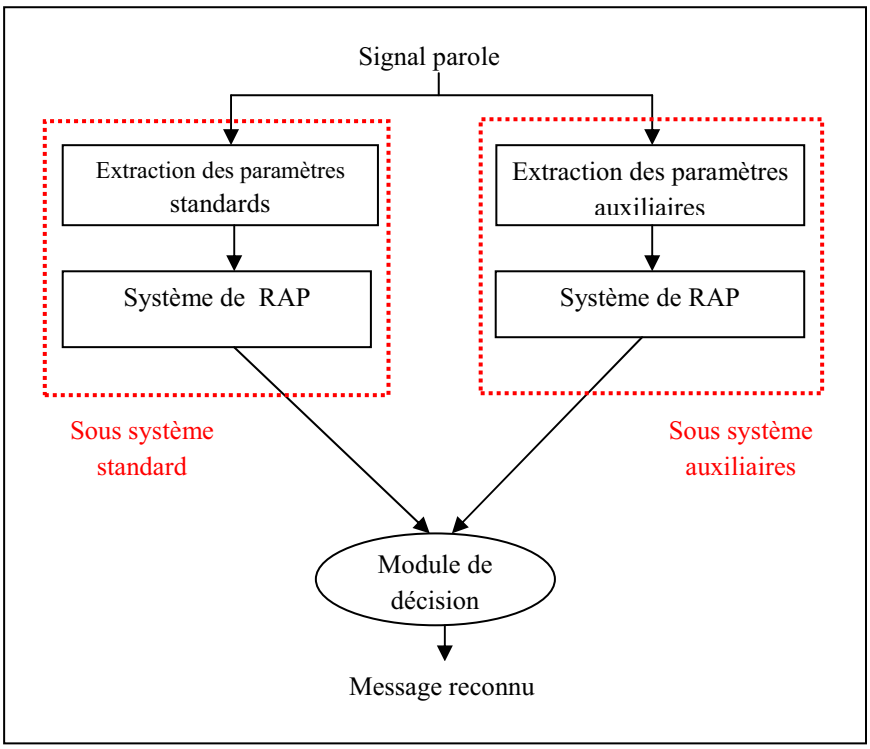


Figure 3.11: Fusion des données auxiliaires dans un système RAP par identification séparée.

Les paramètres standards et auxiliaires sont traités séparément dans chacun des sous systèmes utilisant respectivement des modèles de Markov standards et auxiliaires qui n'ont pas nécessairement la même topologie, comme le montre la figure 3.12.

Les deux sous systèmes, standard et auxiliaire, sont pratiquement identiques à un système de la RAP standard. La seule différence réside dans l'utilisation non pas de l'algorithme de décodage de Viterbi classique mais d'un algorithme permettant d'obtenir les N meilleures solutions de reconnaissance différentes [Chow-89]. La décision de reconnaissance consiste alors à choisir parmi les N meilleures solutions proposées celle qui représente le mieux l'unité à reconnaître.

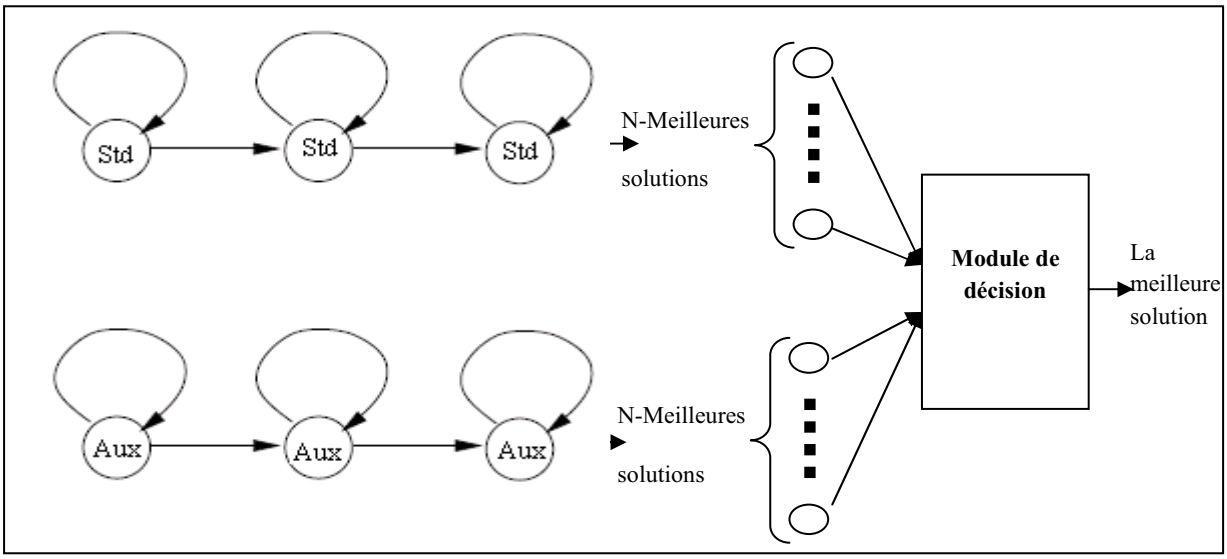


Figure 3.12 : Exemple de modélisation markovienne des paramètres standards et auxiliaires par identification séparée.

3.4.2.1 Méthodes de fusion

Plusieurs méthodes peuvent être utilisées pour réaliser la fusion des données dans notre cas les scores probabilistes à l'intérieur d'un module de décision. Parmi ces méthodes, nous pouvons citer :

- **La fusion dans un cadre stochastique**

Il s'agit de la méthode de fusion la plus simple, où la sortie du module de décision n'est que l'entité correspondante au maximum de l'addition vectorielle des vecteurs probabilistes $standard\ (p_1,.....,p_N)$ et $auxiliaire(p'_1,....., p'_N)$ obtenues respectivement à la sortie des deux sous systèmes pour une même entité à reconnaître. Cette fusion peut être formulée par l'équation 3.15 suivante :

$$Sortie = \arg \underset{N}{Max} (\frac{1}{2} \sum_{i=1}^{N} standard(p_i) + auxiliaire(p'_i)) \tag{3.15}$$

Avec

Sortie : représente la décision obtenue à la sortie du module de décision.

N : représente les N meilleurs scores obtenues à la sorties des deux sous systèmes standard et auxiliaire.

Cette méthode de fusion est fortement liée à la fiabilité des scores standards et auxiliaires qui change d'un contexte à un autre (présence ou absence du bruit). Ceci crée la nécessité d'adapter la simple somme de l'équation (3.15) à la fiabilité relative des deux types de paramètres, en utilisant par exemple une combinaison linéaire des scores. L'équation (3.15) peut être reformulée comme suit :

$$Sortie = \arg \underset{N}{Max} (\sum_{i=1}^{N} \alpha \quad standard(p_i) + \beta \quad auxiliaire(p'_i)) \tag{3.16}$$

Tel que

$\alpha + \beta = 1$: α et β représentent le degré de fiabilité supposé pour les paramètres standards et auxiliaires respectivement. Ceci pose le problème du choix des pondérations, qui change d'un contexte à un autre.

- **La fusion par la technique du vote**

Cette méthode de fusion est appelée méthode du vote et est couramment utilisée pour la fusion de données provenant de différents modalités. La décision repose sur l'accord de la majorité des candidats (les différents sous systèmes ou modalités) à l'entrée du module de décision. Pour chaque entité lexicale modélisée par le système de reconnaissance, on calcule le nombre de candidats mis en accord pour qu'elle corresponde à l'entité à reconnaitre. Le module de décision rend l'entité lexicale dont le nombre de mise en accord est le plus grand. La fiabilité de cette méthode de fusion exige la présence d'un nombre assez grand de candidats (au moins trois candidats).

- **La fusion neuronale**

Un autre moyen de fusion des données est le cadre neuronal, la fusion des scores est dans ce cas réalisée au moyen d'un réseau de neurones qui choisit le meilleur score parmi les 2N scores fournis en entrée, son choix est basé bien entendu sur une phase d'apprentissage préalable. C'est ce type de fusion que nous avons utilisé dans notre travail et qui sera donc développée plus en détail dans la section suivante.

3.5 Réseaux de neurones

C'est initialement par analogie avec le fonctionnement du cerveau humain que les réseaux de neurones ont été conçus. De tels réseaux sont composés d'états appelés "neurones formel" reliés les uns avec les autres avec des transitions appelées "synapses". Les réseaux de neurones ont des propriétés de simulation, de classification, de représentation et d'apprentissage [Jaco-95].

3.5.1 Neurone formel

Un neurone formel est l'unité élémentaire de traitement d'un réseau de neurones. Il est connecté à des scores d'information en entrée, et renvoie une information en sortie qui est une somme pondérée des valeurs de ces entrées (figure 3.13).

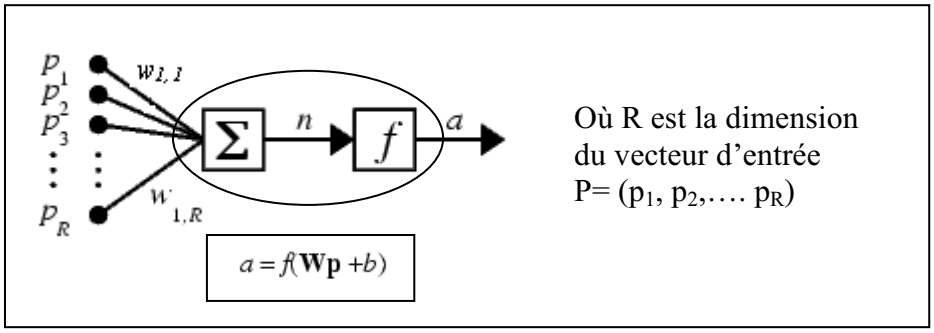


Figure 3.13: Schéma d'un neurone formel.

Comme nous pouvons le voir sur la figure 3.13, pour *R* entrées, la sortie est donnée par la formule générale suivante :

$$a = f(\sum_{i=1}^{R} w_{j,i} p_i + b)) \tag{3.15}$$

où :

$P = (p_1, p_2, ..., p_R)$ représente le vecteur d'entrée de taille R.
$W_j = (w_{j,1}, w_{j,2} .... w_{j,R})$ représente le vecteur de poids.

b : est un seuil dit également biais.

f : est une fonction d'activation, appelée aussi fonction de transfert. Elle sert à introduire une non-linéarité dans le fonctionnement du neurone. Une des fonctions de transfert très courante est la fonction sigmoïde, mais d'autres fonctions peuvent également être utilisées (figure 3.14).

a : représente la sortie du neurone.

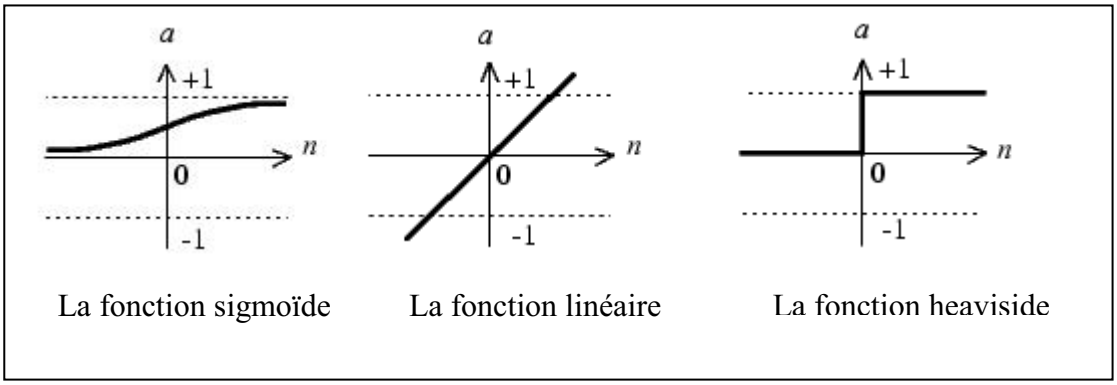


Figure 3.14: Quelques fonctions d'activation.

3.5.2 Réseau de neurones multicouches

Un réseau de neurones multicouches, appelé aussi Perceptrons Multi-Couches (PMC) est défini à partir d'une répartition des neurones en plusieurs couches. Les informations en entrée sont connectées à tous les neurones de la première couche, dite couche d'entrée, et ainsi de suite les sorties des neurones de la couche i forment les entrées de la couche *i+1* jusqu'à la dernière couche, appelée couche de sortie. Toutes les couches exceptées la couche d'entrée et la couche de sortie son considérées comme couches cachées (figure 3.15).

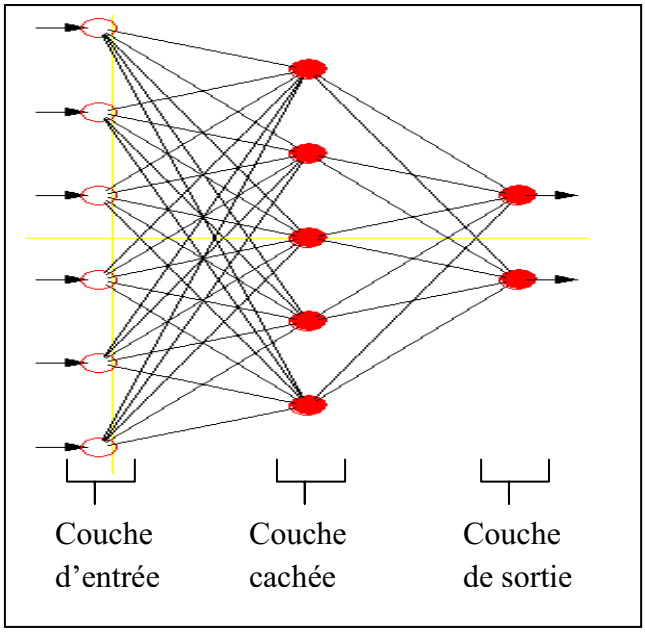


Figure 3.15: Réseau de neurones à trois couches : couche d'entrée avec 6 neurones, couche cachée avec 5 neurones et couche de sortie avec 2 neurones.

Comme dans le cas des modèles de Markov cachés, un réseau de neurones doit être appris sur une base de données d'apprentissage avant d'être utilisé en phase de classification. En général, il existe deux modes d'apprentissage pour les réseaux de neurones :

- Un apprentissage supervisé : lorsque l'on force le réseau à converger vers un état final précis, en même temps qu'on lui présente une donnée d'entrée. C'est le mode d'apprentissage utilisé dans notre travail.

- **Un apprentissage non-supervisé** : à l'inverse du premier mode, lors d'un apprentissage non-supervisé, le réseau est laissé libre de converger vers n'importe quel état final lorsqu'on lui présente une donnée d'entrée.

3.5.3 Apprentissage supervisé d'un réseau de neurones

L'apprentissage supervisé d'un réseau de neurones multicouches est généralement réalisée à partir de l'algorithme de rétro propagation du gradient (Backpropagation en anglais) [Skap-92] qui consiste à comparer les résultats de la couche de sortie avec les résultats attendus, en calculant la valeur de l'erreur et en la minimisant. La minimisation de l'erreur revient à ajuster les poids et les seuils de chaque neurone. Il existe plusieurs critères d'erreur qui peuvent être utilisés.
Notons par T l'ensemble des données d'entraînement, les sorties réelles y et les sorties désirées ε. Les critères les plus connus sont :
- Le critère des moindres carrés :

$$E = \frac{1}{2}\sum_{t\in T}\sum_{l=1}^{L}\left(y_l^{(t)} - \varepsilon_l^{(t)}\right)^2 \tag{3.16}$$

- Le critère entropique

$$E = \frac{1}{2}\sum_{t\in T}\sum_{l=1}^{L}\varepsilon_l^{(t)}\ln(y_l^{(t)}) \tag{3.17}$$

Le gradient de l'erreur est employé pour ajuster les poids de l'élément de sortie et est propagé pour modifier les poids des couches cachées (d'où le nom de rétro-propagation de la couche de sortie vers la couche cachée). L'ajustement des paramètres se fait généralement selon la formule :

$$W^{T+1} = W^T - \eta\left.\frac{\delta E}{\delta w_{ij}}\right|_{W^T} \tag{3.18}$$

Où η est appelé taux d'apprentissage et doit être assez faible pour garantir la convergence du processus et assez grand pour éviter une convergence trop lente.

3.5.4 Classification

La classification (la reconnaissance) consiste simplement à injecter les données de test dans le réseau de neurones (déjà entraîné). Elles sont propagées jusqu'à la couche de sortie donnant le résultat.

3.6 Conclusion

Nous avons présenté dans ce chapitre les paramètres auxiliaires que nous avons étudiés pour les intégrer dans le système de RAP standard (basé seulement sur les paramètres standards MFCCs) afin de le rendre plus robuste aux conditions réelles. Nous avons présenté les méthodes d'extraction et les différentes techniques employées pour les intégrer dans le système de RAP standard en s'inspirant des méthodes de fusion appliquées dans le domaine de la reconnaissance audio-visuelle.

Résultats expérimentaux

4.1 Introduction

Ce chapitre présente le contexte expérimental et l'évaluation de notre travail. Notre objectif était de développer un système de reconnaissance fondé sur les modèles de Markov cachés pour lequel nous incorporons des sources d'informations auxiliaires en plus des paramètres standards utilisés actuellement, et ce afin de le rendre plus robuste à la variabilité du signal de parole. Notre système est mis au point à partir de la plate-forme HTK (Hidden Markov Model Toolkit, ou "boîte à outils de modèles de Markov cachés") de l'Université de Cambridge et évaluer sur deux types de base de données : la base de données de mots isolés ARADIGIT [Amro-07] et la base de données de la parole continue TIMIT [Sene-88].

Notre travail consiste donc en la réalisation de deux systèmes l'un standard et l'autre basé sur la fusion des données.

- Le système de reconnaissance standard est construit avec une représentation du signal par les coefficients cepstraux de type MFCC et leurs coefficients différentiels.

- Le système basé sur la fusion est construit en fusionnant les paramètres standards avec les paramètres auxiliaires le pitch, les fréquences des trois premiers formants et l'énergie selon deux stratégies d'intégration.

4.2 Contexte expérimental

Nous présentons dans cette section les différents outils logiciels qui nous ont permis de développer notre système de reconnaissance ainsi que les différentes bases de données utilisées pour l'évaluer.

4.2.1 La plate-forme HTK

La plate-forme HTK a été développée à l'Université de Cambridge par S.J. Young et son équipe. Elle est constituée d'un ensemble d'outils logiciels qui permettent de construire des systèmes de reconnaissance de la parole à base de modèles de Markov cachés [Youn-05]. HTK offre une très grande liberté de choix tout au long de la construction du système de reconnaissance. Les modèles peuvent représenter des mots ou tout type d'unité sub-lexicale, et leur topologie est librement configurable. Les densités de probabilité d'émission, qui sont associées aux états, sont décrites par des multi-gaussiennes. Les modèles sont initialisés avec l'algorithme de Viterbi, puis ré-estimés par l'algorithme optimal de Baum-Welch. Le décodage est réalisé par l'algorithme de Viterbi, sous la contrainte d'un réseau syntaxique défini par l'utilisateur et éventuellement d'un modèle de langage de type bi-gramme dans la plupart de cas. Les résultats sont enfin évalués par alignement dynamique avec la chaîne phonétique ou lexicale de référence.

L'ensemble de ces outils est écrit en langage C, et la documentation détaille leur utilisation et les principes de leur implémentation, ce qui rend l'outil HTK largement répandu dans le monde de la recherche. En 1992, ses concepteurs revendiquaient déjà plus d'une centaine d'utilisateurs. Tous ces avantages nous ont encouragés à construire notre système de reconnaissance avec HTK.

4.2.1.1 Présentation d'HTK

HTK dans sa version 3.3 est structuré comme le montre la figure 4.1.

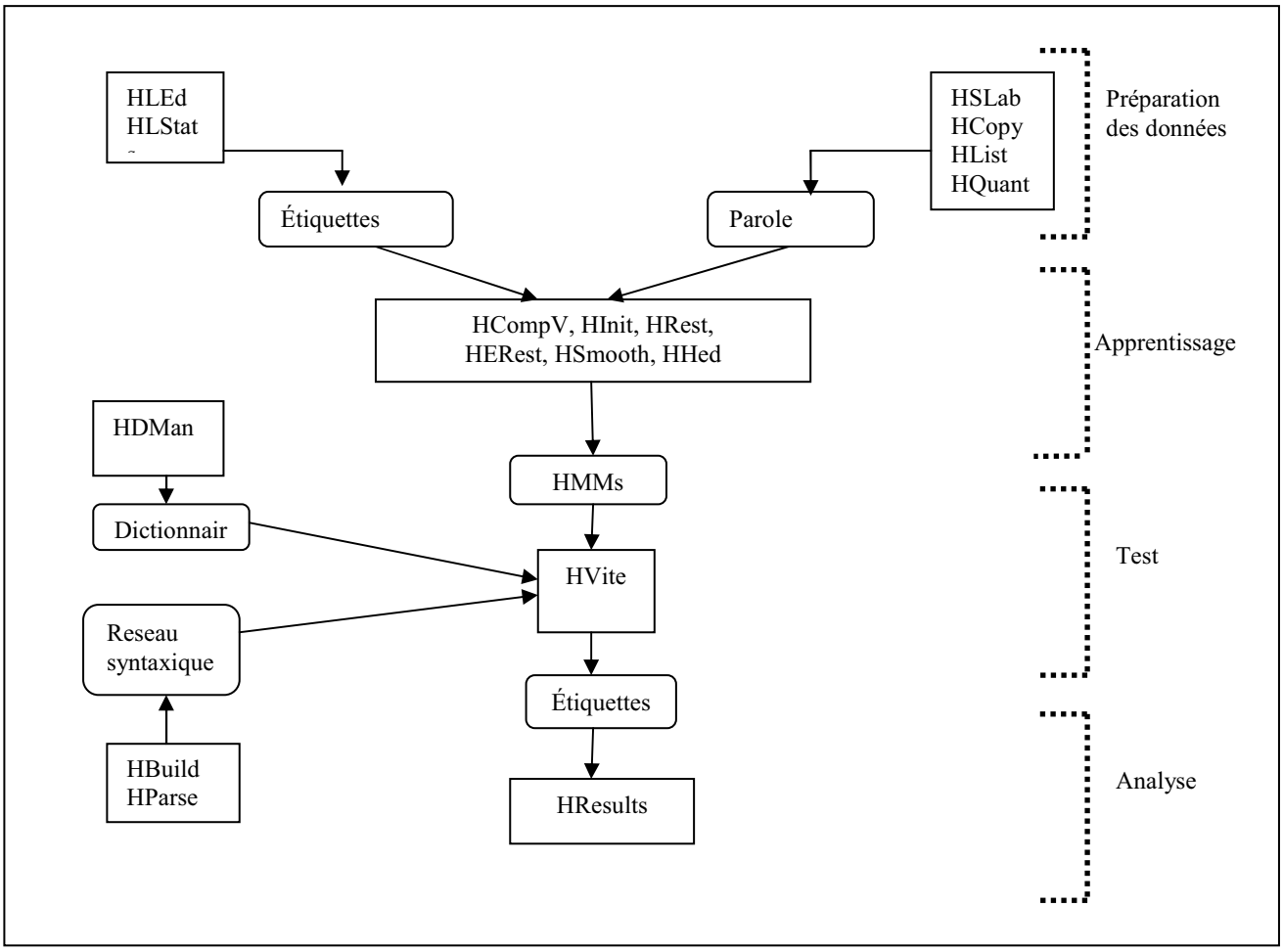


Figure 4.1 : Structure d'un système de reconnaissance avec HTK.

Les principaux outils de base de HTK s'enchaînent naturellement pour réaliser les différentes étapes d'un système de reconnaissance, ces outils ainsi que leurs descriptions sont donnés dans le tableau suivant :

outils	Rôle
Hbuild	Conversion de modèles de langage dans différents types de format.
HcompV	Calcul de la moyenne et de la variance sur un ensemble de données d'apprentissage.
Hcopy	Calcul des paramètres de fichiers signaux.
HDMan	Édition des dictionnaires.
HERest	Phase d'apprentissage - Ré-estimation des HMM en continu (Baum-Welch).

HHEd	Édition des HMM.
Hinit	Phase d'apprentissage - Initialisation d'un HMM.
Hled	Édition des fichiers d'étiquettes.
Hlist	Visualisation en format texte des fichiers de données
HLStats	Calcul de statistiques sur les étiquettes.
Hparse	Génération du graphe de décodage.
Hquant	Quantification vectorielle pour HMM discret.
Hrest	Phase d'apprentissage - Ré-estimation d'un HMM (Baum-Welch).
Hresults	Résultats du décodage (alignement dynamique entre les fichiers de résultats et de références).
HSGen	Générateur automatique de phrase en fonction d'une grammaire.
HSLab	Affichage du signal et des étiquettes.
Hsmooth	Lissage des paramètres des HMM.
Hvite	Décodage parole continue (Viterbi).

Tableau 4.1 : Outils logiciels de base de HTK (Version 3.3)

On va détailler maintenant les étapes d'utilisation de HTK comme elles sont indiquées dans la figure 4.1.

4.2.1.2 Préparation des données

Avant l'apprentissage des modèles, il est nécessaire de préparer les données d'apprentissage en calculant les paramètres du signal et en étiquetant les phrases d'apprentissage :

a- La représentation du signal : est obtenue avec l'outil ***HCopy,*** qui produit en particulier des coefficients MFCC. Ainsi que leurs coefficients différentiels du premier et du second ordre. Ces derniers peuvent être calculés ultérieurement lors de la lecture des fichiers de paramètres, ce qui économise leur stockage en mémoire de masse.

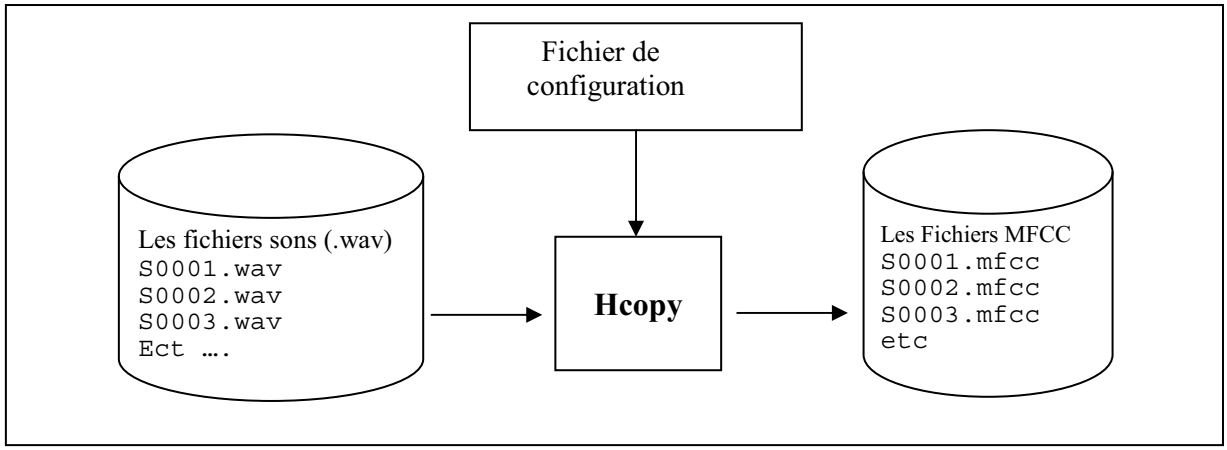


Figure 4 .2: Représentation acoustique du signal.

b- l'étiquetage de la base d'apprentissage : les phrases d'apprentissage doivent être toutes étiquetées en fonction des unités acoustiques modélisées. Les bases de données de parole sont parfois fournies avec un étiquetage phonétique qui ne correspond pas exactement aux unités acoustiques modélisées (comme dans notre cas avec la base TIMIT). L'éditeur ***HLEd*** permet alors de modifier les étiquettes, par exemple pour regrouper plusieurs phonèmes différents dans une seule classe. Dans le cas de la base de données de mots isolés, puisque chaque mot est modélisé par un modèle séparé, l'étiquetage revient à donner le mot lexical correspond à chaque mot de la base.

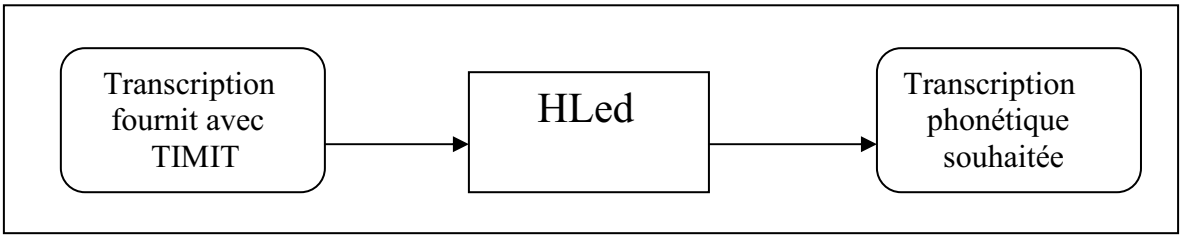


Figure 4. 3 : Transcription phonétique de la base de données de la parole continue TIMIT.

C- Topologie des modèles : Pour chaque unité acoustique, il faut définir un modèle prototype contenant la topologie choisie, à savoir :
- Le nombre d'états du modèle.
- Les transitions possibles entre les états.
- Le type de loi de probabilité associée à chaque état.

Les probabilités d'émission associées aux états sont décrites par une combinaison linéaire de gaussiennes, caractérisées par leur moyenne et leur matrice de covariance dans l'espace des paramètres. La matrice de covariance est théoriquement symétrique, mais peut être choisie diagonale si l'on suppose l'indépendance entre les composantes des vecteurs de paramètres.

4.2.1.3 Apprentissage

L'apprentissage des modèles de Markov est une étape essentielle dans la construction d'un système de la RAP. C'est la qualité de cette modélisation qui conditionne en grande partie les résultats de la reconnaissance.

L'apprentissage consiste à estimer les paramètres des modèles de Markov : les probabilités de transition, les densités d'observation associées aux états, c'est à dire les vecteurs de moyennes et les matrices de covariances d'un ensemble de gaussiennes, ainsi que les pondérations permettant d'établir des mélanges à partir de ces gaussiennes.

L'apprentissage des modèles HMMs nécessite trois étapes décrites comme suit :

- Créer le prototype initial : l'initialisation de ce prototype est indépendante de tout corpus d'entraînement, elle sert juste à définir la topologie du prototype initial et non à l'initialisation de ses paramètres [Young-97]. Nous avons opté pour les mêmes paramètres initiaux fournis avec l'exemple de démonstration de la boite outil HTK :

– Initialisation des vecteurs moyennes μ_k par des zéros. Ceci en supposant que les observations acoustiques sont des variables aléatoires centrées réduites.
– Initialisation des matrices de covariance Σ_k par des matrices diagonales unitaires. Ceci en supposant la décorrélation entre tous les paramètres MFCCs d'une même observation.
– Initialisation équiprobable des poids d'une loi de probabilité gaussienne:

$$c_k = 1/\text{nombre de gaussiens.} \qquad (4.1)$$

-Initialisation de l'apprentissage : Pour chacun des prototypes initiaux (modélisant une unité acoustique), l'outil ***HInit*** initialise les probabilités d'émission des états du modèle au moyen d'une procédure itérative basée sur l'algorithme de Viterbi (figure 4.4). Cette phase aide à répartir d'une façon optimale les trames d'un vecteur acoustique sur l'ensemble des états du modèle correspondant.

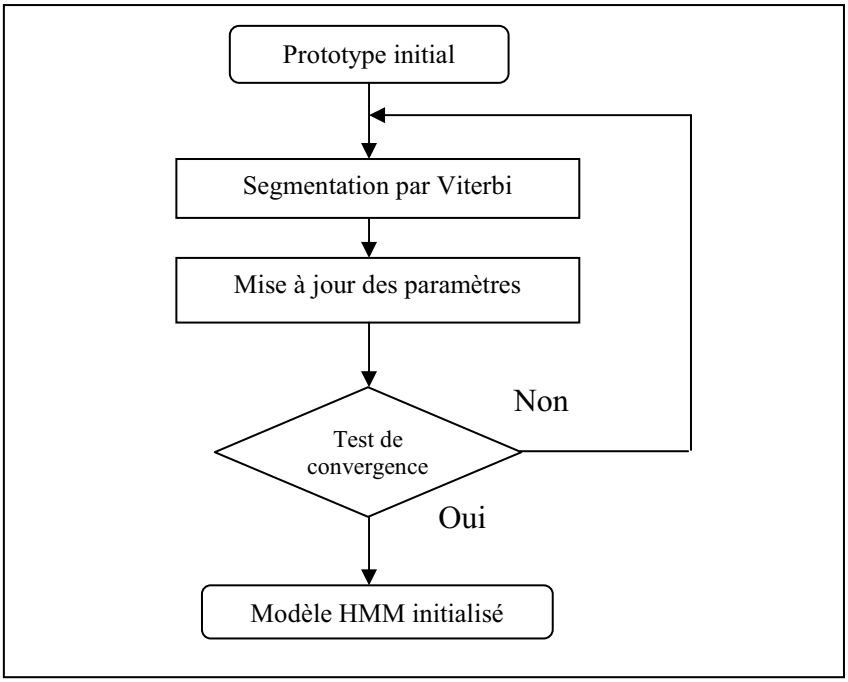


Figure 4.4: Initialisation d'un modèle HMM avec Viterbi.

-Estimation des paramètres d'apprentissage : l'estimation des paramètres d'un modèle est effectuée avec l'outil ***HRest***, qui applique l'algorithme optimal de Baum-Welch jusqu'à la convergence et ré-estime les probabilités d'émission et de transition.

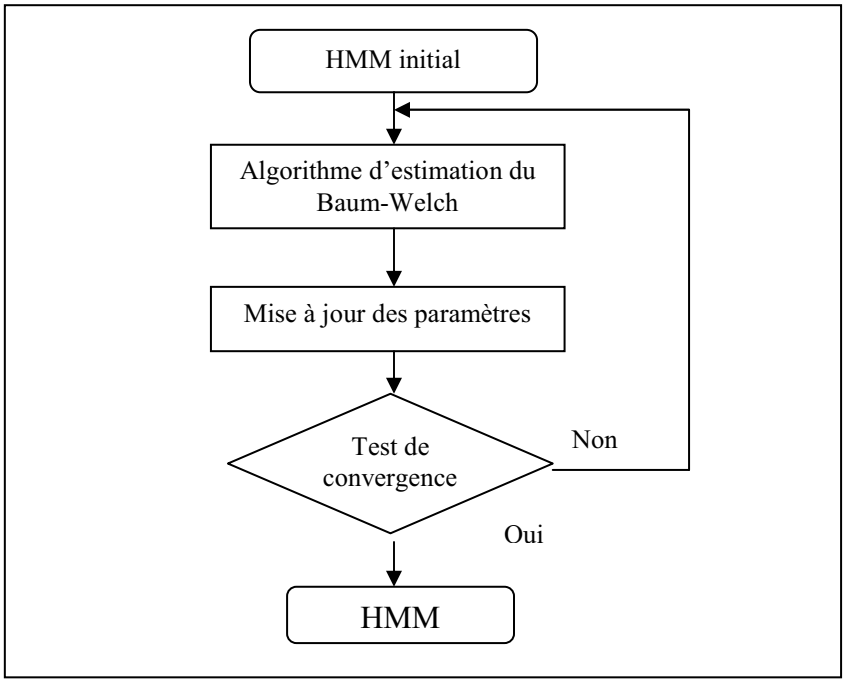


Figure 4.5: Estimation des paramètres d'un modèle HMM avec l'algorithme de Baum-Welch.

L'utilisation de ces deux outils d'apprentissage peut être résumée par le schéma suivant :

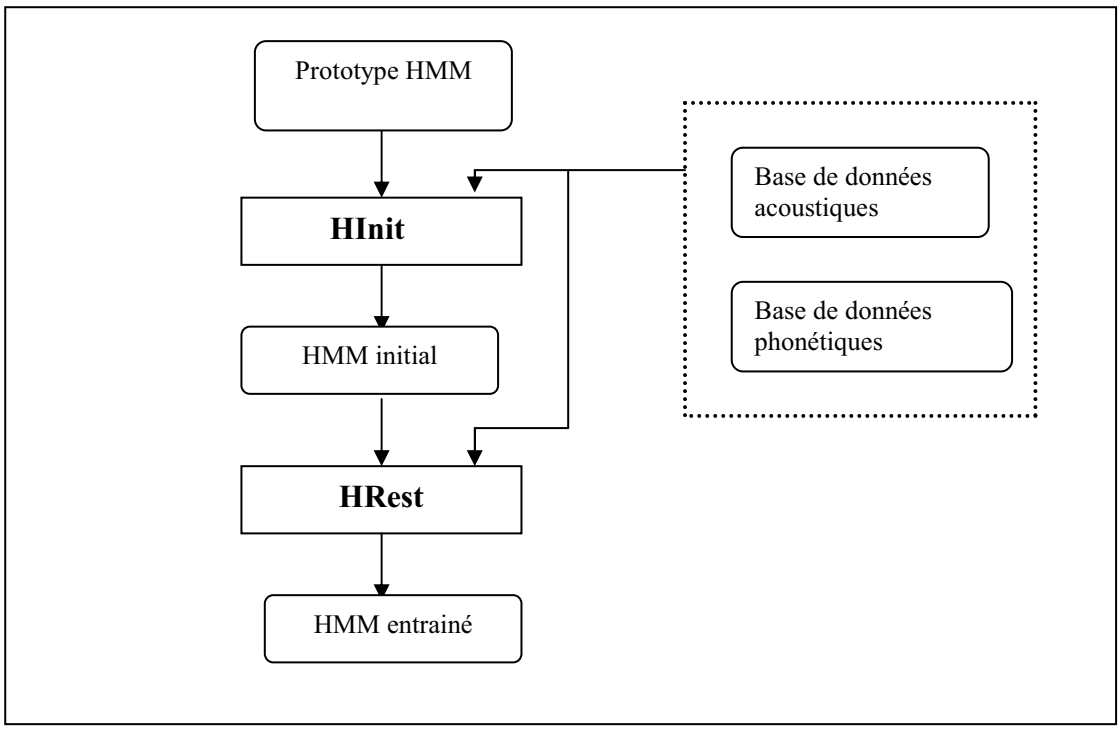


Figure 4.6 : Apprentissage des HMMs avec HTK.

Dans une phase suivante, il est possible pour le mode de reconnaissance de parole continue d'appliquer plusieurs itérations de l'outil ***HERest***, qui ré-estime simultanément l'ensemble des modèles sur de la parole continue non segmentée.

4.2.1.4 Reconnaissance

Le module de décodage de la parole continue, ***HVite***, utilise l'algorithme de Viterbi pour trouver la séquence d'états la plus probable correspondant aux paramètres observés dans un modèle composite, et en déduire les unités acoustiques correspondantes. Le modèle composite autorise la succession des modèles acoustiques en fonction d'un réseau syntaxe choisi par le concepteur du système. La syntaxe de ***HVite*** tient compte aussi d'un modèle de langage, le plus souvent de type bi-grammes, estimé sur les étiquettes des phrases d'apprentissage par l'outil ***HLStats***.

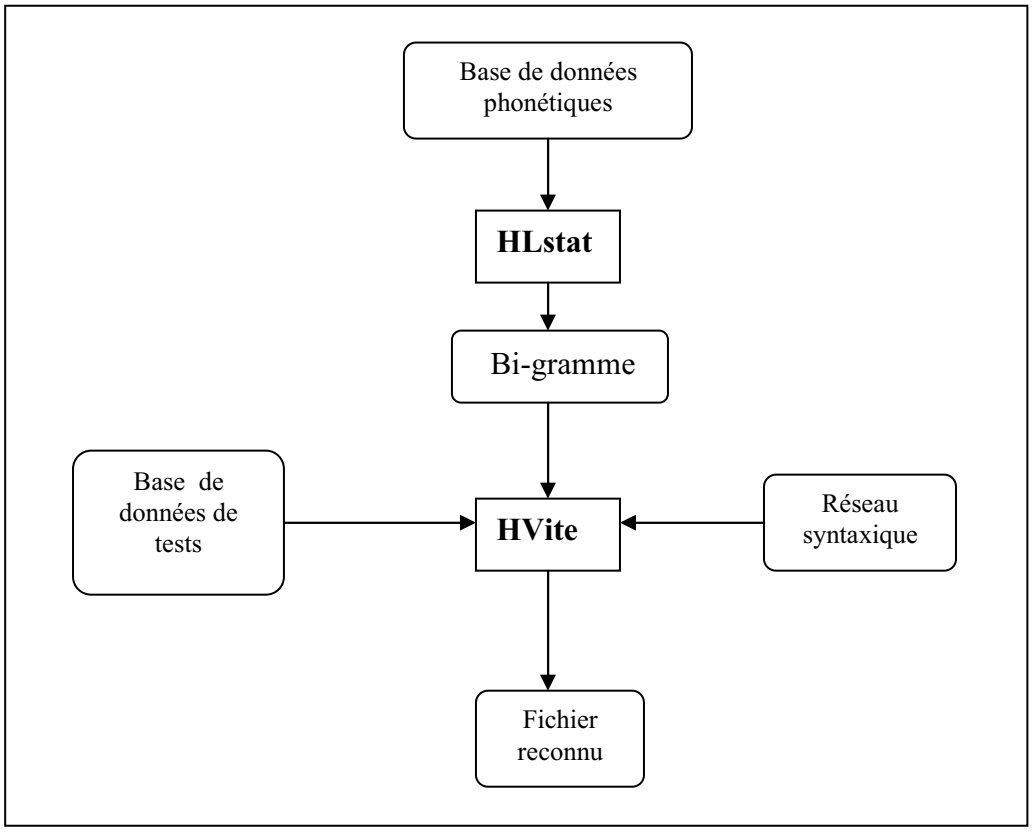


Figure 4.7: la reconnaissance par HTK.

4.2.1.5- Évaluation des résultats

Le résultat du décodage est comparé aux étiquettes de référence par un alignement dynamique réalisé par l'outil ***HResults***, afin de compter les étiquettes identifiées, omises, substituées par une autre, et insérées, et de calculer le taux de reconnaissance.
Les performances des systèmes RAP sont évaluées par le pourcentage d'identification (**Correct** dans HTK) et le pourcentage de reconnaissance (**Corr** dans HTK).

Le pourcentage d'identification correspond à l'équation suivante :

$$\%\ \text{Correct} = \frac{N - O - S}{N} * 100 \qquad (4.2)$$

Le pourcentage de reconnaissance correspond à l'équation suivante :

$$\% \text{ Corr} = \frac{N - O - S - I}{N} \tag{4.3}$$

Avec :
N : le nombre total d'unités.
O : le nombre d'omissions (le nombre d'unités non détectés).
S : le nombre de substitutions (le nombre d'unités pour lesquels le système à commis une erreur).
I : le nombre d'insertions (le nombre d'unités admettent comme reconnus alors qu'aucun mot n'a été prononcé)

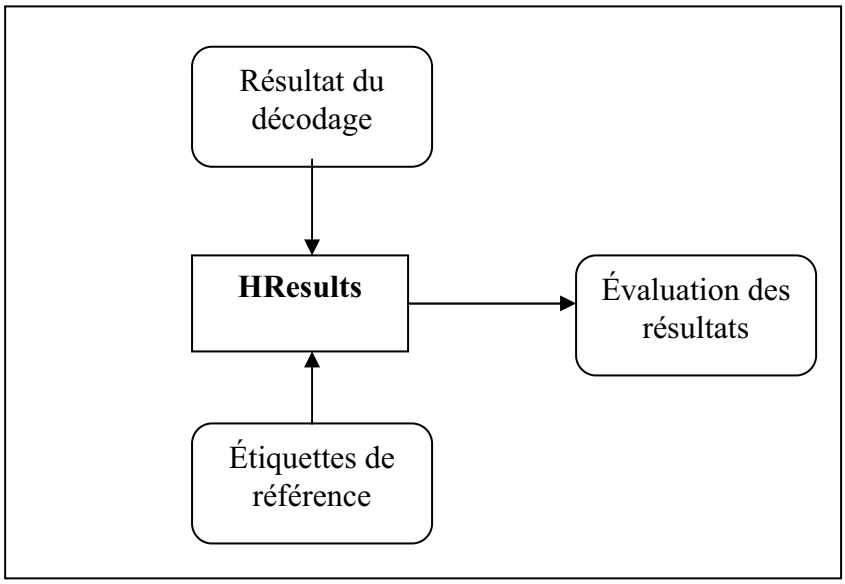


Figure 4.8: Evaluation des résultats

4.2.2 Les bases de données utilisées

Nous avons utilisés deux types de bases de données, une pour la reconnaissance des mots isolés et une autre pour la reconnaissance de la parole continue.

4.2.2.1 La base de données ARADIGIT

Cette base a été conçue au laboratoire LCPTS de la faculté d'Electronique et d'Informatique de l'USTHB [Amro-07]. Elle est constituée de prononciations isolées des 10 chiffres de la langue arabe de 0 jusqu'à 9. La base d'apprentissage est constituée de 1800 fichiers, prononcés par 60 locuteurs des deux sexes qui répètent le même chiffre 3 fois. Alors que la base de test est constituée de 1000 fichiers, prononcée par 50 locuteurs des deux sexes qui répètent le même chiffre 2 fois.
La base de données a été enregistrée par des locuteurs algériens âgés entre 18 et 50 ans dans un environnement calme avec un niveau de bruit ambiant inferieur à 35 dB, sous le format WAV, avec une fréquence d'échantillonnage de 16 KHz.

4.2.2.2 La base de données TIMIT

Le choix de la base de données TIMIT est justifié par plusieurs raisons. Tout d'abord, cette base a été constituée pour illustrer au mieux la variabilité acoustique de l'anglais américain, et

elle est fournie avec une segmentation phonétique de référence qui simplifie l'apprentissage initial des modèles phonétiques. De plus, TIMIT peut être considérée comme une base de données de référence. Sa large diffusion dans la communauté internationale permet une évaluation objective des performances des systèmes développés.

1) Description de la base TIMIT

La base TIMIT est une base de données acoustique et phonétique dédiée à la reconnaissance de la parole en mode independent du locuteur [Sene-88]. Les enregistrements de locuteurs sont répartis en 8 "dialectes régionaux ("dr1" à "dr8") et prononçant chacun 10 phrases. Ces phrases peuvent être classées en 3 types :
- SA : phrases prononcées par tous les locuteurs, servant à illustrer les variations régionales (2 phrases par locuteur).
- SI : phrases choisies pour maximiser les contextes acoustiques, chaque phrase n'est prononcée qu'une seule fois (3 phrases par locuteur).
- SX : phrases phonétiquement équilibrées (5 phrases par locuteur)

Pour chaque phrase, nous disposons de :
- le signal échantillonné à16 kHz sur 16 bits (ficher .adc).
- le texte en anglais (fichier .txt).
- la segmentation phonétique en 62 classes (fichier .phn).

Pour des raisons de disponibilité, notre base de données TIMIT est limitée à :
- Une base d'apprentissage de 3800 phrases.
- Une base de test de 200 phrases.

2) l'étiquetage de la base de données TIMIT

L'étiquetage d'origine en 62 classes est généralement jugé trop détaillé pour l'apprentissage des modèles phonétiques [Barr-96], et une réduction du nombre de classes phonétiques par regroupement de phonèmes est réalisée en 48 classes phonétiques par :
- La fusion de phonèmes (**m/em, n/nx, ng/eng, ax/ax-h, er/axr, ux/uw, hh/hv, jh/j**).
- L'insertion de nouvelles étiquettes : **sil** pour regrouper les silences, **h#/pau**, **cl** pour les occlusions sourdes **pcl/tcl/kcl**, **vcl** pour les occlusions voisées **bcl/dcl/gcl**
- La suppression de l'étiquette **q** qui ne correspond pas toujours à une occlusive.

Lors du calcul des taux de décodage, certaines confusions sont autorisées entre classes (sh/zh, n/en, l/el, ih/ix, aa/ao, ax/ah, sil/epi/cl/vcl), conduisant finalement à des résultats sur 39 classes. Nous avons choisi ces regroupements à des fins de comparaison, afin de comparer nos résultats à ceux déjà publiés.

Nous présentons dans le tableau 4.2 des statistiques sur les 48 phonèmes utilisés, en spécifiant pour chaque phonème le nombre de représentants dans l'ensemble d'apprentissage ainsi que la durée moyenne des segments en millisecondes. L'obtention de ces statistiques est possible grâce à l'outil ***HLstats***.

phonèmes	Nombre d'occurrences	Durée moyenne	phonèmes	Nombre d'occurrences	Durée moyenne
aa	2578	124.4	iy	6479	88.3
ae	3469	149.7	jh	1029	60.0
ah	1959	91.0	k	4276	51.3
ao	2713	124.0	l	5239	60.5
aw	564	163.3	m	3596	62.0
ax	3661	48.7	n	7019	50.8
ay	2029	146.2	ng	1090	63.0
b	1973	18.0	ow	1898	128.1
ch	691	87.2	oy	582	163.3
cl	13136	59.0	p	2328	44.7
d	3167	24.1	r	5811	61.7
dh	2489	38.1	s	6394	112.8
dx	2281	28.3	sh	1899	116.3
eh	3267	91.5	sil	8986	187.7
el	869	90.3	t	3821	49.2
en	643	79.0	th	625	90.8
epi	1149	45.7	uh	464	77.0
er	4862	96.3	ux	2041	106.9
ey	2049	127.5	v	1741	60.5
f	2034	101.5	vcl	7763	53.5
g	1845	31.5	w	2824	68.9
hh	1798	67.3	y	1466	68.2
ih	4163	79.3	z	3361	88.0
ix	7266	52.0	zh	142	82.8

Tableau 4.2 : Statistiques sur le nombre de représentants et la durée moyenne des 48 phonèmes utilisés.

4.3 Construction du système standard

Nous commençons tout d'abord par construire un système standard de référence avec une représentation acoustique du signal par des coefficients cepstraux (MFCCs) dits paramètres standards. Nos expériences ont été réalisées à la fois avec un système de reconnaissance de mots isolés sur le corpus des chiffres arabes ARADIGIT et un système de reconnaissance de parole continue sur la base de donnée TIMIT. Dans ce dernier cas nous avons fait l'évaluation par deux approches :

- Le décodage acoustique-phonétique (DAP) **:** qui consiste généralement à transcrire le signal parole en une suite d'unités élémentaires (phonèmes). C'est ce système que nous avons pris comme système de référence dans le cas de la parole continue.

- La reconnaissance en mots : qui consiste à transcrire le signal parole en mots, en tenant compte d'informations sur le lexique et la syntaxe du langage utilisé dans l'application. Les taux de reconnaissance obtenus par ce système vont être donnés en annexe A.

L'évaluation des modèles acoustiques par le DAP permet une mise au point rapide du système de reconnaissance, et surtout une meilleure mise en évidence des améliorations apportées au système de référence. Nous auront ainsi l'avantage d'observer la performance du système au niveau le plus bas de reconnaissance (sous le niveau de modèles acoustique), sans aucune interférence de sources lexicales qui peuvent couvrir les résultats réels de décodage acoustique par leurs post-traitements appliqués.

4.3.1 Caractéristiques du système standard

Le schéma synoptique du système standard donné en figure 1.14 a été développé avec les caractéristiques suivantes :

- **Paramétrisation du signal acoustique**

Le signal de parole est paramétré en utilisant les coefficients MFCCs. Ces paramètres sont extraits toutes les 10 ms, sur des fenêtres de 25 ms. La fenêtre de pondération utilisée est la fenêtre de Hamming. Douze (12) coefficients MFCC sont calculés pour chaque trame à partir d'un banc de 24 filtres répartis dans l'échelle fréquentielle Mel. Des coefficients différentiels du premier (△) et du second ordre (△△) sont ensuite calculés pour former un vecteur de dimension 36 (12 MFCC + △12 MFCC+ △△12MFCC). D'après la littérature [Barr-96] [Igou-98], le choix de cette combinaison semble être la plus satisfaisante pour représenter le signal de parole dans le cadre de la RAP.

- **Modèles acoustiques**

Les modèles acoustiques dépendent du mode de reconnaissance réalisée : mode de reconnaissance mots isolés ou parole continue. La reconnaissance de mots isolés utilise 10 modèles HMMs, chacun des 10 chiffres du vocabulaire est modélisé par un modèle acoustique séparé (modèle HMM). La reconnaissance de la parole continue utilise 48 modèles correspondants aux 48 phonèmes indépendants du contexte adoptés pour décrire la langue anglaise.

- **Topologie des modèles**

Tous les modèles acoustiques (HMMs) ont une topologie de type Bakis à 3 états émetteurs (figure 4.10), choix très utilisé en RAP.

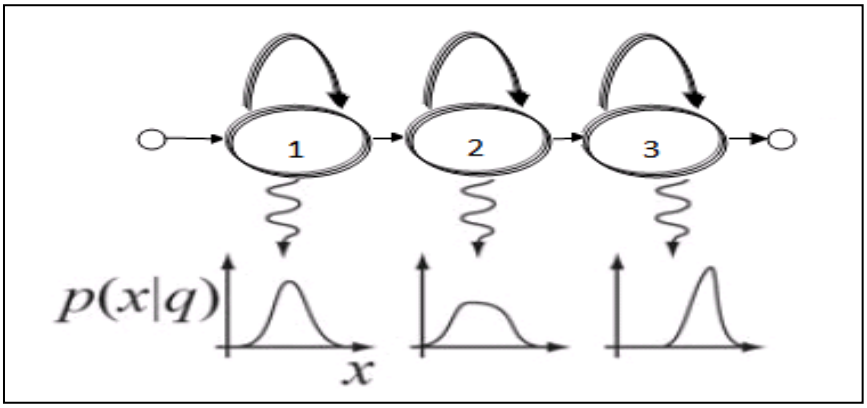


Figure 4.9 : Modèle du Bkis à 3 états émetteurs.

L'état initial et l'état final ont la particularité de ne pas émettre d'observation, mais de servir uniquement à la connexion des modèles en parole continue (nous avons conservé la même topologie dans le cas des mots isolés pour des raisons d'homogénéité de formalisme des modèles prototypes).

- **Probabilité d'émission**

Pour chaque état émetteur du modèle, la probabilité d'émission est modélisée par une combinaison linéaire de gaussiennes à matrice de covariance diagonale.
L'augmentation du nombre de gaussiennes permet une amélioration de la modélisation des phénomènes acoustiques. Pour des raisons de stabilité des procédures HTK, il est préférable de faire un apprentissage multi-gaussiennes basé sur une augmentation progressive du nombre de gaussiennes. Son principe est le suivant : tout d'abord, des modèles à une gaussienne par état sont appris, puis, le nombre de gaussiennes par état est augmenté progressivement, par clonage en utilisant l'outil ***HHed***. Enfin, les modèles sont ré-estimés.

- **Modèle de langage**

Le modèle de langage est caractéristique du mode de reconnaissance de type parole continue. Le modèle de langage utilisé dans notre travail est une grammaire de type bi-grammes phonétique.

4.3.2 Validation du système de référence

Pour valider notre système de référence pour les différents modes de reconnaissance (mots isolés et parole continue) nous avons étudié sa performance en se basant sur des paramètres pour lesquels il existe déjà des résultats équivalents dans la littérature. De cette manière nous pouvons valider notre système de référence, étant donné qu'il constitue le noyau de notre travail, d'une manière objective**.**

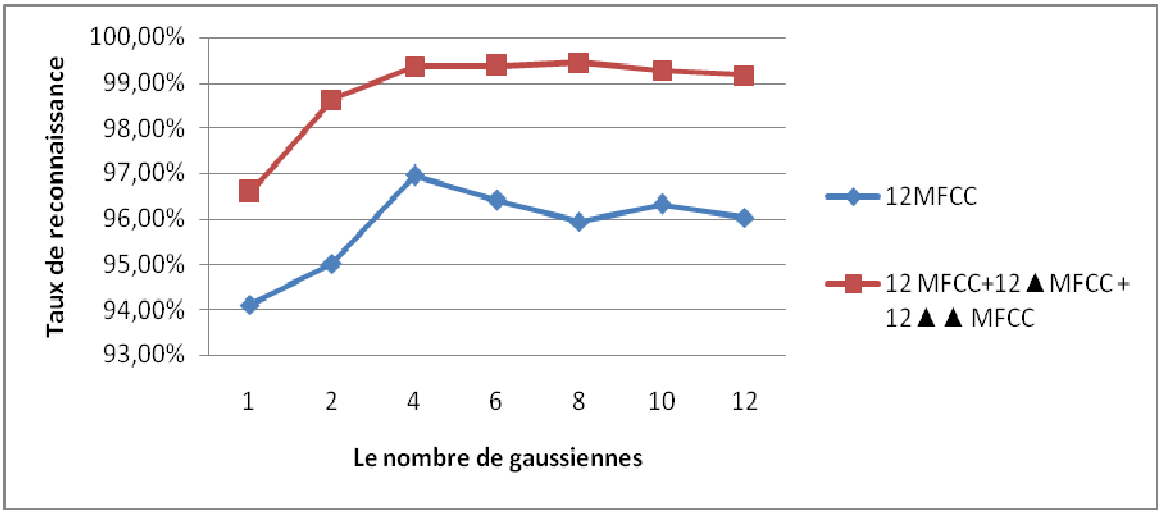
En particulier, nous testons :
- L'apport du nombre de mélange de gaussiennes par état.
- L'apport des coefficients différentiels du premier et du second ordre par rapport aux coefficients statiques seuls.
- Dans le cas de la parole continue : l'utilité d'un modèle de langage bi-gramme.

4.3.2.1 Aapport des coefficients différentiels en fonction du nombre de gaussiennes

Les tableaux 4.2, 4.3 et les figures 4.10, 4.11 présentent les résultats de reconnaissance qui montrent l'apport des coefficients différentiels du premier et du second ordre par rapport aux coefficients statiques seuls et cela en fonction du nombre de gaussiennes :

Nombres de gaussiennes	Taux de reconnaissance (en %)	
	Vecteur acoustique : 12 MFCC	Vecteur acoustique : 12 MFCC+12△MFCC + 12△△ MFCC
1	94.10%	96.59%
2	95.02%	98.62%
4	**96.96%**	99.35%
6	96.40%	99.37%
8	95.94%	**99.45%**
10	96.31%	99.26%
12	96.03%	99.17%

Tableau 4.3: Taux de reconnaissance comparatifs coefficients statiques/coefficients dynamiques en fonction du nombre de gaussiennes pour les mots isolés.



Figures 4.10: Taux de reconnaissance comparatifs coefficients statiques/coefficients dynamiques en fonction du nombre de gaussiennes pour la reconnaissance de mots isolés.

Nombres de gaussiennes	Vecteur acoustique 12 MFCC		Vecteur acoustique : 12 MFCC+12▲MFCC+ 12▲▲ MFCC	
	Phonèmes identifiés	Phonèmes reconnus	Phonèmes identifiés	Phonèmes reconnus
1	48.14%	44.14%	58,73%	51,55%
2	50.99%	46.86%	61,65%	54,37%
4	53.99%	49.71%	64,87%	57,87%
8	55.27%	50.80%	66,14%	58,84%
10	55.72%	51.30%	66,74%	59,51%
12	**56.71%**	**52.36%**	**67,33%**	**60,39%**

Tableau 4.4 : Taux de reconnaissance comparatifs coefficients statiques/coefficients dynamiques en fonction du nombre de gaussiennes pour le DAP.

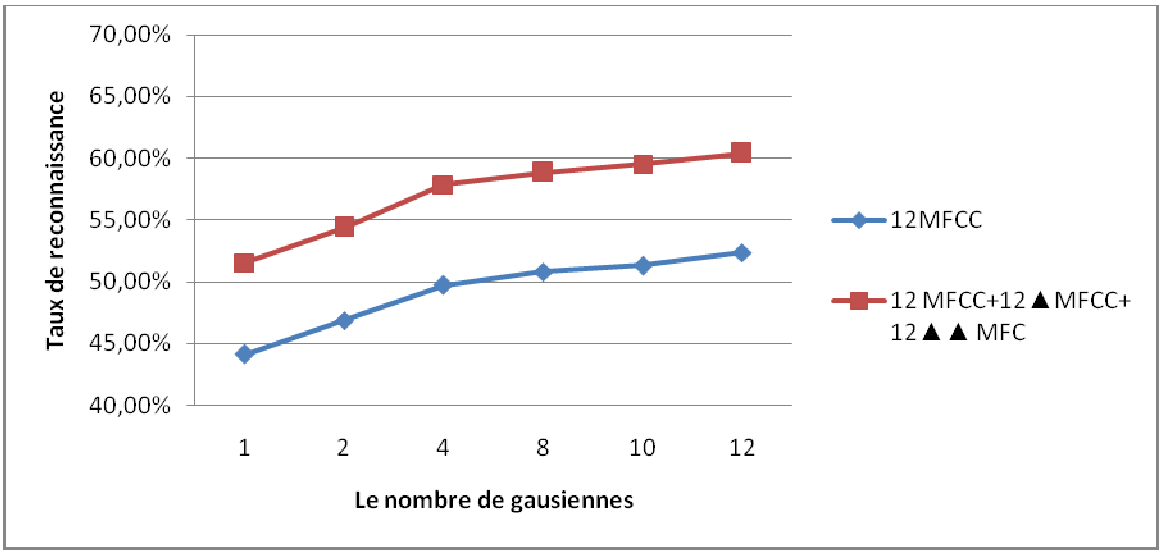


Figure 4.11: L'apport des coefficients différentiels aux taux de reconnaissance d'un système DAP en fonction du nombre de gaussiennes.

Les résultats obtenus montrent que l'accroissement du nombre de gaussiennes permet une meilleure modélisation de l'espace acoustique. Dans le cas d'un système de reconnaissance de mots isolés nous avons atteint le nombre de gaussiennes optimal qui permet une stabilité des taux de reconnaissances (nombre de gaussienne égale à 8). Alors que, dans le cas de la reconnaissance de la parole continue où les temps de calcul du système croient rapidement en fonction du nombre de paramètres, c'est-à-dire le nombre de gaussiennes des HMM utilisés. Il est donc nécessaire de trouver un compromis entre le taux de reconnaissance et le nombre de paramètres. Nous remarquons qu'au-delà de 10 gaussiennes, les performances du système ne sont plus significativement améliorées (59.51 vers 60,93), tandis que la convergence de l'apprentissage est plus longue à atteindre (exemple : pour un ordinateur Pentium 4 et une vitesse CPU 2.8 HZ : le passage de 10 gaussiennes à 12 gaussiennes nécessite un temps du calcul estimé de 2970 secondes).

Toujours dans les mêmes figures (4.11 et 4.12), les résultats montrent que l'apport des coefficients différentiels du premier et du second ordre est majeur surtout pour le cas d'un

système DAP (reconnaissance continue), où nous observons une amélioration de l'ordre de 10 à 12% sur le taux de reconnaissance.

4.3.2.3 L'utilité d'une grammaire de type bi-grammes

À partir des paramètres optimaux obtenus grâce aux expériences précédentes (vecteurs acoustiques composés de 12 MFCC+12△MFCC + 12△△ MFCC, et une modélisation de 12 gaussiennes par état), Nous présentons dans le tableau 4.5 les résultats qui montrent l'utilité d'une grammaire de type bi-grammes phonétique (estimée sur le corpus d'apprentissage) par rapport à un système sans bi-gramme où tous les modèles phonétiques sont équiprobables. Le gain du taux de reconnaissance apporté par le bi-gramme est important, de l'ordre de 6 %.

Grammaire de type Bi-gramme	Taux de reconnaissance
avec	**60,39%**
sans	54,01%

Tableau 4.5 : Influence d'un bi-gramme phonétique sur le taux de reconnaissance.

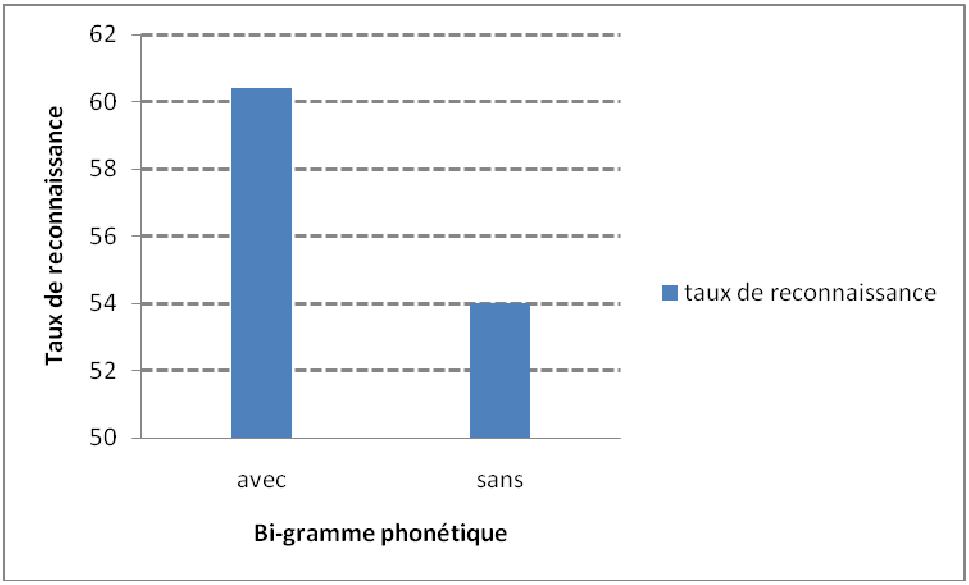


Figure 4.12: Apport de la grammaire bi-grammes phonétiques sur le taux de reconnaissance.

4.4- Fusion des données auxiliaires dans un système de RAP

Comme déjà mentionné, l'objectif de ce travail est de mettre au point un système de reconnaissance fondé sur les modèles de Markov cachés en incorporant des sources d'informations auxiliaires. Les sources d'informations auxiliaires proposées dans ce travail sont : le pitch, les fréquences des trois premiers formants et l'énergie.

Nous avons étudié la robustesse des différentes stratégies de la fusion (section 3.4) dans un environnement calme et dans un environnement bruité**.** Pour cela nous avons dégradé les deux bases de données acoustiques utilisées dans ce travail par différents types de bruits, qui sont

respectivement le bruit d'usine, le bruit d'avion, le bruit de la foule et le bruit de la nature. Ces bruits ont été extrais à partir de la base de données NOISEX [Varg-92] et dont les spectrogrammes sont visualisés par la figure 4.15.

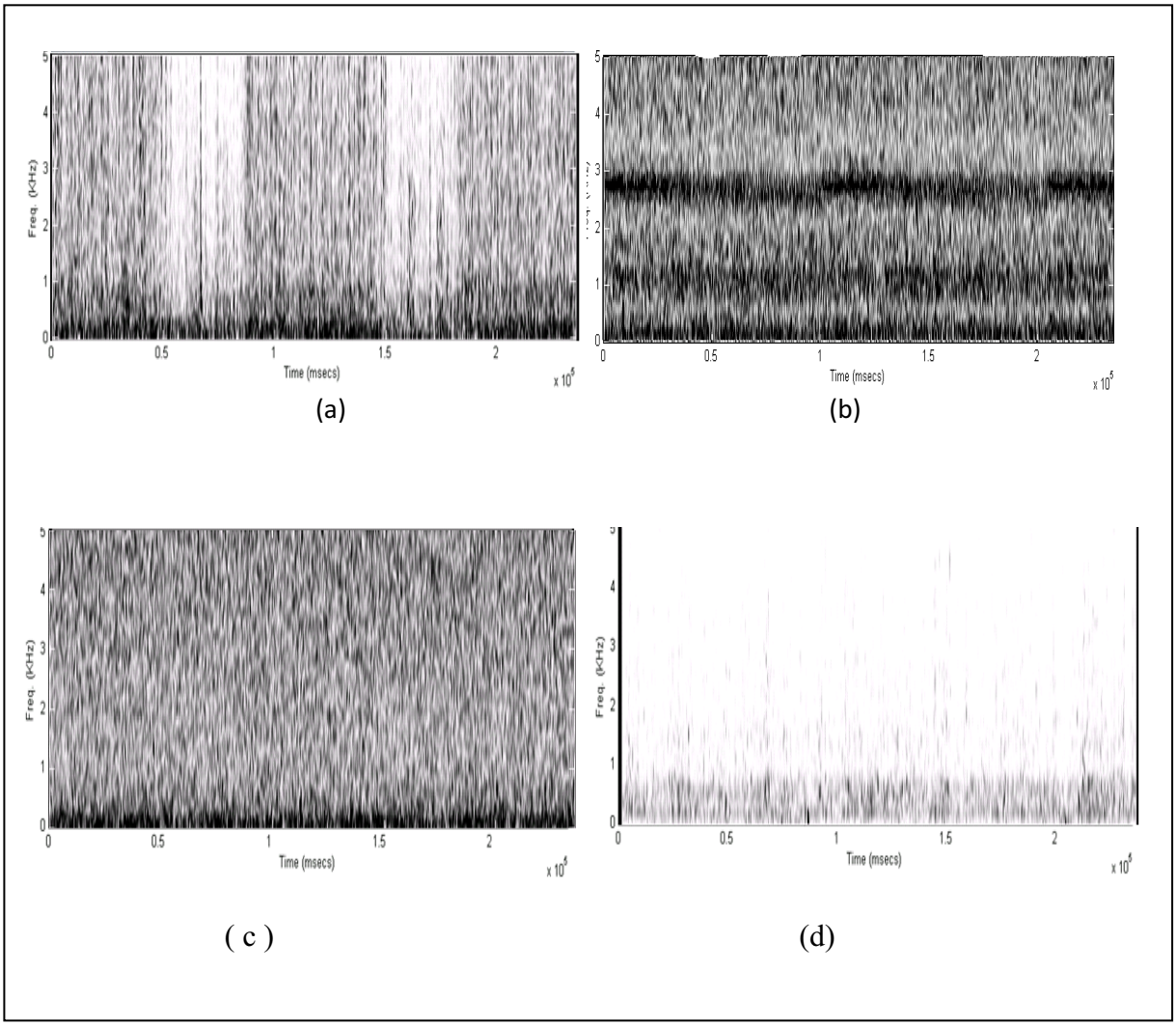


Figure 4.13: Spectrogrammes des quatre bruits utilisés : (a) le bruit d'usine ; (b) le bruit d'un avion, (c) le bruit de la nature et (d) le bruit de la foule.

L'entraînement des modèles acoustiques a été réalisé avec les bases de données non bruitées, le bruit a été seulement ajouté pour les corpus de test. Ces derniers ont été dégradés à différents niveaux RSB (Rapport Signal/Bruit) 15 dB, 10 dB, 5 dB et 0 dB. Le niveau RSB d'un signal parole est estimé par la formule 4.4 suivante:

$$RSB = 20 \log_{10}\left(\frac{A_{signal}}{A_{bruit}}\right) \tag{4.4}$$

Tel que :

A_{signal} et A_{bruit} représente respectivement la racine carré d'amplitude du signal et du bruit.

4.4.1 La stratégie ID

Comme nous l'avons déjà présenté à la section 3.4.1, dans la stratégie ID les paramètres standards et auxiliaires sont concaténés dans le même vecteur acoustique.

Le vecteur acoustique à l'entrée de système de reconnaissance est un vecteur composé des 12 MFCCs du système de référence, plus les 5 paramètres auxiliaires : le pitch, l'énergie et les trois premiers formants, ainsi que leurs dérivées temporelles première et seconde. Ce qui forme en total un vecteur de dimension 51.
Les taux de reconnaissance du système ID (le système qui utilise la stratégie ID pour fusionner les paramètres standards et auxiliaires) comparés à ceux du système de référence dans les différents environnements sont donnés par les tableaux 4.6 et 4.7 respectivement pour le mode isolé et pour la parole continue.

		Système de référence	Système ID
calme		99,45%	98,52%
Bruit d'usine	15 dB	79,61%	84,04%
	10 dB	58,03%	64,58%
	5 dB	33,12%	39,58%
Bruit de la nature	15 dB	77,95%	81,64%
	10 dB	57,10%	62,20%
	5 dB	30,63%	33,03%
Bruit de la foule	15 dB	95,11%	74,35%
	10 dB	61,25%	63,25%
	5 dB	38,93%	45,57%

Tableau 4.6: Taux comparatifs système ID/système de référence en mode parole isolée.

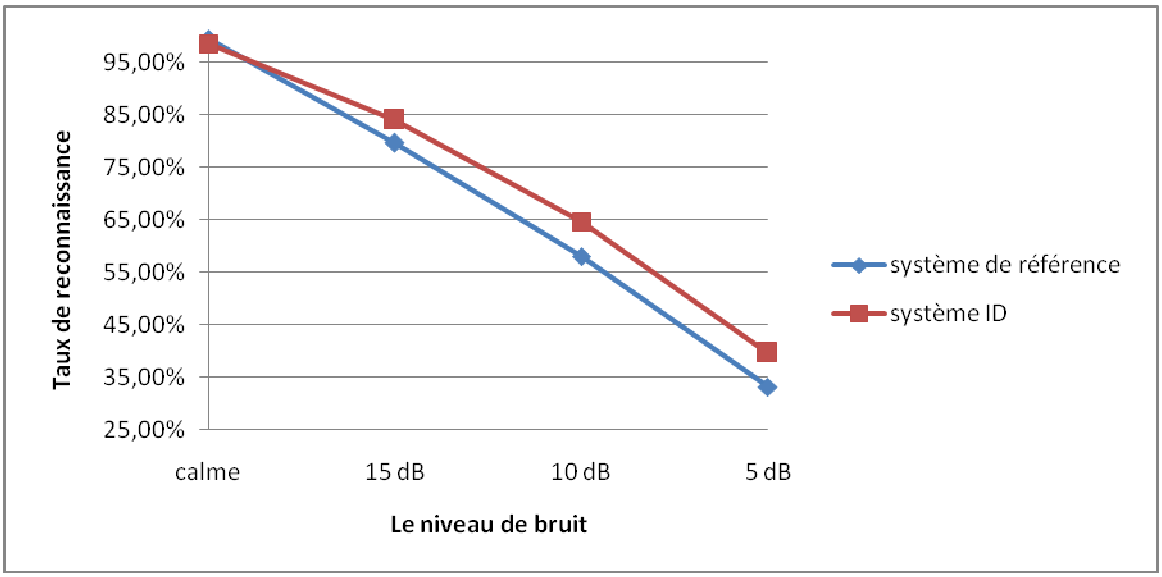


Figure 4.14: Taux comparatifs système ID/système de référence en mode parole isolée dans un environnement bruité avec le bruit d'usine.

		Système de référence	Système ID
calme		60,39%	57,54%
Bruit d'usine	15 dB	41,79%	36,96%
	10 dB	32,79%	30,48%
	5 dB	23,79%	24,02%
	0 dB	14,94%	18,42%
Bruit d'avion	15 dB	39,46%	34,89%
	10 dB	30,27%	29,05%
	5 dB	19,88%	23,86%
	0 dB	10,46%	19,11%
Bruit de la nature	15 dB	38,60%	34,32%
	10 dB	29,72%	27,94%
	5 dB	19,10%	22,20%
	0 dB	9,50%	16,80%

Tableau 4.7 : Taux comparatifs système ID/système de référence en mode parole continue.

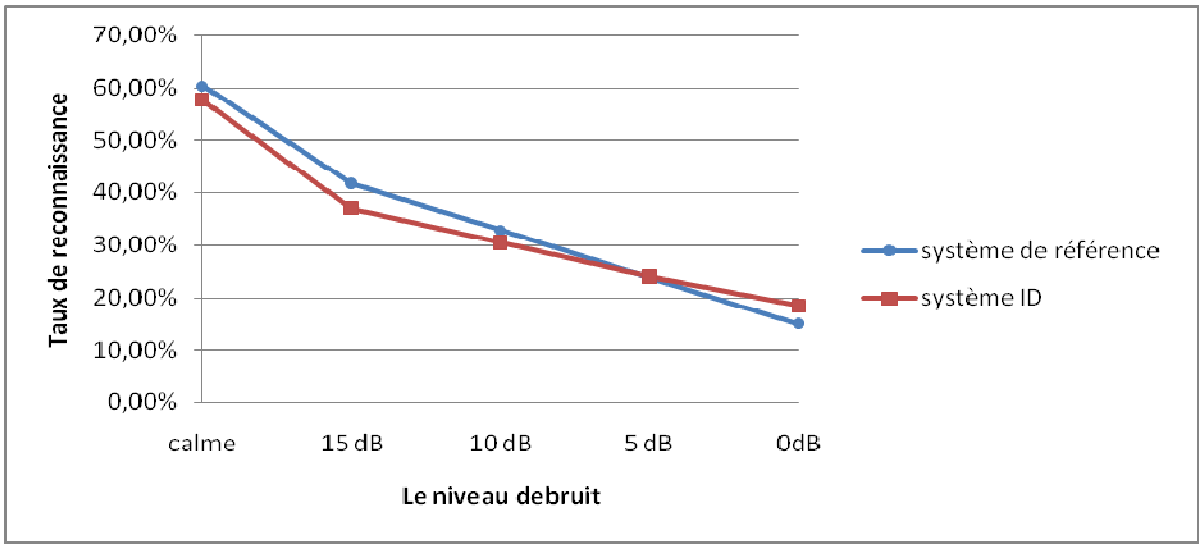


Figure 4.15: Taux comparatifs système ID/système de référence en mode parole continue dans un environnement bruité avec le bruit d'usine.

L'amélioration des taux, par rapport au système de référence, n'est obtenue que dans le cas d'environnement bruité (figure 4.14, figure 4.15). Ceci sera justifié dans la section d'interprétation de résultats (section 4.4.3).

4.4.2 La stratégie IS

Nous rappelons que le principe de la stratégie IS (section 3.4.2) consiste à modéliser chaque type de paramètres (standard et auxiliaire) par un sous système de reconnaissance séparé. Les résultats (scores probabilistes) en sortie de chacun de ces sous systèmes de reconnaissance sont fusionnés dans un module de décision qui donne le résultat final.

Les combinaisons des vecteurs acoustiques à l'entrée des deux sous systèmes est comme suite **:**

- Les vecteurs acoustiques du sous système standards sont composés de 36 coefficients : 12 coefficients MFCCs plus leurs dérivées première et seconde.

- Les vecteurs acoustiques du sous système auxiliaire sont composés de 15 coefficients : le pitch, l'énergie, les fréquences des trois premiers formants, ainsi que leurs dérivées première et seconde.

Le module de décision a été réalisé dans un cadre neuronal, au moyen d'un réseau de neurones de type PMC (Perceptrons Multi-Couches) qui fusionne les scores de probabilité à la sorties des deux sous systèmes : standards et auxiliaires. Le réseau de neurone a été implémenté avec le logiciel Praat [Boer-08]. Il est composé de :

- Une couche d'entrée : composée de 20 neurones (20 scores de probabilités).
- **Deux couches cachées :** chacune composée de 15 neurones**.**
- **Une couche de sortie** : composée de 10 neurones (10 chiffres à reconnaître).

Le réseau de neurones a été d'abord entraîné par les 2N meilleurs scores associés à chaque fichier acoustique de la base d'entraînement. Ce qui veut dire que nous avons été obligé de tester la base d'entraînement pour extraire les 2N meilleurs scores.

Pour des raisons de complexité de la recherche des N meilleures solutions dans le graphe lexico-syntaxique fournit par l'outil HTK dans le cas de la parole continue, nous nous limitons dans ce travail à l'implémentation de cette stratégie de fusion seulement pour le cas de la reconnaissance de mots isolés.

Les taux de reconnaissance du système IS (le système qui utilise la stratégie IS pour fusionner les paramètres standards et auxiliaires) pour les différents environnements (calme et bruité) sont donnés par le tableau 4.8 suivant:

		Système de référence	Système IS
calme		99,45%	99,46%
Bruit d'usine	15 dB	79,61%	79,69%
	10 dB	58,03%	58,15%
	5 dB	33,12%	33,21%
Bruit de la nature	15 dB	77,95%	78,02%
	10 dB	57,10%	57,18%
	5 dB	30,63%	30,71%
Bruit de la foule	15 dB	95,11%	95,19%
	10 dB	61,25%	61,29%
	5 dB	38,93%	38,97%

Tableau 4.8: Taux comparatifs système IS/système de référence en mode parole isolée.

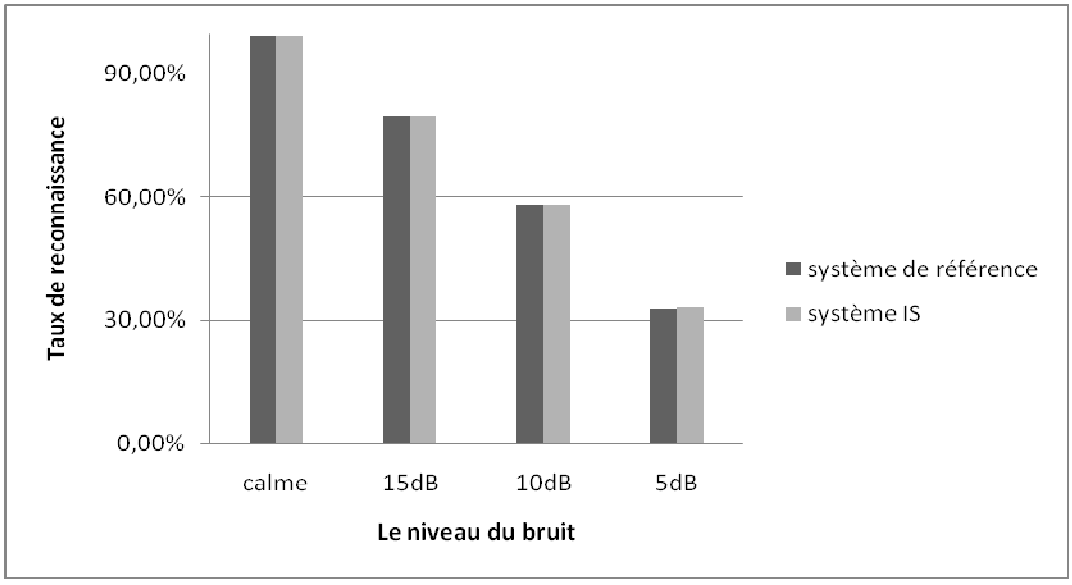


Figure 4.16: Taux comparatifs système IS/système de référence en mode parole isolée dans un environnement bruité avec le bruit d'usine.

4.4.3 Interprétation des résultats

L'interprétation des résultats est basée sur la comparaison des taux de reconnaissances des trois systèmes implémentés : le système de référence, le système ID et le système IS. Nous exploitons également les résultats obtenus avec le sous système auxiliaire pour mieux interpréter ces résultats.
Et pour mieux interpréter les taux obtenus nous exploitons les expériences réalisées au cours de la construction du système IS pour rajouter une entrée dans le tableau de résultats qui correspond au système auxiliaire. C'est-à-dire le système de RAP où l'entrée acoustique correspond aux paramètres auxiliaires.

Comme les deux stratégies sont seulement implémentées dans le cas de la reconnaissance de mots isolés, le tableau comparatif des différents systèmes (tableau 4.9) va présenter les résultats obtenus pour ce dernier mode de mots isolés.

		Système de référence	Sous système auxiliaire	Système ID	Système IS
calme		99,45%	93.27%	98,52%	99,46%
Bruit d'usine	15 dB	79,61%	73,90%	84,04%	79,69%
	10 dB	58,0 3%	52,12%	64,58%	58,15%
	5 dB	33,12%	26,01%	39,58%	33,21%
Bruit de la nature	15 dB	77,95%	68,54%	81,64%	78,02%
	10 dB	57,10%	50,96%	60,20%	57,18%
	5 dB	30,63%	31,09%	33,03%	30,71%
Bruit de la foule	15 dB	95,11%	60,52%	74,33%	95,19%
	10 dB	61,25%	58,67%	63,25%	61,29%
	5 dB	38,93%	32,20%	45,57%	38,97%

Tableau 4.9 : Taux comparatifs des différents systèmes mis en œuvre.

Il apparaît clairement d'après le tableau 4.9 qu'en l'absence du bruit, les taux de reconnaissance obtenus par le système ID sont légèrement faibles par rapport à ceux obtenus par le système de base (99.45% vs 98.52%). Ceci peut s'expliquer par le fait que les paramètres auxiliaires additifs perturbent les paramètres standards plus fiables (le taux de reconnaissance du sous système auxiliaires n'est que de 93.27%). Cette perturbation se situe non seulement au niveau décisionnel, mais également au niveau de l'entraînement. Une autre raison qui peut expliquer cette dégradation des taux de reconnaissance est le fait que la modélisation des paramètres du système ID par une distribution multi gaussiennes peut être non adéquate pour modéliser le vecteur allongé par les paramètres auxiliaires. De plus, la contrainte des matrices de covariances diagonales est très forte pour les paramètres auxiliaires.

Cependant, en présence de bruit les résultats montrent une nette amélioration des taux de reconnaissance du système ID par rapport au système de référence (par exemple : avec le bruit d'usine au niveau 5dB : 33,12 % vs. 39,58%, c à d une amélioration de l'ordre de 7%). Nous constatons également que les performances du système ID s'améliorent lorsque le RSB diminue. Ceci montre la sensibilité réduite des paramètres auxiliaires au bruit comparativement aux paramètres MFCC.

Les performances du système IS sont toujours meilleures que celles obtenues avec le système de référence (dans les conditions calmes et bruitées). Mais cette amélioration est relativement moins importante (de l'ordre de 0.1%). Cela peut s'expliquer à notre avis par le fait que la fusion dans le système IS dépend grandement des paramètres standards. C'est-à-dire que nous avons entraîné le module de décision (le réseau de neurones) sur l'hypothèse que le sous système standard est le plus adapté (le plus fiable) que le sous système auxiliaire.

Globalement, le système ID est le système le plus robuste dans l'environnement bruité. Ceci peut s'expliquer par le fait que l'information auxiliaire qui est moins sensible au bruit intervient dés la première étape du processus de reconnaissance (entraînement + test) ce qui n'est pas le cas pour le système IS où l'information auxiliaires n'influe qu'au niveau décisionnel.

4.5 Conclusion

Dans ce chapitre expérimental, nous avons réalisé un système de reconnaissance fondé sur les HMM en incorporant des sources d'informations auxiliaires. Nous avons commencé par la construction d'un système de référence avec la plate-forme logicielle HTK de l'Université de Cambridge. Ce système de référence nous a permit d'évaluer l'apport des paramètres auxiliaires et les différentes stratégies de fusion (ID et IS) que nous avons étudié dans le chapitre précédent. Nous avons évalué, dans les mêmes conditions, les performances des systèmes développés au regard de celles du système de référence. Et d'après les résultats obtenus les informations auxiliaires ont amélioré significativement les taux de reconnaissance dans le milieu bruité, particulièrement dans la fusion de type ID. Ceci encourage l'utilisation de ces paramètres auxiliaires dans les conditions réelles.

Conclusion et perspectives

Conclusion et Perspectives

Dans ce travail, le but était de construire un système de reconnaissance de la parole robuste aux variabilités acoustiques et plus particulièrement celles causées par l'environnement. La performance d'un système de reconnaissance de la parole est étroitement liée à la qualité de la modélisation acoustique des données utilisées. Ce qui nous a motivé à concentrer nos efforts sur l'amélioration de la modélisation acoustique standard utilisée actuellement, en lui fusionnant d'autres types de paramètres dits paramètres auxiliaires.

Nous avons constaté que les performances des systèmes de reconnaissances se dégradent remarquablement dans les conditions réelles (bruitées) et ceci même si nous utilisons une bonne paramétrisation acoustique telle que les coefficients MFCC. Dans ce travail nous avons proposé une nouvelle modélisation acoustique du signal parole basée sur la fusion des paramètres MFCC et des nouveaux paramètres acoustiques dits auxiliaires qui sont le pitch, l'énergie et les fréquences des trois premiers formants. La fusion de ces paramètres avec les paramètres MFCC a été faite de deux manières différentes. Soit par une simple concaténation des deux types de paramètres dans le même vecteur acoustique (la stratégie de fusion directe) ou par l'intégration d'un module de décision qui choisi la meilleure solution parmi les 2N meilleures solutions présentes à la sortie des deux sous systèmes de reconnaissances basé chacun sur un type de paramètres.

Malgré les améliorations remarquables des taux de reconnaissance obtenus dans les environnements bruités, le travail proposé dans ce mémoire reste à notre avis perfectible. Comme perspectives à ce travail nous proposons :

- La dimension du vecteur acoustique (qui contient à la fois les paramètres MFCC et les paramètres auxiliaires) à l'entrée du système ID peut être jugé comme un peu trop grande pour une efficace modélisation probabiliste du vecteur, vu les problèmes posés par la modélisation des vecteurs de grandes dimensions par des lois gaussiennes [Bouv-06]. Ceci nous motive à proposer comme une première voie d'amélioration d'utiliser la technique ACP (Analyse en Composantes Principales) pour réduire la dimension de l'espace des paramètres acoustiques permettant ainsi de donner plus de robustesse à la modélisation de la distribution acoustique par des lois gaussiennes.

- Le choix de la loi gaussienne pour modéliser la distribution acoustique standard à toujours donner les meilleurs résultats dans la littérature. Mais la généralisation de ce type de loi pour modéliser le nouveau flux acoustique standard auxiliaire peut être à notre sens un peu limitatif vu que rien ne garanti que la nouvelle distribution probabiliste de l'espace acoustique sera toujours bien modélisée par des lois gaussiennes. De plus, nous savons que les paramètres MFCC sont bien décorrélés entre eux ce qui a permis de poser la contrainte des matrices de covariance diagonales, ceci n'est pas le cas pour les paramètres auxiliaires. Pour résoudre ce problème nous proposons d'utiliser les réseaux de neurones pour modéliser sans aucune contrainte la distribution probabiliste de l'espace acoustique.

- Les paramètres auxiliaires utilisés dans ce travail sont obtenus à partir d'une implémentation propre écrite en langage de programmation MATLAB. Alors que, les paramètres standards sont extraits depuis la plateforme HTK, et donc, leur calcul est plus fiable. Cet état de fait à limiter la pertinence des paramètres auxiliaires. Pour palier à ce problème nous proposons d'utiliser des outils logiciels pour calculer les

paramètres auxiliaires. Dans ce sens des expériences en utilisant le logiciel Praat [Boer-08] ont était effectuées et sont données en annexe B. Les résultats obtenus ont montré une amélioration plus significative des taux de reconnaissance par rapport à ceux obtenus par le calcul propre sous MATLAB.

- Le formalisme des HMMs ne permet pas de prendre en compte la nature hétérogène des paramètres standards et auxiliaires fusionnés dans le même vecteur acoustique, ceci peut être vu comme un manque d'exploitation de l'information réelle existante. Des nouvelles variantes des HMMs sont apparues sous le mon des réseaux bayésiens. Ces derniers fournissent une très grande souplesse de modélisation. Ils offrent entre autre la possibilité de modéliser le flux acoustique comme un ensemble de variables aléatoires dont on peut conditionner certaines par rapport à d'autres. Les variables peuvent également être supposées comme des processus observables ou cachés.

Bibliographie

[ALL-91] A. Alliche « Conception et réalisation d'un logiciel d'un de traitement de la parole et étude de l'emphase sur la structure formantique des voyelles de l'arabe standard ». Thèse de magister. Institue Electronique d'USTHB. 1991.

[Amro-07] A. Amrouche « Reconnaissance automatique de la parole par les modèles connexionnistes ». Thèse de doctorat, faculté d'électronique et d'informatique, USTHB. 2007.

[Baki-76] R. Bakis « Continuous speech recognition via centisecond acoustic states» 91th. Meeting of the Acoust.Soc. Am., avril 1976.

[Bakk-08] N.Bakkir, « Rconnaissance Automatique de la Parole par fusion audiovisuelle dans un milieu réel» Thèse Magister, faculté d'Electronique et d'Informatique, USTHB. Juillet 2008.

[Barr-96] C. Barras «Reconnaissance de la parole continue : adaptation au locuteur et contrôle temporel dans les modèles de Markov cachés » Thèse de doctorat en informatique, Université de Paris VI. Mai 1996.

[Baud-93] G. Baudoin, P. Jardin et d'autres « Comparison de techniques paramétrisation spectrale pour la reconnaissance vocale en milieu bruité » quatorzième colloque gretsi. Septembre 1993.

[Baum-72] L. Baum « An inequality and association maximization technique in statistical estimation for probabilistic functions of Markov processes » Inequality, vol. 3, 1972.

[Baum-68] L.E. Baum et G.R. Sell « Growth functions for transformations on manifolds » Pac.J.Math; vol 27, no.2 pp.211-277, 1968.

[Bell-57] R. Bellman, «Dynamic programming » Princeton University Press, 1957.

[Bour-90] H. Bourlard & C.J. Wellekens, « Links between Markov models and multilayer perceptrons» IEEE Trans. on Pattern Anal. and Machine Intell., vol. 12, no. 12, pp. 1-12, 1990.

[Boer-08] P. Boersma et Weenink, D. «Praat: doing phonetics by computer» Depuis le site web: http://www.praat.org/ . 2008.

[Boit-00] :R. Boite et H.Bourlard et autres « Traitement automatique de la parole » Collection électricité. Lausanne : Presses Polytechniques et Universitaires Romandes, 2000.

[Buni-97] L. Buniet « Traitement automatique de la parole en milieu bruité : étude de modèles connexionnistes statiques et dynamiques» Thèse doctorat. L'Université Henri Poincaré - Nancy 1. février 1997.

[Bouv-06] C. Bouveyron « Modélisation et classification des données de grande dimension. Application à l'analyse d'images» Thèse de doctorat. Université Grenoble 1. Septembre 2006.

[Chow-89] Y. Chow et R. Schwartz «The N-Best Algorithm: An Efficient Procedure for Finding Top N Sentence Hypotheses» Proceedings of the DARPA Speech and Natural Language Workshop, Octobre 1989.

[Clav-07] C. Clavel « Analyse et reconnaissance des manifestations acoustiques des émotions de type peur en situations anormales » Thèse du doctorat. L'École Nationale Supérieure des Télécommunications Spécialité. Mars 2007.

[Cloa-06] G. Cloarec, D. Jouvet « Influence de la corrélation entre le pitch et les paramètres acoustiques en reconnaissance de la parole » IRISA / CNRS, le XXVIèmes JEP. Juin 2006.

[Davi-80] S.B Davis et P. Mermelstein « Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences» IEEE Transactions on Acoustics, Speech, and Signal Processing, 28(4) :357–366, 1980.

[Deby-07] M. Debyeche « Reconnaissance automatique de la parole appliqué à la langue arabe » Thèse de Doctorat d'Etat, USTHB, 2007.

[Deha-07] N. Dehak, P. Kenny et P. Dumouchel « Continuous Prosodic Features and Formant Modeling with Joint Factor Analysis for Speaker Verification » In Proceedings of Interspeech 2007, pp. 1234-1237. Août 2007.

[Dekk-03] M. Dekker « Speech Processing, a Dynamic and Optimization-Oriented Approach» Series: Signal Processing and Communications Series . ISBN-13: 9780824740405. Juin 2003.

[Delé-96] P. Deleglise, A. Rogozan , M. Alissali, « Asynchronous integration of audio and visual sources in bi-model automatic speech recognition» Proceedings of the VIII European Signal Processing Conference, Trieste (Italy). Septembre 1996.

[Doss-05] M. Doss «Using auxiliary sources of knowledge for automatic speech recognition » Thèse PHD; École Polytechnique Fédérale de Lausane. Juillet 2005.

[Duto-00] T. Dutoit « Introduction au Traitement Automatique de la Parole» Rapport de la faculté polytechnique de Mons, Belgium, 2000.

[El-Bè-90] M. El-Bèze « Choix d'Unités Appropriées et Introduction de Connaissances dans des Modèles Probabilistes pour la Reconnaissance Automatique de la Parole» Thèse de Doctorat en Informatique Fondamentale, Université Paris VII. Novembre 1990.

[Ezza-02] H. Ezzaidi « Discrimination Parole/Musique et étude de nouveaux paramètres et modèles pour un système d'identification du locuteur dans le contexte de conférences téléphoniques » Thèse de doctorat. L'Université du Québec à Chicoutimi. Département des Sciences Appliquées. Octobre 2002.

[Forn-73] G.D. Forney « The Viterbi algorithm» Proc. of the IEEE, vol. 61, no. 3, pp. 268-278, 1973.

[Geno-98] D. Genoud « Reconnaissance et transformation de locuteurs » Thèse du doctorat. L'Ecole Polytechnique Fédérale de Lausanne (EPFL). 1998

[Gers-92] A. Gersho and R. Gray « Vector Quantization and Signal Compression» Kluwer Academic Press, 1992.

[Gray-84] R.M. Gray « Vector quantization » IEEE ASSP Mag., vol. 1(2):4-29, 1984.

[Hass-02] H. Ezzaidi « Discrimination Parole/Musique et étude de nouveaux paramètres et modèles pour un système d'identification du locuteur dans le contexte de conférences téléphoniques» Thèse de doctorat. L'Université du Québec à Chicoutimi ; Département des Sciences Appliquées. Octobre 2002.

[Herm-90] P. Hermansky, «Perceptual Linear Predictive (PLP) analysis of speech » Journal of the Acoustical Society of America, 87-4 :1738–1752, 1990.

[Hoch-94] M. Hochberg, S. Renals et A. Robinson, « ABBOT : The CUED hybrid connectionist-HMM large-vocabulary recognition system» In Proc. ARPA Spoken Language Technology Workshop, 1994.

[Igou-98] S. Igounet « Eléments pour un système de reconnaissance automatique de la parole continue du français » Thèse de doctorat en Informatique, Université d'Avignon et des Pays de Vaucluse. juillet 1998.

[Jaco-95] B. Jacob « Un outil informatique de gestion de Modèles de Markov Cachés : expérimentations en reconnaissance automatique de la parole » Thèse de doctorat, Université Paul Sabatier. Septembre 1995.

[Jeli-76] F. Jelinek « Continuous speech recognition by statistical methods» Proc. of the IEEE, vol. 64, no. 4, pp. 532-556, 1976.

[Juan-82] B. Juang, D. Wang, A. Gray « Distortion Performance of Vector Quantization for LPC Voice Coding » IEEE Trans ASSP, 30(2), 1982.

[Keil-00] F. Keiler , D. Arfib et U. Zolzer « Efficient Linear Prediction for Digital Audio Effects» Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, Italy. Décembre 2000.

[Klat-77] D. H. Klatt, « Review of the ARPA Speech Understanding Project » JASA, Vol. 62,N°6, pp. 1345-1366. Décembre 1977.

[Lame-96] L. F Lamel, G Adda et M Adda-Decker « Les lexiques de prononciation dans les systèmes de reconnaissance de la parole ».Séminaire GDR-PRC, Lexique et communication parlée, Toulouse. 1996

[Lang-95] PH. Langlais «Traitement de la prosodie en reconnaissance automatique de la parole» Thèse du doctorat. L'université d'Avignon et des Pays de Vaucluse. Octobre 1995.

[Lévy-03] C. Lévy, G. Linarès et P. Nocera, «Comparison of several acoustic modeling techniques and decoding algorithms for embedded speech recognition systems» Workshop on DSP in Mobile and Vehicular Systems, Nagoya - Japan, 2003.

[Loiz-03] P. Loizou «COLEA: A MATLAB software tool for speech analysis» Depuis le site web :http://www.utdallas.edu/~loizou/ speech/colea.htm. Octobre 2003.

[Mark-76] J.D. Markel et A.H. Gray « Linear Prediction of Speech » Communication and Cyberneties,Springer-Verlag.Berlin Heidelberg New York ,1976.

[Mary-08] L. Mary, B. Yegnanarayana « Extraction and representation of prosodic features for language and speaker recognition» Speech Communication 50, 782–796.April 2008.

[Morg-95] N. Morgan, H. Bourlard « Continous Speech Recognition - An introduction to the hybrid HMM/connectionist approach", IEEE Signal Processing Magazine, pp. 25-42, mai 1995.

[Nguy-02] Q.C. Nguyen « Reconnaissance de la Parole en Langue Vietnamienne » thèse de doctorat. Institut national polytechnique de Grenoble. Juin 2002

[Noll-67] A.M. Noll « Cepstrum Pitch Determination » Journal of the Acoustical Society of America, vol 14, pp 293-309, 1967.

[Rabi-77] L.R. Rabiner « On the Use of Autocorrelation Analysis for Pitch Detection» IEEE transaction on acoustics, speech, and signal processing,vol-25, 1. Février 1977 .

[Rabi-89] L.R. Rabiner, «A tutorial on hidden Markov models and selected applications in speech recognition» Proc. of the IEEE, vol. 77, no. 2, pp. 257-286, 1989.

[Rabi-93] L. Rabiner et B-H. Juang. «Fundamentals of speech recognition. signal» Processing. Prentice-Hall Inc. 1993.

[Rogo-99] A. Rogozan « Etude de la fusion des données hétérogènes pour la reconnaissance automatique de la parole audio-visuelle» Thèse doctorat de l'université d'Orsay Paris XI. Juillet 1999.

[Roua-2005] J-L. Rouas « Caractérisation et identification automatique des langues» Thèse du doctorat. L'Université Toulouse III – Paul Sabatier. Mars 2005.

[Sako-78] H. Sakoe et Chiba « Dynamic programing algorithm optimization for spoken word recognition» IEEE Trans. on ASSP, 26(1):43–49, 1978.

[Skap-92] D. M Skapura et J. A .Freeman « Backpropagation. Neural Networks Algorithm Applications and Programming Techniques » 89-125. 1992.

[Sene-88] S. Seneff et V. Zue, «Transcription and alignment of the TIMIT database» In Getting started with the DARPA TIMIT CD-ROM: an acoustic-phonetic continuous speech database, NIST, 1988.

[Step-03] T. A. Stephenson « Speech Recognition with Auxiliary » thèse PHD. École Polytechnique Fédérale de Lausane. Mai 2003.

[Trem-82] T.-E. Tremain « The government standard Linear Predictive Coding algorithm: LPC10» Speech Technology Magazine, tome 1-2, pages 40–49, 1982.

[Youn-89] S.J. Young, N.H. Russel et J.H. Thornton, « Token Passing: a Simple Conceptual Model for Connected Speech Recognition Systems» Cambridge University Engineering Departement rapport technique, 1989.

[Youn-97] S. Young, J. Odell, et d'autres « Hidden Markov model toolkit V2.1 reference manual» Technical report, Speech group, Engineering Department, Cambridge University, UK.1997.

[Youn-05] S. Young, J. Odell, et d'autres « The HTK Book Version 3.3» Speech group, Engineering Department , Cambridge University. April 2005.

[Vite-67] A.J Viterbi « Error bounds for convolutional codes and an asymptotically optimal decoding algorithm» IEEE Trans. Informat. Theory, vol. IT-13. 1967

ANNEXE A

ANNEXE A : L'outil HTK

Introduction

Nous décrivons dans cette partie les détails techniques de la mise en œuvre de l'outil HTK (Hidden Markov model Toolkit) pour la construction d'un système de reconnaissance par mots de la parole continue. L'implémentation de ce système est réalisée par un programme écrit en langage script *Perl* sous le système d'exploitation *Windows*.

HTK sous Windows

L'utilisation de l'outil HTK sous Windows commence par le téléchargement de la version binaire disponible sous format '.exe', ensuite elle est exécutée directement depuis le langage de commande MS-DOS. La version la plus récente sous format binaire est la version HTK3.3, et c'est celle-là que nous avons utilisée dans notre travail. Son téléchargement est effectué via l'adresse web : http://htk.eng.cam.ac.uk/ftp/software/htk-3.3-windows-binary.zip

Les outils HTK utilisés

L'implémentation de notre système de reconnaissance fait appel à onze outils HTK qui sont :

-HCopy : Calcul les paramètres acoustiques à partir de fichiers signaux.
-HCompv : Initialise les paramètres d'HMMs.
-HERest: Ré-estime les paramètres d'un ensemble d'HMMs par l'algorithme Baum-Welch.
-HVit: Décode les signaux à identifiés par l'algorithme de Viterbi.
-HHed : manipule la structure des HMMs.
-HLstat: Calcul le modèle de langage.
-HBuild : Construit le réseau syntaxique
-HList : Visualise des fichiers de données.
-HLed : Manipule les fichiers d'étiquettes.
-HDMan : Crée le dictionnaire.
-HResult : Calcul le taux de reconnaissance.

Étapes de la réalisation d'un système de reconnaissance

1- Préparation des données acoustiques et lexicales
2- Description des modèles de Markov : définir les modèles HMMs prototypes
3- Apprentissage : apprentissage des modèles HMMs avec le corpus d'entraînement.
4- Reconnaissance : définir le réseau lexico-syntaxique à suivre, puis effectuer la reconnaissance et l'évaluation des performances sur un corpus de test.

1. Préparation des données

L'étape de préparation de données consiste à fournir les données acoustiques et lexicales nécessaires pour l'étape d'entraînement et l'étape de test.

1.1 Paramètres acoustiques

Les paramètres acoustiques retenus pour modéliser le signal parole sont 12 coefficients MFCCs plus leurs dérivées premières et secondes. Les fichiers de paramètres sont calculés par l'outil **HCopy** qui prend comme entrée deux fichiers. Le premier fichier indique la liste des fichiers parole à paramétrer et l'emplacement souhaité pour les stocker. Le deuxième fichier sert à mentionner la configuration acoustique souhaitée. L'outil HCopy rend les fichiers de paramètres calculés dans l'emplacement indiqué.

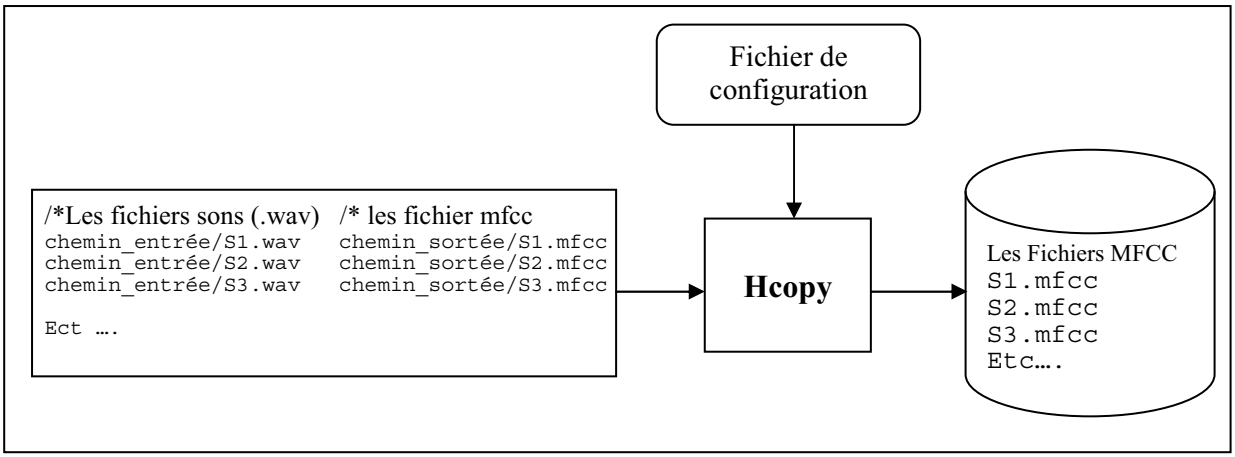


Figure 1 : Le calcul des paramètres acoustiques par l'outil HCopy.

Un exemple des informations qui peuvent être données dans le fichier de configuration sont:

```
SOURCEFORMAT = TIMIT      # le format des fichiers sources
TARGETKIND = MFCC_D_A          #  le type de coefficients à utiliser

# L'unité de temps est = 0.1 micro-second :

WINDOWSIZE = 250000.0      # = 25 ms = la longueur de la trame
TARGETRATE = 100000.0      # = 10 ms = la période des trames

USEHAMMING = T             # utilisation de la fenêtre de pondération Hmming
PREEMCOEF = 0.97           # le coefficient de préaccentuation
NUMCHANS = 26              # le nombre des bancs de filtre MEL
CEPLIFTER = 22             # la longueur des banc de filtre MEL

 NUMCEPS = 12              # le nombre de coefficients MFCCs.
```

Figure 2: Exemple d'un fichier de configuration pour l'outil HTK HCopy.

Commande HTK sous perl : le calcul des paramètres MFCCs peut être effectué avec la commande perl 'system' comme suit :

```
system("HCOPY -C $fich_config_mfcc  $fichier_liste_chemins ");
```

tel que :

- **$fich_config_mfcc** : est le fichier de configuration des paramètres acoustiques.
- **$list_liste_chemins** : est le fichier qui contient la liste des fichiers parole et des coefficients MFCC.

Pour visualiser le contenu d'un fichier MFCCs, HTK propose l'outil **HList**, qui fonctionne comme suit :

```
system("HList   -h -e 2  SA1.mfcc");
```

-h: permettre l'affichage des informations de la tête de fichier (header, en anglais)
-e 2 : permettre l'affichage de deux lignes seulement.

```
---------------------------------- Source: SA1.mfcc ----------------------------------------------------
  Sample Bytes:  48       Sample Kind:   MFCC
  Num Comps:     12       Sample Period: 10000.0 us
  Num Samples:   395      File Format:   HTK
------------------------------- Observation Structure ---------------------------------------------------
x:      MFCC-1  MFCC-2  MFCC-3  MFCC-4  MFCC-5  MFCC-6  MFCC-7  MFCC-8  MFCC-9  MFCC-10  MFCC-11  MFCC-12
------------------------------------ Samples: 0->4 -----------------------------------------------------
0:     -21.427  -1.583  -5.036  -0.172  -5.287   2.142   4.020   2.780  -3.007  -0.876   1.560    -3.646
1:     -18.490   1.303  -4.545  -4.229  -9.187  -4.599   2.357  -0.508   0.994   0.222   2.772     4.462
2:     -17.065  -0.396 -10.693  -9.223 -13.201 -12.257 -11.571   0.095   0.559   6.355   1.196     5.491

---------------------------------------- END ---------------------------------------------------------
```

Figure 3 : Exemple de contenu d'un fichier MFCC, visualisé par l'outil HTK HList.

1.2 Paramètres lexicaux

Pour créer la transcription phonétique correspondante à chaque fichier parole de la base de données acoustique, il faut disposer d'une liste de toutes les phrases prononcées (cette liste est disponible avec la base TIMIT) et d'un dictionnaire phonétique. Il suffit alors de faire appelle à l'outil **HLed** en lui fournissant ces deux dernières ressources pour obtenir la transcription phonétique de la base acoustique.

1.2.1 Création du dictionnaire phonétique

La première étape dans la création du dictionnaire phonétique est de créer une liste ordonnée de tous les mots du vocabulaire qui appartient à la base de données. Pour créer cette liste, HTK facilite la tâche de ses utilisateurs en leur fournissant des scriptes qui permettent à partir d'un fichier texte, contenant la transcription en phrase des fichiers paroles, de créer une liste de tous les mots qui constituent ces phrases. Le script fournit est sous le nom : **prompts2wliste**.

Commande HTK sous perl : l'exécution de scripte prompts2wliste.pl est donnée par :

```
system("prompts2wlist.pl    transcription_phrases.txt    liste_mots.txt");
```

Avec:
- transcription_phrases.txt : est le fichier de la transcription en phrase des fichiers parole.
- liste_mots.txt : est le fichier résultat, il contient la liste de tous les mots du vocabulaire qui appartiennent aux phrases fournies comme entrée au script.

Le dictionnaire phonétique est, en suite, créé en utilisant l'outil **HDMan** et une source de dictionnaire extérieure. Comme dictionnaire extérieur, nous avons utilisé le dictionnaire libre Beep [Youn-05].

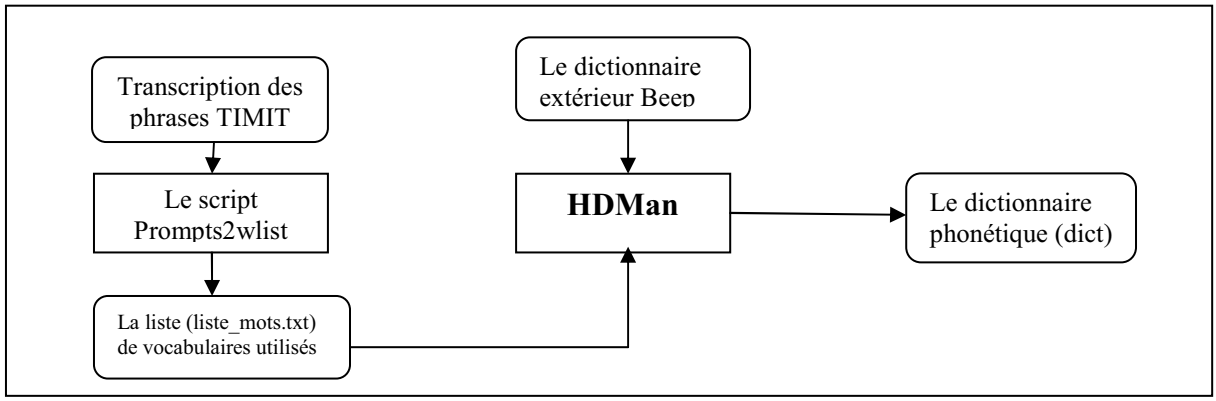


Figure 4 : Les étapes de création de dictionnaire phonétique par les outils HTK.

Un exemple de dictionnaire phonétique obtenu est donné par la figure 5 suivante :

```
A            ax sp
A            ey sp
ABBREVIATE   ax b r iy v ih ey t sp
ABDOMEN      ae b d ax m ax n sp
ABIDES       ax b ay d z sp
ABILITY      ax b ih l ih t iy sp
ABLE         ey b ax l sp
ABLY         ey b l iy sp
ABOLISH      ax b aa l ih sh sp
ABORIGINE    ae b ax r ih jh ax n iy sp
```

Figure 5 : Exemple de dictionnaire phonétique.

Dans ce dictionnaire, 'sp' fait référence à la petite pause entre un mot et un autre.

Commande HTK sous perl

```
system("HDMan  global.ded –w liste_mots.txt –n transcription_phonétique1
dictionnaire_phonétique      beep")
```

Avec:

-gobal.ded : est un fichier de configuration qui va permettre de poser quelques modifications sur le dictionnaire. Il va contenir les trois commandes suivantes :

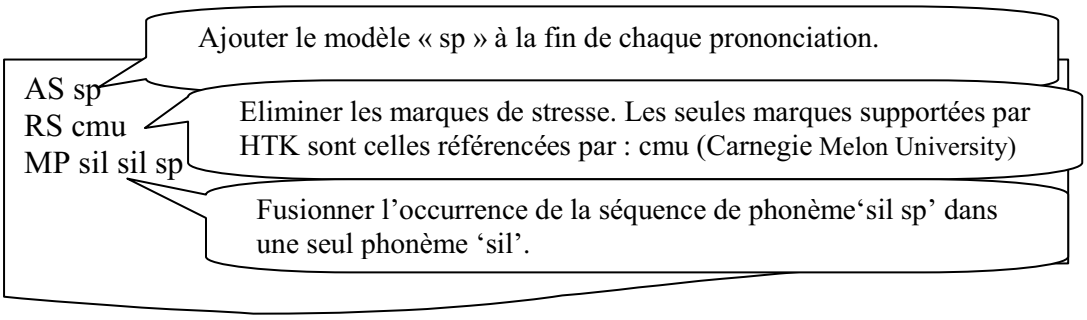


Figure 6 : le fichier de configuration pour l'outil HDMan.

1.2.2 Création de la transcription phonétique

Une fois le dictionnaire phonétique créé, nous l'utilisons pour associer à chacune des phrases de la base TIMIT une transcription phonétique. Pour des raisons de stabilité des procédures, le document d'aide fourni avec HTK [Youn-05] conseille de créer deux transcriptions phonétiques des phrases. Une première transcription qui ne prend pas en compte la petite pause (sp : Short-Pause, en anglais) entre les mots. Le phonème « sp » va être inséré ultérieurement dans la deuxième transcription, crée lors de la phase d'apprentissage.
La première étape dans la transcription phonétique consiste à transformer chaque phrase par un ensemble de mots (créer la liste words.mlf). HTK facilite la tâche pour ses utilisateurs en leur fournissant un script qui permet de réaliser cette transformation. Le script fourni dans ce cas est sous le nom **prompts2mlf**.

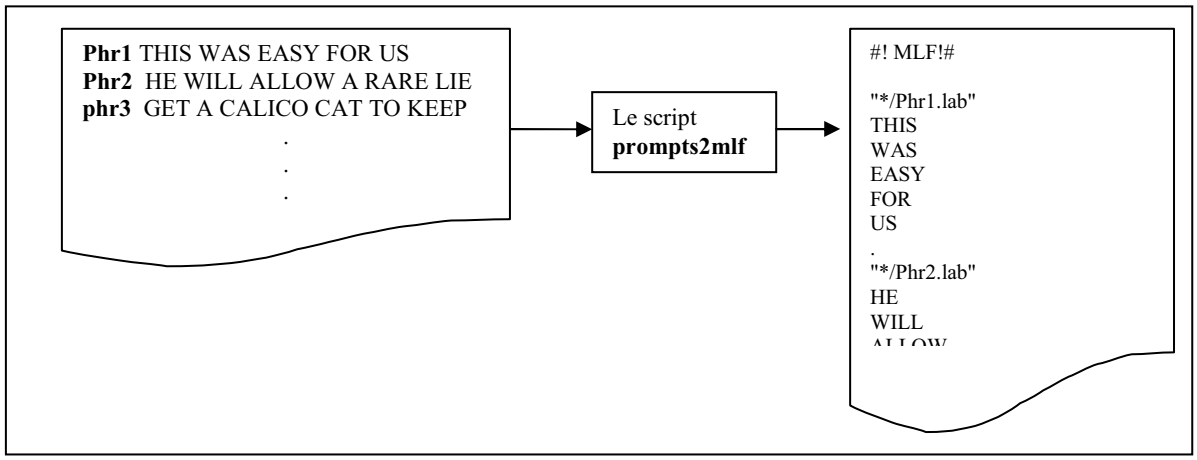


Figure 7 : Transcription en mots des phrases d'apprentissage.

Commande HTK sous perl

system("prompts2mlf.pl transcription_mots.mlf transcription_phrases.txt ");

Avec :

- transcription_mots.txt : est le fichier de la transcription en mots des fichiers parole.

A partir de la transcription en mots et du dictionnaire phonétique l'outil **HLed** va nous permettre de créer la transcription phonétique comme le montre le schéma suivant :

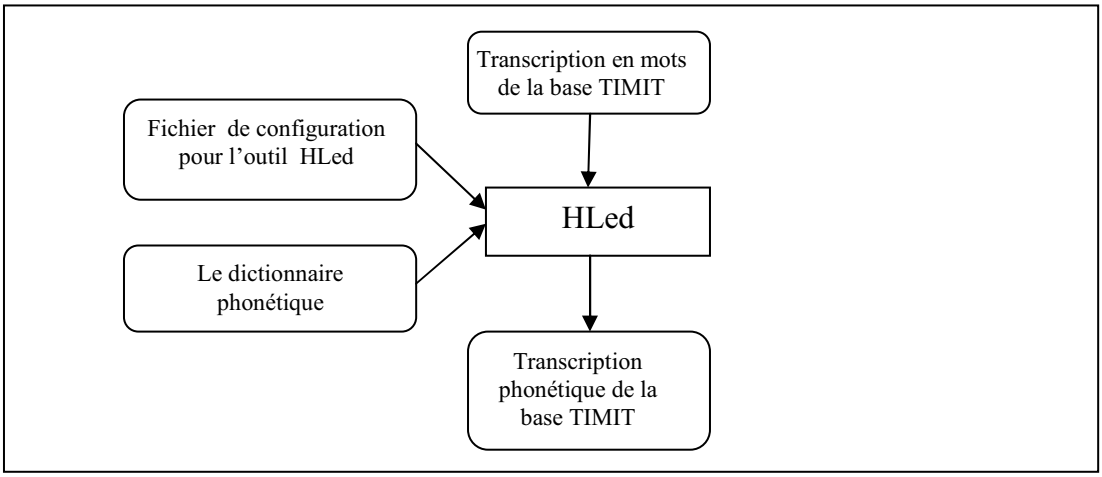


Figure 8 : Transcription phonétique des phrases d'apprentissage.

Le fichier de configuration contient les commandes suivantes :

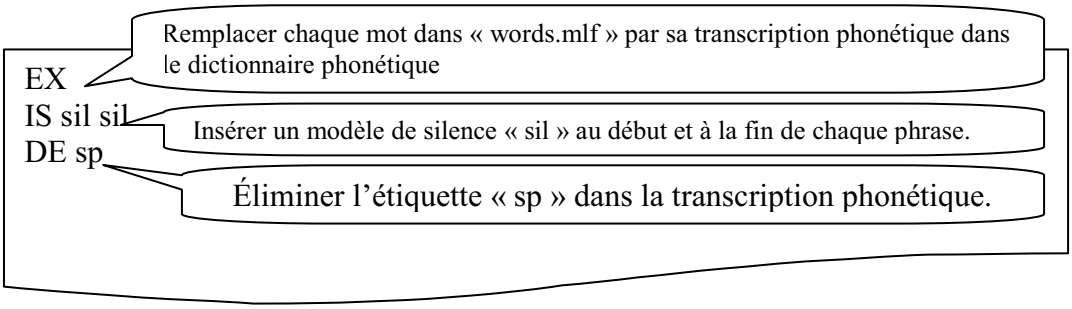


Figure 9 : Le fichier de configuration pour l'outil HLed.

La transcription phonétique obtenue est de la forme :

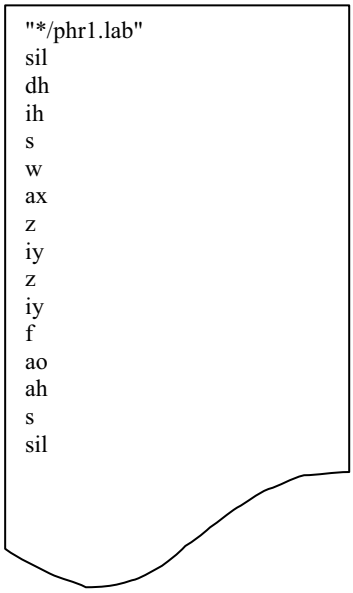


Figure 10 : La transcription phonétique des fichiers parole.

Commande HTK sous perl
system("HLed -d dictionnaire_phonétique –i transcription_phonétique.mlf mkphones1.led transcription_mots.mlf ");

Avec:
mkphones1.led : est le fichier de configuration pour l'outil HLed.

2. L'apprentissage

L'apprentissage des modèles revient à faire apprendre à partir des données acoustiques et lexicales déjà préparées, le vecteur moyen, la matrice de covariance et la matrice de probabilité de transition entre états, pour chaque modèle HMM.
Le processus d'entraînement commence par la définition manuelle sous une syntaxe propre à HTK des modèles HMMs (dits : modèles prototypes) associés pour chaque entité lexicale (le phonème dans notre cas), ensuite ces modèles prototypes sont initialisés et entrainés pour la première fois sur le corpus lexical qui n'inclut pas le phonème « sp ». Le deuxième entraînement est effectué après le rajout du modèle « sp ».

2.1 Définition du modèle prototype

Définir le modèle prototype revient à définir manuellement la structure des modèle HMMs à utiliser sous une syntaxe propre à HTK. Le modèle prototype utilisé dans notre travail est un modèle gauche-droite à trois états émetteurs avec une gaussienne à matrice de covariance diagonale de dimension 36 (12 coefficients MFCCs plus leurs dérivées premières et secondes) définit comme suit :

```
~o <VecSize> 36 <MFCC_D_A> <StreamInfo> 1 36
 ~h "proto"
<BeginHMM>
 <NumStates> 5
 <State> 2
  <Mean> 36
   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0....
  <Variance> 36
   1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0....
 <State> 3
  <Mean> 36
   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0....
  <Variance> 36
   1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0....
 <State> 4
  <Mean> 36
   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0....
  <Variance> 36
   1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0....
 <TransP> 5
 0.0 1.0 0.0 0.0 0.0
 0.0 0.6 0.4 0.0 0.0
 0.0 0.0 0.6 0.4 0.0
 0.0 0.0 0.0 0.7 0.3
 0.0 0.0 0.0 0.0 0.0
<EndHMM>
```

Figure 11 : Exemple d'un modèle HMM prototype.

2.2 Initialisation des modèles prototypes

L'initialisation des modèles prototypes peut se faire par deux façon différentes, soit par l'initialisation de chaque modèle indépendamment des autres en se basant sur la base de données acoustiques et l'alignement phonétique correspondant ou par une initialisation globale basée sur le calcul global des moyennes et des variances de toute la base de donnée acoustiques. Dans ce dernier cas, tous les modèles prototypes sont initialisés par des valeurs identiques. Nous avons opté la deuxième méthode afin d'initialiser nos modèles pour ce mode de reconnaissance.

L'outil HTK chargé de l'initialisation globale est l'outil **HCompV**, il va balayer toute la base de données acoustique pour calculer une moyenne et une variance globale. Ces paramètres globaux seront les paramètres initiaux de tout modèle prototype.

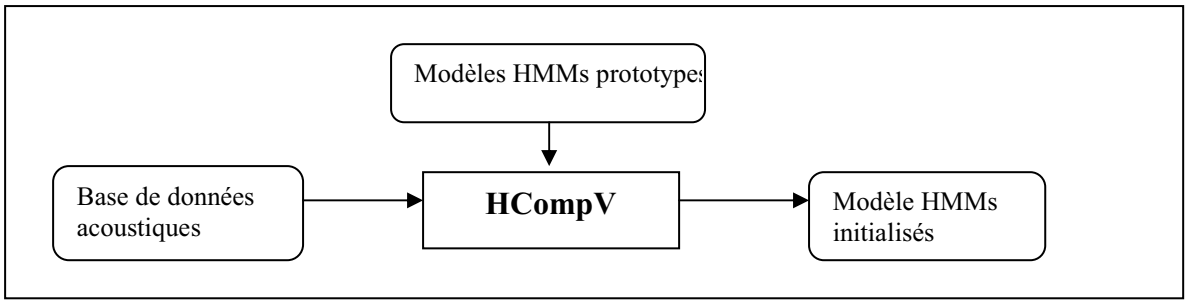


Figure 12 : Initialisation des modèles prototypes par l'outil HCompv.

Commande HTK sous perl
system("HCompv -S fichiers_acoustiques.txt prototype");
Avec:
-fichiers_acoustiques.txt : est un fichier texte contient la liste des fichiers MFCCs de la base d'apprentissage.
-prototype : est le fichier prototype définie manuellement dans l'étape précédente.

2.3 Entraînement des modèles HMMs

Après l'initialisation des modèles acoustiques, l'étape suivante consiste à les entrainer en utilisant la base acoustique et la base phonétique.

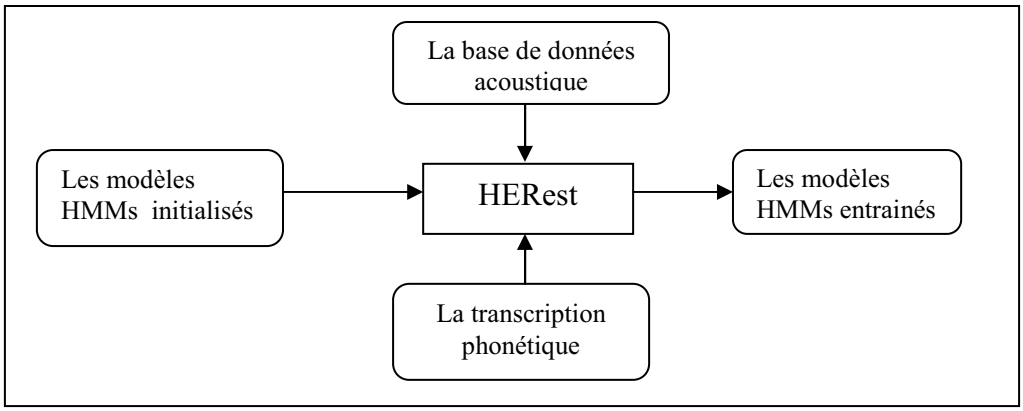


Figure 13: L'entraînement des modèles HMMs par l'outil HERest**.**

Commande HTK sous perl
system("HERest -I transcription_phonétique.mlf -S fichiers_acoustiques.txt -H hmms_initiaux liste_hmms.txt");

Avec:
liste_hmms : c'est une liste qui contient tous les noms de modèles HMMs qui vont être entrainés lors de cette étape.
hmms_initiaux : est l'ensemble des modèle HMMs déjà initialisés dans l'étape précédente.

2.5 Ajout de modèle « sp »

Comme nous l'avons déjà mentionné auparavant, et pour des raisons de stabilité des procédures HTK, le premier entraînement des modèles HMM ne prend pas en compte le phonème de Short-pause « sp ». Ce dernier va être ajouté après la première phase

d'entrainement. Pour ne pas créer une confusion entre le comportement du modèle Short-pause « sp » et le modèle silence « sil », HTK propose de les fusionner dans un seul et unique modèle par l'outil HHed.

L'outil HHed va fusionner les deux modèles « sp » et « sil » en créant une transition qui relie l'état central du modèle « sil » (le douzième état émetteur) avec le seul état émetteur du modèle « sp ». Ceci se fait sous HTK comme suit :

- Créer un modèle nommé « sp » à un seul état émetteur. Ce dernier va prendre comme paramètres les mêmes paramètres que ceux associés à l'état 2 du modèle « sil ».

- Exécuter l'outil HHed en lui fournissant le nouvel ensemble des modèles HMMs incluant le modèle « sp » et un script de commandes propre à HTK permettant de réaliser la fusion entre les deux modèles désirés (figure 15). Le scripte qui permet de réaliser cette tâche est inclus dans le package HTK téléchargeable. Il est donné sous le nom : **sil.hed.**

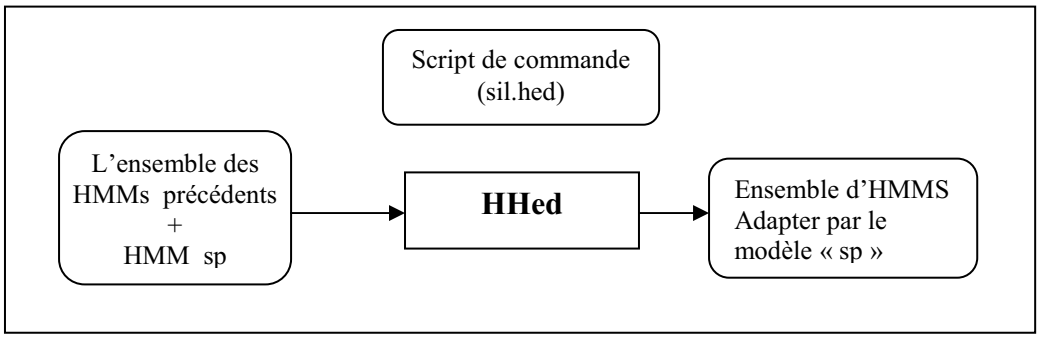


Figure 14 : Ajouter le moèle « sp » par l'outil HHed.

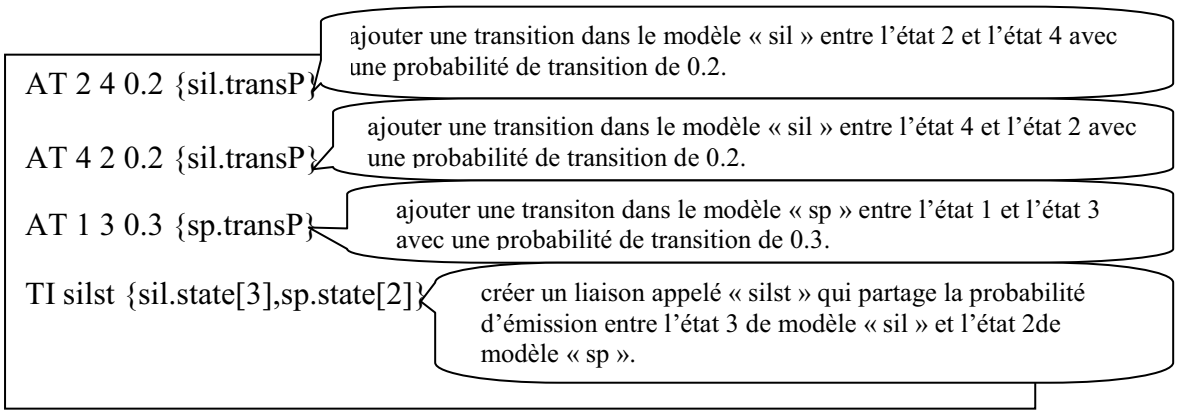


Figure 15 : Le fichier de configuration pour l'outil HHed.

Commande HTK sous perl

```
system("HHEd –H hmms_entrainés sil.hed transcription_phonétique2.mlf ");
```

Avec :
- **hmms_entrainés** : est l'ensemble de modèles HMMs déjà entrainé dans l'étape précédente.
- **transcription_phonétique2.mlf** : est la transcription phonétique qui contient le phonème « sp ».

Après cette adaptation des modèles par la Short_pause « sp », un nouvel entraînement des modèles est effectué par l'outil HERest.

3. Reconnaissance

La reconnaissance se fait avec l'algorithme de Viterbi, implémenté sous HTK avec l'outil **Hvit,** cet outil nécessite l'utilisation de différentes connaissances : les modèles HMMs (déjà entraînés), le réseau syntaxique des différents chemins à suivre, le modèle du langage et le dictionnaire.

3.1 Le modèle de langage

L'utilisation d'un modèle de langage permet d'améliorer la qualité de la reconnaissance. Le modèle de langage utilisé dans notre cas est de type bigramme de mots. Le bigramme de mots peut être estimé sur l'ensemble des paramètres lexicaux fourni pour l'apprentissage. Il donne la probabilité de transition entre chaque deux mots du vocabulaire. La plateforme HTK estime le bigramme de mots par l'outil **HLStats** comme suit :

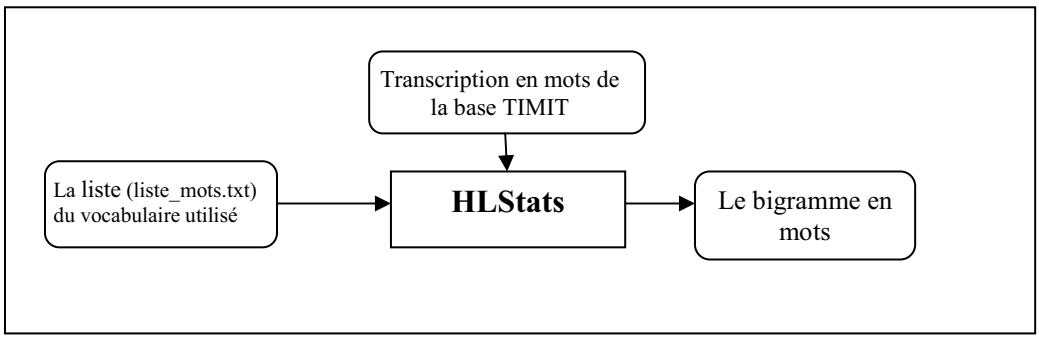


Figure 16 : L'estimation du bigramme de mots par l'outil HLStats.

Commande HTK sous perl

system("HLstats -b bigram_mots –o liste_mots.txt transcription_mots.mlf");

tel que:

-bigram_mots : est le fichier qui contient le modèle de langage de type bi-gramme.

3.2 Le réseau syntaxique

Le réseau syntaxique est le réseau qui sert à déterminer les différents chemins à suivre dans le processus de reconnaissance. Il consiste en ensemble de nœuds et d'arcs, chaque nœud représente un mot de vocabulaire et chaque arc représente une transition possible entre deux mots.

Comme aucune grammaire particulière n'est imposée sur les phrases de la base de données de test, le réseau syntaxique n'est qu'un modèle composite en boucle sur l'ensemble des mots du vocabulaire. Ce qui veut dire que tout enchainement de mots est autorisé. HTK nous permet de créer le réseau syntaxique grâce à l'outil **HBuild :**

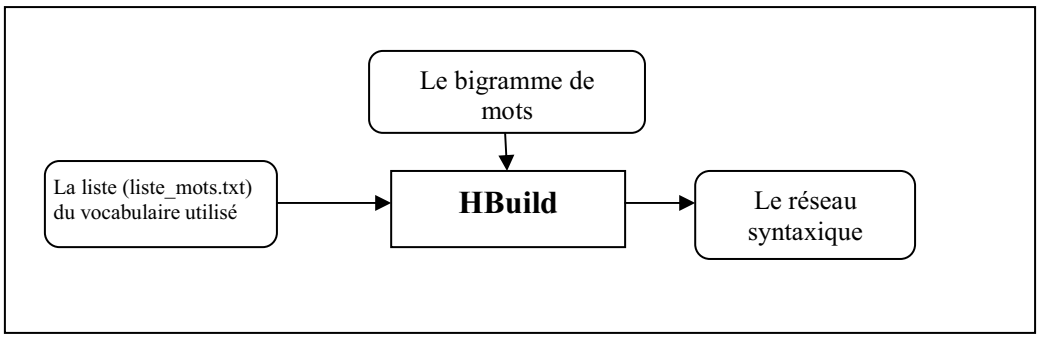


Figure 17: La construction d'un réseau syntaxique par l'outil HBuild.

Commande HTK sous perl
system("HBuild -n bigram_mots liste_mots.txt réseau_syntaxique.slf");
tel que :
-réseau_syntaxique : est le fichier qui contient le réseau syntaxique.

3.3 Le décodage

Après l'estimation du bigramme et la construction de réseau syntaxique, il ne reste qu'a exécuté la commande **HVite** pour effectuer la reconnaissance proprement dite. La commande **HVite** s'implémente de la façon suivante :

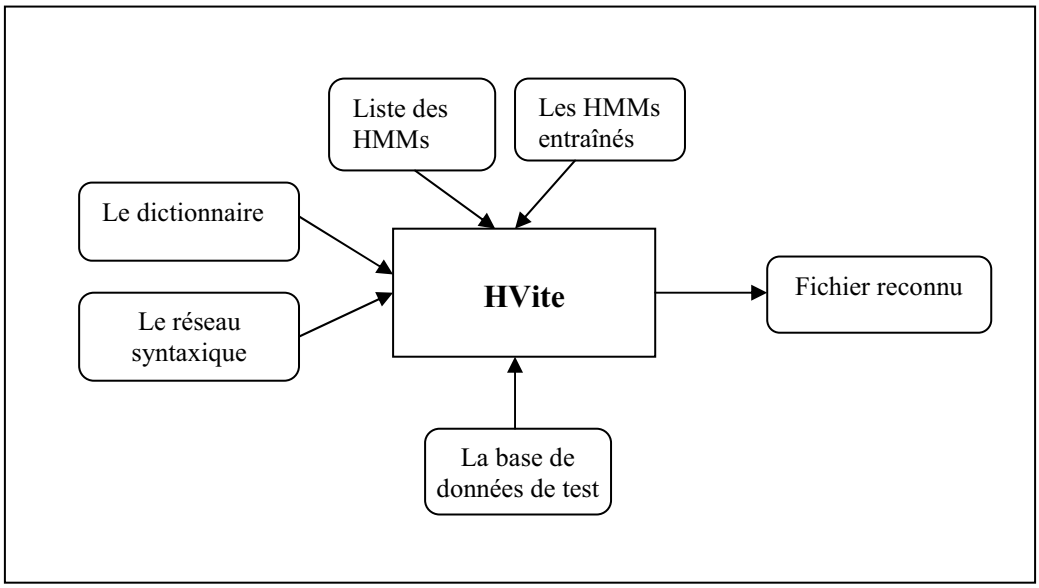


Figure 18 : Le décodage par l'outil HVite.

Commande HTK sous perl
system("$HVite   -H hmms_entrainés -S list_test.txt -i fichiers_reconnus
-w réseau_syntaxique dictionnaire liste_hmms) ;

Avec :
-**fichiers_reconnus :** le fichier résultats qui contient le décodage des fichiers de test.

4. Evaluation des résultats

L'évaluation des résultats revient à comparer les résultats obtenus après l'étape de décodage avec des résultats de référence. L'outil **HResuls** permet de réaliser cette comparaison par alignement dynamique entre le décodage obtenu et la source de référence.

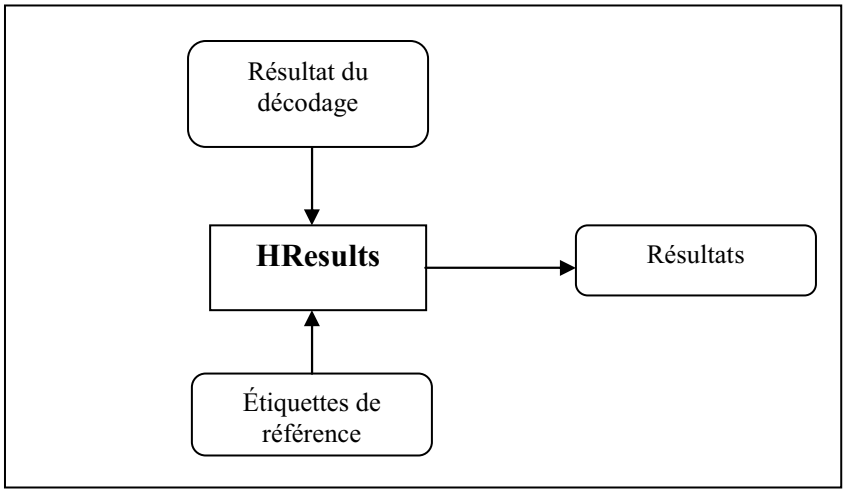


Figure 19: Evaluation des résultats par l'outil HResults.

Les résultats fournissent les taux d'identifications correct des phrases complètes (SENT: %Correct), ainsi que les taux de mots reconnus dans chaque phrases (WORD: %Corr=, %Acc=). Dans ce dernier cas HTK précise :
- le taux de reconnaissance avec insertion (%Corr): c'est-à-dire que les étiquettes identifiées en plus de celles prévues être identifiées n'affectent pas le taux de reconnaissance.
- le taux de reconnaissance sans insertion (%Acc) : c'est le taux de reconnaissance avec insertion en lui soustrayant le taux des étiquettes identifiées en plus de celles prévues être identifiées.

Commande HTK sous perl
system("HResult -I transcriptions_mots_test.mlf liste_hmms fichiers_reconnus.mlf > fichier_taux.txt");

Avec:
-fichier_taux.txt : est le fichier qui contient les taux de reconnaissance.
Résultats obtenus

. A partir d'une base d'apprentissage de 4000 phrases et une base de test de 24 phrases, les résultats obtenus sont donnés par le tableau suivant :

```
---------------------- Sentence Scores ----------------------
====================== HTK Results Analysis =======================
  ----------------------------------Overall Results -------------------------
SENT: %Correct=12.50 [H=3, S=21, N=24]
WORD: %Corr=69.06, Acc=58.01 [H=125, D=7, S=49, I=20, N=181]
====================================================================
```

Un exemple de la transcription obtenue par décodage est comparé à celle de référence comme suit :

SX248.rec: 57.14(57.14) [H= 4, D= 1, S= 2, I= 0, N= 7]
Aligned transcription: SX248.lab vs SX248.rec
LAB: **READING IN POOR** LIGHT GIVES YOU **EYESTRAIN**
REC: **READING IN POOR** LIKE USUAL **EYESTRAIN**

Nous rappelons que :

H : le nombre d'unités reconnues
D : le nombre d'omissions (le nombre d'unités non détectées).
N : le nombre total d'unités.
S : le nombre de substitutions (le nombre d'unités pour lesquels le système à commis une erreur).
I : le nombre d'insertions (le nombre d'unités admettent comme reconnus alors qu'aucun mot n'a été prononcé).

5. Conclusion

Comme nous l'avons mentionné au débout de cet annexe, le but de ce dernier était de fournir les détails technique nécessaire pour la construction d'un système de reconnaissance de la parole continue par mots. :

ANNEXE B

ANNEXE B : Paramètres auxiliaires avec le logiciel Praat.

Afin de mieux valider notre travail, nous présentons dans cette annexe les résultats obtenus par fusion qui utilise des paramètres auxiliaires extraits à l'aide du logiciel Praat [Boer-08]. Nous rappelons que la base de données utilisée a été superposée à trois types de bruit : le bruit d'usine, le bruit de la nature et le bruit de la foule. La figure B.1 suivante montre les spectrogrammes relatifs à ces bruits.

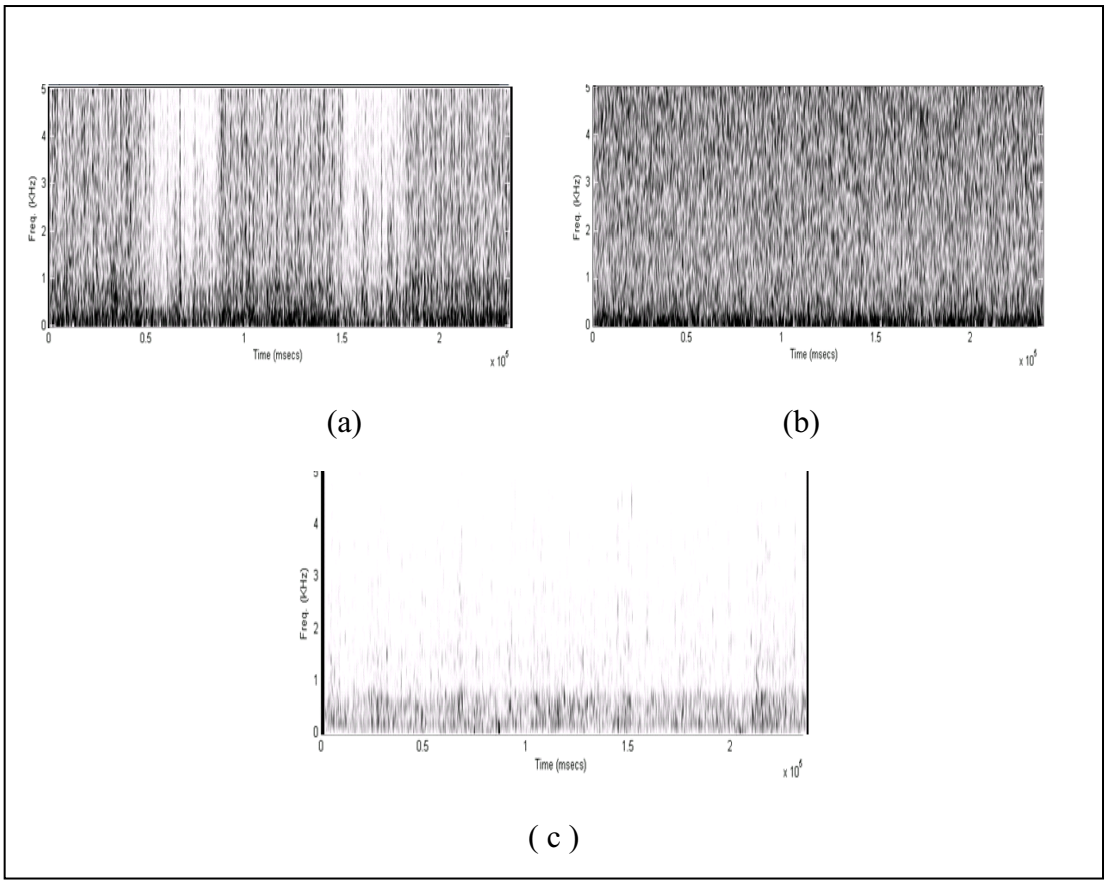


Figure B.1: Spectrogrammes des trois bruits utilisés : (a) le bruit d'usine ; (b) le bruit de la nature et (c) le bruit de la foule.

Le tableau B.1 suivant résume les résultats comparatifs des différents systèmes mis en œuvre, à savoir le système de référence, le système ID et le système IS. Pour ces deux derniers systèmes nous comparons également les résultats avec ceux déjà obtenus avec les paramètres auxiliaires extraits de nos programmes propres écrits en langage Matlab (ces résultats sont indiqués dans la colonne : calcul propre).

		Système de référence	Système ID		Système IS	
			Calcul propre	Calcul avec Praat	Calcul propre	Calcul avec Praat
calme		99,45%	98,52%	98,52%	99,46%	99,46%
Bruit d'usine	15 dB	79,61%	84,04%	**84,78%**	79,69%	79,69%
	10 dB	58,0 3%	64,58%	**80,30%**	58,15%	58,15%
	5 dB	33,12%	39,58%	**60,33%**	33,21%	33,21%
Bruit de la nature	15 dB	77,95%	81,64%	**89,58%**	78,02%	78,02%
	10 dB	57,10%	60,20%	**79,98%**	57,18%	57,18%
	5 dB	30,63%	33,03%	**59,32%**	30,71%	30,71%
Bruit de la foule	15 dB	95,11%	74,33%	**79,70%**	95,19%	95,19%
	10 dB	61,25%	63,25%	**74,17%**	61,29%	61,29%
	5 dB	38,93%	45,57%	**60,42%**	38,97%	38,97%

Tableau B.1 : Taux comparatifs des différents systèmes mis en œuvre.