Reconnaissance automatique de la parole continue : compensation des bruits par transformation de la parole

Vincent Barreaud

Département de formation doctorale en informatique **École doctorale IAEM Lorraine**
UFR STMIA

Reconnaissance automatique de la parole continue : compensation des bruits par transformation de la parole

THÈSE

présentée et soutenue publiquement le 9 Novembre 2004

pour l'obtention du

Doctorat de l'université Henri Poincaré – Nancy 1
(spécialité informatique)

par

Vincent Barreaud

Composition du jury

Rapporteurs :	Jean-François Bonastre Jean-Pierre Martens
Examinateurs :	Christian Wellekens Jean-Yves Marion Jean-Paul Haton Irina Illina
Invité :	Denis Jouvet

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Mis en page avec la classe thloria.

Remerciements

Les trois années que j'ai passé à la rédaction de cette thèse ont été très enrichissantes. Tout les jours n'avaient pas forcément la même teinte de rose, mais rétrospectivement, je les goûte tous. Au cours de cette période, j'ai eu le loisir d'assouvir en parti ma curiosité, la liberté d'explorer mes idées et la possibilité de découvrir un vaste et passionnant sujet qui sera le cadre de ma vie de chercheur.

Je tiens à remercier tous ceux qui m'ont soutenu et encadré tout au long de cette formidable expérience. Les membres de l'équipe PAROLE, en particulier, ont toujours été là pour m'aiguiller et parfois même, m'aiguillonner. Je remercie donc Irina I. et Dominique F. pour leurs conseils avisés, autant sur mes orientations scientifiques que professionnels. Je n'oublie pas Jean-Paul H., Odile M., Yves L., Christophe C., Anne B. et Christophe A..

J'aimerais remercier les membres du jury pour s'être intéressé à mes travaux et s'être investis dans leur lecture. J'en profite pour saluer les membres de la communauté scientifique qui ont eu la gentillesse de me faire part de leurs remarques et de leur conseils.

Je remercie de plus ce qui m'ont supporté avec bonne humeur pendant ces trois années. Je compte parmi ces bonnes âmes David L. et Vincent C. qui ont toujours eu l'amabilité de rire à mes blagues les plus mauvaises. J'ai une pensé particulière pour Armelle B., Vincent T., Regis L., Fabrice L., Alain D., Iadine C., Laurent J-P., Eric L., Renato D-C., Redime H., Bernard G.. Je suis heureux de les considérer comme des amis avant de les voir comme des collègues.

Avant de partir en stage post-doctoral, j'aimerais saluer tous les amis que je me suis fais à Nancy. Nous nous reverrons bientôt, je l'espère.

Je dédie ces années de travail à mon père, à ma mère, à mon frère et à ma famille. Elles m'ont éloigné d'eux seulement physiquement.

J'aimerais enfin remercier celle qui donne un sens à tout : Elise.

Le moment donné par le hasard vaut mieux que le moment choisi.
Proverbe Chinois

Table des matières

Chapitre 6
Structure hiérarchique de transformations de compensation

Chapitre 7
Reconnaissance dans un milieu non-stationnaire

Table des figures

Liste des tableaux

Introduction générale

La Reconnaissance Automatique de la Parole (RAP) est le domaine de recherche dont l'objectif est l'extraction par un ordinateur de l'information contenue dans un signal de parole. Ce domaine d'étude, né dans les années 50, a connu un grand essor lié à la découverte de nouveaux algorithmes et des avancées en mathématiques, électronique et informatique.

L'énergie investie dans ces recherches est à la mesure de son ambition : pouvoir interagir avec un ordinateur comme l'homme le fait avec un congénère, c'est-à-dire en lui parlant. Les domaines d'application sont très variés : commander les instruments d'une voiture sans quitter le volant des mains, dicter un texte à une machine, transcrire automatiquement des archives sonores...

Malgré des efforts constants et quelques avancées spectaculaires, la capacité d'une machine à reconnaître la parole est encore loin d'égaler celle de l'être humain. En effet, les Systèmes de Reconnaissance Automatique de la Parole (SRAPs) actuels voient leurs performances diminuer de manière significative lorsque les conditions dans lesquels ils ont été entraînés et celles dans lesquels ils sont utilisés diffèrent. On dit que ces systèmes sont peu *robustes*. Les causes de variabilité existantes entre ces conditions sont multiples. Elles peuvent être liées au locuteur, à l'environnement acoustique et/ou au circuit d'acquisition du signal sonore :

- La variabilité qui existe entre les locuteurs de test et d'entraînement est introduite par des différences dans le timbre de la voix, la vitesse d'élocution, l'accent, l'état psychologique et la morphologie. Cette variabilité peut être interprétée comme une distorsion (linéaire ou non) du signal de parole de test, en comparaison avec le signal de parole d'entraînement.
- Le matériel de capture du son introduit lui aussi une distorsion. Celle-ci peut être due à des bruits électriques qui peuvent varier dans le temps (la distorsion est non stationnaire). Il peut s'agir aussi simplement de l'architecture adoptée pour l'acquisition. Ainsi un changement de microphones entre les phases de test et d'entraînement peut modifier la forme générale du spectre de la parole et provoquer une chute des performances du SRAP.
- L'environnement acoustique ajoute au signal de la parole une composante perturbatrice. Cette composante est le plus souvent indépendante du signal de parole. La grande variabilité du signal de parole introduite par ces bruits de fond est certainement la source principale de la dégradation des performances des SRAPs. En effet, le bruit introduit par l'environnement acoustique peut être stationnaire ou au contraire varier plus ou moins rapidement. A la variabilité introduite par la nature du bruit s'ajoute donc une variabilité provoquée par son évolution dans le temps.

Généralement un SRAP cherche à se prémunir contre ces variations en posant des hypothèses sur leur nature. Ces hypothèses restreignent le cadre applicatif dans lequel il peut opérer efficacement. En d'autres termes le SRAP est robuste pour un ensemble limité de conditions d'utilisation. Pourtant un SRAP peut être amené à fonctionner dans des conditions différentes de celles pour lesquelles ces hypothèses ont été posées. Dans ces conditions d'utilisation inattendues (inédites) les performances du SRAP ont tendance à diminuer.

Un grand nombre de travaux de recherche s'attachent à augmenter la robustesse des SRAP par rapport à ces variabilités. Certaines approches s'emploient à adapter les SRAPs au locuteur de test. D'autres cherchent à obtenir une paramétrisation du signal de parole qui réduit l'influence du canal de transmission. D'autres enfin mettent en place des mécanismes pour supprimer la corruption de l'environnement acoustique.

Nos travaux de recherche s'inscrivent dans cette dernière catégorie d'algorithmes dits de compensation. La plupart des méthodes de cet ensemble utilise une mise en correspondance entre données de test et données d'entraînement pour élaborer une fonction de compensation. C'est le mécanisme de cet appariement qui implique l'utilisation d'hypothèses *a priori* sur la nature de l'environnement acoustique.

L'objectif de notre recherche est l'élaboration d'un algorithme de compensation permettant à un SRAP d'être *robuste* par rapport à une grande variété d'environnements. En particulier, cet algorithme doit permettre de garantir les performances du SRAP dans un milieu acoustique non-stationnaire. Nous cherchons de plus à poser un minimum d'hypothèses sur la nature de l'environnement acoustique. L'algorithme de robustesse obtenu pourra ainsi fonctionner dans des conditions d'utilisation très larges.

De nombreux travaux se sont donnés des objectifs similaires. L'un d'entre eux a abouti à la formulation du *Stochastic Matching*. Comme la plupart des méthodes de compensation, le *Stochastic Matching* utilise une mise en correspondance entre données de test et données d'entraînement pour élaborer une fonction de compensation. Cependant dans le *Stochastic Matching* cette mise en correspondance est assurée par l'appariement stochastique de la séquence des observations bruitées avec la séquence de modèles proposée par le processus de reconnaissance. Cette mise en correspondance stochastique permet de se passer d'hypothèse *a priori* sur l'environnement acoustique. Toutefois cet algorithme opère en temps-différé ce qui rend la compensation de bruits non-stationnaires peu efficace.

Les recherches exposées dans ce document se fondent sur cette méthode. Nous y proposons une approche en temps-réel du *Stochastic Matching*. Dans ces conditions, la compensation s'effectue en parallèle avec le processus de reconnaissance. L'algorithme de compensation obtenu ne pose pas d'hypothèses *a priori* sur l'environnement acoustique. Il augmente la robustesse d'un SRAP utilisé dans des environnements acoustiques inédits, non-stationnaires et variant lentement. Cette approche est améliorée pour obtenir un algorithme de compensation utilisable dans un environnement qui varie rapidement et inopinément.

Cette thèse s'organise en deux grandes parties. La première est consacrée au positionnement du problème de robustesse. Nous y exposerons d'abord les principes de la reconnaissance et les éléments principaux d'un système de reconnaissance. Nous présenterons ensuite les méthodes classiquement proposées pour en augmenter la robustesse. Nous étudierons les points forts et les points faibles de chacune d'entre elles. Enfin, nous présenterons en détail l'approche *Stochastic Matching* en temps-différé.

Dans une deuxième partie, nous exposerons notre approche temps-réel du *Stochastic Matching*. Cette partie comporte trois chapitres. Le premier est consacré à l'étude comparative de notre algorithme face aux méthodes de robustesse les plus répandues. De nombreuses expériences permettront de mesurer sa réactivité, son adaptativité et sa robustesse. Le deuxième chapitre est consacré à une amélioration de cet algorithme. On y présentera une structure arborescente de transformations construites selon notre approche du *Stochastic Matching*. Cette structure permet d'élaborer une fonction de compensation non-linéaire dont les paramètres dépendent de la région acoustique dont est issu la parole à compenser. Le processus de compensation ainsi obtenu

permet de distinguer le traitement de segments de parole en fonction de leurs caractéristiques acoustiques. Enfin, nous présenterons dans le troisième chapitre un mécanisme permettant à notre algorithme de fonctionner dans un environnement variant de façon inopinée. Nous verrons comment un processus surveillant les changements dans l'environnement peut déclencher la réinitialisation du processus de compensation et ainsi augmenter considérablement son efficacité.

1

Reconnaissance automatique de la parole

Sommaire

Ce chapitre présente le problème de la reconnaissance automatique de la parole (RAP). Pour commencer, nous mettrons en évidence la particularité du signal de parole et la façon dont il est codé pour être reconnu par le SRAP. Puis nous évoquerons le principe général de la RAP et en particulier l'approche que la plupart des SRAP utilisent de nos jours : l'approche Bayésienne. Enfin, nous entrerons dans les détails de l'implantation d'un système bayésien particulier : un système utilisant les modèles de Markov cachés (HMM).

Le système de reconnaissance de la parole (SRAP) générique que nous aurons décrit à la fin de ce chapitre atteindrait de très bonnes performances dans des conditions d'utilisation très précises. Cependant, un SRAP peut être amené à fonctionner dans des conditions autres que celles-ci. Dans ce cas, les performances du SRAP ont tendance à diminuer. On dit alors que le SRAP n'est pas *robuste*. A chaque étape de ce chapitre, nous ferons des constatations sur les éléments d'un SRAP qui mettront en évidence la nécessité de le rendre plus robuste.

1.1 Le signal de parole

1.1.1 Variabilité du signal de parole

La parole peut être vue comme un signal, la variation d'une valeur au cours du temps. Dans l'air, le signal de parole est une fluctuation locale de la pression. Cette variation de pression peut être captée par la membrane d'un microphone et convertie en une grandeur électrique (comme expliqué en introduction dans [d'Allessandro, 2002]). Cette conversion entre gradient de pression et grandeur électrique se retrouve aussi dans le système auditif de tout être humain [Calliope, 1989].

Le signal de parole est produit par un ensemble de cavités résonnantes, flux d'air et surfaces vibrantes appelé *appareil phonatoire*. L'onde acoustique porteuse du signal de parole est le résultat de l'excitation des cavités nasales et/ou orales par une ou deux sources acoustiques [Calliope, 1989]. La première est un flux laryngé. L'autre peut s'ajouter ou se substituer à la première : ce sont des bruits d'explosion ou de friction produits dans la cavité orale (de la glotte aux lèvres) [Landercy and Renard, 1982].

Le signal de parole est une concaténation de réalisations acoustiques, c'est-à-dire de mouvements et d'actions de l'appareil phonatoire. L'équivalent de ces réalisations élémentaires dans le signal acoustique sont les *phonèmes*. Ce sont les plus petites unités acoustiques. Elles permettent de distinguer deux mots [Calliope, 1989]. Par exemple, en français les sons **[p]** et **[b]** représentent deux phonèmes différents car la permutation de ces deux unités permet de distinguer de nombreuses paires de mots (paie/baie, pas/bas, pot/beau, ...).

L'appareil phonatoire est soumis à des contraintes mécaniques qui limitent les variations rapides de ses parties mobiles (la langue ne peut pas se déplacer instantanément de l'arrière du palais vers les lèvres). Cette influence qu'a la réalisation d'un phonème sur la réalisation de ses voisins est un phénomène connu sous le nom de *coarticulation*. Ainsi, pour un même locuteur, une réalisation acoustique associée à un même phonème peut varier en durée mais aussi dans la forme du conduit vocal utilisé pour le produire. Cette variabilité est étendue si l'on considère la totalité des réalisations d'un même phonème qui peut être prononcé par un ensemble de locuteurs d'âge, de sexe et de morphologie différents. En effet, la forme du conduit vocal, qui conditionne les propriétés de l'onde acoustique d'une réalisation, est propre à chaque individu. Il en résulte qu'un même phonème correspond à une grande variété de réalisations acoustiques.

En résumé, le signal de parole est un signal formé d'entités élémentaires stationnaires. Cependant, un même mot peut être prononcé de multiples façons. La variabilité observée dans les réalisations acoustiques d'un même mot peut avoir pour origine :

la variabilité intra-locuteur : Par sa nature mécanique, le processus de phonation n'est pas déterministe et un même mot, prononcé par un même locuteur, n'aura pas toujours la même réalisation acoustique selon sa place dans la phrase ou l'état émotionnel du locuteur.

la variabilité inter-locuteurs : La forme des ondes acoustiques émises ainsi que leur enchaînement dépend des caractéristiques morphologiques de chaque individu. La coarticulation, l'intonation et donc l'amplitude du signal seront influencées quant à elles par l'origine sociale et géographique du locuteur (comme rappelé dans [Haton *et al.*, 1991]).

A cette difficulté s'ajoute le fait que le signal de parole, une fois produit, transite par un milieu (l'air d'abord puis le microphone et le câblage) qui n'est pas exempt de perturbations et qui vient le corrompre. En effet, on recense plusieurs interactions possibles :

- des sons étrangers peuvent s'ajouter à l'onde de parole.

- la forme de l'onde sonore peut être modifiée par la géométrie de la pièce (phénomène d'écho).
- le signal acoustique peut être modifié lors de sa conversion par le microphone et son transit sous forme d'onde électrique.

Ces interactions augmentent d'autant la variabilité du signal de parole et les difficultés pour le reconnaître.

Le signal de parole, pour qu'il soit un moyen efficace de communication, est élaboré de façon à ce que le sens qu'il porte y soit robustement inscrit. De fait, le signal de parole code l'information, le sens, de façon redondante afin de résister aux perturbations du milieu ambiant. C'est pourquoi un être humain peut saisir ce qu'un nouvel interlocuteur lui dit, dans bien des contextes de bruits, même inédits. Un ordinateur ne peut pas gérer aussi bien cette variabilité. On dit que la RAP n'est pas aussi *robuste* que l'appareil auditif humain. De nombreuses méthodes ont été déployées pour y remédier. Dans cette thèse, nous proposons une nouvelle technique qui permet de rendre la plupart des systèmes de reconnaissance automatique de la parole plus robustes aux sources de variabilité que nous venons d'évoquer.

Dans la section suivante, nous présenterons la façon dont le signal est paramétrisé dans un SRAP, c'est-à-dire comment s'effectue la conversion du signal acoustique en un signal utilisable pour le traitement automatique de la parole. Cette conversion (ou codage) constitue la première étape de la reconnaissance et peut intégrer des mécanismes pour rendre la RAP plus robuste[1].

1.1.2 Paramétrisation

Un système de paramétrisation du signal a pour rôle de fournir et d'extraire des informations caractéristiques et pertinentes du signal. Il produit ainsi une représentation moins redondante de la parole. Le signal analogique est fourni en entrée et une suite discrète de vecteurs, appelés *vecteurs acoustiques* ou *vecteurs d'observation* est obtenue en sortie. En reconnaissance de la parole, les paramètres extraits doivent être :

pertinents : extraits de mesures suffisamment fines, ils doivent être précis mais leur nombre doit rester raisonnable afin de limiter le coût de leur calcul 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 aux variations de niveau sonore ou à un bruit de fond.

La conversion du signal acoustique en séquence de vecteurs d'observation repose sur un modèle régit par un ensemble de paramètres numériques. La paramétrisation du signal de parole consiste à estimer les valeurs des paramètres du modèle permettant l'observation du signal de parole. Il existe de nombreux modèles de parole. On distingue :

les modèles articulatoires : ils permettent de réaliser une simulation numérique du mécanisme de phonation. Les paramètres codent dans ce cas la position de la langue, l'ouverture des lèvres, ... La paramétrisation fait intervenir des équations de mécanique des fluides.

les modèles de production : ils permettent de réaliser une simulation de l'équivalent électrique de l'appareil phonatoire. Cet équivalent est en fait un modèle linéaire simplifié du modèle articulatoire. Dans ce cas, on considère le signal de parole comme étant produit par un ensemble de générateurs et de filtres numériques. Les paramètres calculés sont ceux

[1] cf chapitre 2

qui contrôlent ces éléments. On trouvera dans cette catégorie, les codages LPC (*Linear Prediction Coding*) et AR (*AutoRegressive coding*).

les modèles phénoménologiques : ils cherchent à modéliser le signal indépendemment de la façon dont il a été produit. Les algorithmes associés à la paramétrisation sont issus du traitement du signal. Les modèles basés sur l'analyse de Fourier en sont un exemple.

Les coefficients les plus utilisés en reconnaissance de la parole sont certainement les *cepstres*. Ils peuvent être extraits de deux façons : soit par l'analyse paramétrique, à partir d'un modèle de production de type LPC, soit par l'analyse spectrale (modèle phénoménologique). Dans le premier cas, on parlera de LPCC (*Linear Prediction Cepstral Coefficient*) et dans le deuxième de MFCC (*Mel Frequency Cepstral Coefficients*).

Nous avons vu le principe selon lequel le signal de parole, très complexe, est converti en une séquence de paramètres utilisable par le système de reconnaissance. Le sujet de la section suivante porte sur le principe même de la reconnaissance et les techniques employées classiquement dans ce domaine.

1.2 Principe de la reconnaissance

Le principe général de la reconnaissance automatique peut être résumé par la figure 1.1.

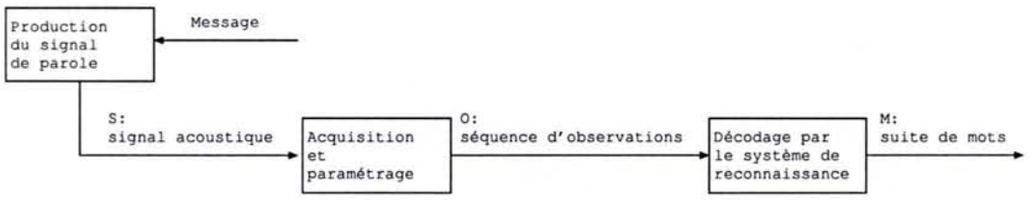


FIG. 1.1 – Principe de la reconnaissance de la parole.

- Le message à transmettre est converti en un signal acoustique S par l'appareil phonatoire.
- Le signal acoustique est alors transformé en une séquence de vecteurs d'observations $\mathbf{O}$.
- Finalement, le système de reconnaissance s'efforcera d'interpréter $\mathbf{O}$ en une séquence de mots M'.

Le principe même de la reconnaissance automatique de la parole est de parvenir à une séquence de mots M' correspondant au message transmis à partir de la séquence d'observations $\mathbf{O}$.

1.2.1 Reconnaissance par comparaison à des exemples

Les premiers systèmes de reconnaissance effectuaient une reconnaissance de mots par comparaison ou *par exemples*. L'idée consiste à faire prononcer un ou plusieurs exemples de mots susceptibles d'être reconnus. Ces exemples sont alors convertis sous forme de séquences de paramètres acoustiques (typiquement sous forme de LPC). Ceci correspond à la phase d'*entraînement* du système.

Lors de la phase de test, le signal de parole est lui-même encodé sous forme de séquence de vecteurs acoustiques. L'étape de reconnaissance consiste en fait à comparer la séquence obtenue

à toutes les séquences obtenues lors de la phase d'entraînement. Le mot reconnu sera alors celui qui correspond à la séquence d'entraînement se rapprochant le plus de la séquence de test.

La comparaison de deux séquences se fait grâce à un algorithme classique de programmation dynamique appelé DTW (*Dynamic Time Warping*) qui permet de tenir compte de la variabilité dans la vitesse de prononciation.

Cette technique ne peut s'appliquer qu'à condition que le nombre d'exemples différents (c'est-à-dire la taille du vocabulaire) soit faible. Ceci implique que :

- le vocabulaire de l'application soit limité, c'est-à-dire que le nombre maximal de mots à reconnaître soit faible.
- l'usage de l'application soit réservé à un seul locuteur qui aura lui-même entraîné le système.

Dans le cas contraire, les volumes de mémoire nécessaires à l'application seraient très importants et la vitesse d'exécution plus lente que celle observée pour d'autres types de SRAP.

1.2.2 Approche probabiliste

Dès que l'on cherche à concevoir un système réellement multi-locuteurs et à grand vocabulaire, il devient nécessaire de mener la reconnaissance sur des unités de parole plus petites que le mot (typiquement, les *phonèmes*). La reconnaissance ne se basera plus sur des exemples de réalisation mais sur des modèles probabilistes. On définit un modèle par unité acoustique, c'est pourquoi ils sont appelés *modèles acoustiques*. Ces modèles sont des structures paramétrées permettant de générer des séquences de vecteurs représentant des unités acoustiques. Ce sont des modèles statistiques entraînés à partir de collections de réalisations acoustiques d'unités élémentaires de parole. Les modèles utilisés peuvent être des réseaux de neurones artificiels (ANN, *Artificial Neural Network*) ou un ensemble de fonctions paramétriques comme un GMM (*Gaussian Mixture Model*).

L'approche probabiliste consiste à chercher la séquence de mots M' dont la probabilité sachant la séquence d'observations $\mathbf{O}$ est la plus grande.

$$M' = \arg\max_{M} P(M|\mathbf{O})$$

L'utilisation de la règle de Bayes permet de décomposer la probabilité *a posteriori* $P(M|\mathbf{O})$ en deux composantes :

$$M' = \arg\max_{M} \frac{P(M)P(\mathbf{O}|M)}{p(\mathbf{O})}$$

Ainsi, le but de l'approche probabiliste est de trouver la séquence de mots qui maximise le produit de :

$P(M)$: probabilité *a priori* d'observer la séquence M indépendamment du signal, déterminée par le *modèle de langage*.

$P(\mathbf{O}|M)$: probabilité d'observer la séquence des vecteurs acoustiques $\mathbf{O}$ sachant la séquence de mots spécifique M. Cette probabilité est déterminée lors de l'étape de reconnaissance des unités acoustiques décrite dans les sections suivantes.

$P(\mathbf{O})$ ne dépendant pas des paramètres du modèle, sa valeur n'intervient pas dans la maximisation.

La plupart des SRAP actuels utilisent intensivement une modélisation statistique du signal acoustique par des modèles de Markov cachés (HMM,*Hidden Markov Model*). La section suivante

propose un descriptif de l'utilisation des HMMS dans les systèmes de reconnaissance. Elle se conclura par un bref exposé de l'utilisation alternative des réseaux bayésiens et des modèles neuronaux dans la RAP.

1.3 Utilisation des HMMs en RAP

1.3.1 Présentation générale

L'outil statistique le plus souvent utilisé pour la modélisation acoustique dans les SRAP actuels est le HMM[2].

Un HMM peut être représenté comme un ensemble discret de nœuds (ou *états*) reliés entre eux par des arcs de transition (ensemble Q). Les transitions d'un état à un autre sont régies par des probabilités définies lors de l'apprentissage des modèles. Chaque état contient une densité de probabilité qui permet de mesurer la probabilité pour un élément $\mathbf{o}_t$ de la séquence d'observations $\mathbf{o}$ d'être associé à (émis par) cet état. C'est la probabilité d'émission.

Cet ensemble d'états, de transitions et de probabilités forme un automate qui modélise une unité acoustique. Il est conçu de sorte que toute prononciation de cette unité puisse être mise en correspondance avec un parcours dans l'automate depuis un état d'entrée jusqu'à un état de sortie. La mise en correspondance est stochastique, comme nous le verrons dans la suite et fournit donc un chemin *optimal* d'états $S = \{q(1), q(2), \ldots, q(T)\}$ qui permet de rendre compte de la séquence d'observations.

Les HMMs peuvent modéliser toute unité acoustique (mot, phonème, ...), selon l'application. Pour des SRAP de petits vocabulaires (reconnaissance de chiffres, par exemple), on pourra utiliser des modèles acoustiques de mots. Dans le cas d'un vocabulaire plus grand, on utilisera plutôt des modèles acoustiques de phonèmes ou d'unités acoustiques de taille réduite.

Formellement, le HMM modélisant une unité acoustique peut être entièrement défini par l'ensemble des paramètres λ :

$$\lambda = (N, A, B, \pi)$$

où

- N est le nombre d'états du modèle :

$$Q = \{q_i; i = 1 \ldots N\}$$

- A est la matrice des transitions entre les états de l'ensemble Q

$$A = \{a_{q_i q_j}\}$$

  Pour la plupart des SRAP utilisant des HMMs, $a_{q_i q_j} = P(q_i|q_j)$. Ces HMMs sont dits d'ordre 1. Dans ce cas, les probabilités de transition entre deux états ne dépendent ni du temps ni de l'historique des états. On utilise le plus souvent des HMM dits de *Bakis*. Ce sont des HMMs où

$$a_{q_i q_j} \begin{cases} = 0 & \text{si} \quad j < i; \\ > 0 & \text{si} \quad j \geq i \end{cases}$$

[2]Une bonne description de l'utilisation des HMMs en RAP peut être trouvée dans [Rabiner and Juang, 1993]

- B est l'ensemble des probabilités d'émission, c'est-à-dire la probabilité qu'un vecteur paramétrique $\mathbf{x}$ soit émis par un état particulier :

$$B = \{b_{q_j}(\mathbf{x}); j = 1 \dots N\} = \{P(\mathbf{x}|q_j); j = 1 \dots N\}$$

Dans les HMMs dits *continus*, B est représenté par une collection de densités de probabilité (PDF, *Probability Density Function*), le plus souvent des GMMs. Un GMM est la somme de K fonctions normales multidimensionnelles (ou *Gaussiennes*) de moyenne $\mu_{j,k}$ et de matrice de covariance $\mathbf{\Sigma}_{j,k}$, pondérées par $w_{j,k}$:

$$b_{q_j}(\mathbf{x}) = \sum_{k=1}^{K_j} w_{j,k} \mathcal{N}(\mathbf{x}; \mu_{j,k}, \mathbf{\Sigma}_{j,k})$$

- π est la distribution initiale des états

$$\pi = \{P(q(1) = j); j = 1 \dots N\}$$

où $q(0)$ est l'état initial.

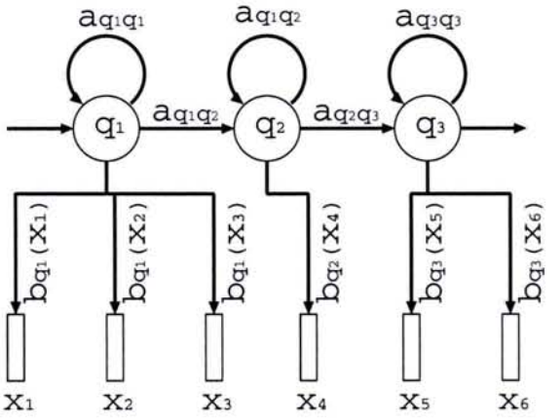


FIG. 1.2 – Modèle de Markov caché à 3 états.

La figure 1.2 représente un HMM à 3 états de type Bakis (seules les transitions non-nulles sont représentées). Chaque état peut émettre une observation x selon une probabilité régie par B.

1.3.2 Application à la reconnaissance

Soit $\mathbf{o}$ une sous-séquence de $\mathbf{O}$. La reconnaissance de la séquence de mots M passe par le calcul de la probabilité *a posteriori* qu'un modèle acoustique génère la sous-séquence d'observations $\mathbf{o}$:

$$P(\lambda|\mathbf{o})$$

La probabilité *a posteriori* que la séquence M ait été prononcée dans la séquence totale $\mathbf{O}$ s'obtient par un processus de concaténation qui relie les états de sortie de chaque modèle aux

états d'entrée de tous les autres par des probabilités. Le lecteur intéressé par ce processus de concaténation pourra se reporter au chapitre 7 de [Rabiner and Juang, 1993].

La loi de Bayes permet de réécrire l'expression :

$$P(\lambda|\mathbf{o}) = \frac{P(\mathbf{o}|\lambda)P(\lambda)}{P(\mathbf{o})}$$

et l'opération de reconnaissance se réduit à maximiser $P(\mathbf{o}|\lambda)P(\lambda)$, le produit de la *vraisemblance* de la séquence d'observations $\mathbf{o}$ étant donné le modèle λ par la probabilité *a priori* du modèle.

$P(\lambda)$, la probabilité du modèle acoustique λ est obtenue par le modèle de langage. Le modèle de langage, parfois appelé grammaire règle l'enchaînement des modèles acoustiques lors de la reconnaissance. Nous ne traiterons pas dans ce document son influence sur le mécanisme de reconnaissance car cette étude dépasse le cadre de notre recherche.

Maximiser $P(\mathbf{o}|\lambda)$ revient à chercher la séquence optimale des états de λ, c'est-à-dire la séquence d'états qui explique au mieux la séquence des observations $\mathbf{o} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$.

L'utilisation des HMMs dans un système de reconnaissance suppose de pouvoir résoudre les trois problèmes suivants [Rabiner and Juang, 1993] :

a. Evaluation : Etant donnée une séquence d'observations $\mathbf{o} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$ et le modèle $\lambda = (N, A, B, \pi)$, comment calculer $P(\mathbf{o}|\lambda)$?

b. Décodage : Etant donnée une séquence d'observations $\mathbf{o} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$ et le modèle $\lambda = (N, A, B, \pi)$, comment déterminer la séquence d'états $S = \{q(1), \ldots, q(T)\}$ qui explique le mieux $\mathbf{o}$?

c. Apprentissage : Comment déterminer les paramètres du modèle $\lambda = (N, A, B, \pi)$ afin de maximiser $P(\mathbf{o}|\lambda)$?

a. Evaluation

La probabilité de la séquence d'observations $\mathbf{o}$ sachant le modèle λ est obtenue par la somme de $P(\mathbf{o}, S|\lambda)$ sur toutes les séquences d'états S possibles :

$$P(\mathbf{o}|\lambda) = \sum_S P(\mathbf{o}, S|\lambda)$$

Or,

$$P(\mathbf{o}, S|\lambda) = P(S|\lambda)\ P(\mathbf{o}|S, \lambda)$$

La probabilité de la séquence S peut s'écrire sous la forme suivante :

$$P(S|\lambda) = \pi_{q(1)} \prod_{t=2}^{T} a_{q(t-1)q(t)}$$

sachant que la probabilité d'observer $\mathbf{o}$ pour une séquence d'états S de λ est :

$$P(\mathbf{O}|S, \lambda) = \prod_{t=1}^{T} b_{q(t)}(\mathbf{o}_t)$$

Pour un modèle à N états et une séquence d'observations de durée T le calcul de cette probabilité nécessite $(2T-1)N^T$ multiplications et $N^T - 1$ additions. Cependant, il est possible d'obtenir cette solution plus efficacement, en faisant intervenir :

- la probabilité *avant* : $\alpha_t(i) = P(\mathbf{o}_1, \ldots, \mathbf{o}_t, q(t) = q_i|\lambda)$, calculée récursivement :

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{q_i q_j}] b_{q_j}(\mathbf{o}_{t+1}) \tag{1.1}$$

avec

$$\alpha_1(i) = \pi(i) b_{q_i}(\mathbf{o}_1) \text{ et } P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

- la probabilité *arrière* : $\beta_t(j) = P(\mathbf{o}_{t+1}, \ldots, \mathbf{o}_T|q(t) = q_j, \lambda)$, elle aussi calculée récursivement :

$$\beta_t(i) = \sum_{j=1}^{N} a_{q_i q_j} b_{q_j}(\mathbf{o}_{t+1}) \beta_{t+1}(j) \tag{1.2}$$

avec

$$\beta_T(i) = 1 \text{ et } P(\mathbf{o}|\lambda) = \sum_{i=1}^{N} \alpha_t(i) \beta_t(i)$$

b. Décodage

L'algorithme suivant est appelé algorithme de *Viterbi*. Il permet de trouver la séquence d'états optimale qui maximise $P(\mathbf{o}, S|\lambda)$. Pour cela, on définit les quantités $\delta_t(i)$ et $\Phi_t(j)$:

$$\delta_t(i) = \max_{q(1),\ldots,q(t-1)} P(q(1), \ldots, q(t-1), q(t) = q_i, \mathbf{o}_1, \ldots, \mathbf{o}_t|\lambda)$$

et

$$\Phi_t(j) = \arg \max_{i \in \{1,\ldots,N\}} [\delta_{t-1}(i) a_{q_i q_j}]$$

Ces quantités sont initialisées :

$$\delta_1(i) = \pi_i b_{q_i}(\mathbf{o}_1) \text{ et } \Phi_1(i) = 0$$

Puis, par récurrence, on calcule :

$$\delta_t(j) = \max_{i \in \{1,\ldots,N\}} [\delta_{t-1}(i) a_{q_i q_j}] b_{q_j}(\mathbf{o}_t)$$

La probabilité finale P^{opt} du chemin optimum est alors :

$$P^{opt} = \max_{i \in \{1,\ldots,N\}} [\delta_T(i)]$$

Et la séquence d'états optimum $Q^{opt} = \{q^{opt}(1), \ldots, q^{opt}(T)\}$ est obtenue par :

$$q^{opt}(t) = \Phi_{t+1}(q^{opt}(t+1))$$

c. L'apprentissage

Ce problème consiste à trouver les paramètres du modèle $\lambda_m = (N_m, A_m, B_m, \pi_m)$ du mot M_m afin de maximiser $P(\mathbf{O}_m|\lambda_m)$, la probabilité d'une séquence d'observations formant le mot M_m. Il n'a pas de solution analytique connue. Idéalement, le critère d'entraînement serait la maximisation de la probabilité *a posteriori* globale de toutes les unités acoustiques en fonction des réalisations d'entraînement et de tous les modèles (Λ). Mais ce calcul est difficile et on transforme généralement cette probabilité par la loi de Bayes en :

$$P(M_m|\mathbf{O}_m, \Lambda) = \frac{P(\mathbf{O}_m|M_m, \Lambda)P(M_m|\Lambda)}{P(\mathbf{O}_m|\Lambda)}$$

Au lieu de maximiser la probabilité *a posteriori*, on cherchera à maximiser la vraisemblance globale (ML, *Maximum Likelihood*) $P(\mathbf{O}_m|M_m, \Lambda)$. Cette simplification de l'apprentissage lui fait perdre son caractère discriminant puisqu'on néglige dans ce cas l'influence de

$$P(\mathbf{O}_m|\Lambda) = P(\mathbf{O}_m|M_m, \Lambda) + \sum_{n \neq m} P(\mathbf{O}_m|M_n, \Lambda)$$

qui rend compte de l'importance de minimiser le terme $\sum_{n \neq m} P(\mathbf{O}_m|M_n, \Lambda)$. Des techniques d'apprentissage permettant de conserver cet aspect discriminant ont été élaborées. On comptera parmi elles celles utilisant les critères de maximisation de l'information mutuelle (MMI, *Maximum Mutual Information*) et de minimisation de l'erreur de classification (MCE, *Minimum Classification Error*). Une bonne description du MCE peut être trouvée en [Reichl and Ruske, 1995].

La technique la plus utilisée pour l'entraînement *ML* est celle de Baum-Welch (description disponible dans [Rabiner and Juang, 1993]). Il s'agit d'une procédure itérative, cas particulier de l'algorithme d'*Estimation-Maximisation.*

Le système de reconnaissance ESPERE (*Engine for SPEech REcognition*) utilisé pour toutes les expériences exposées dans ce document[3] utilise des HMMs comme modèles acoustiques, le décodage par l'algorithme de Viterbi et un apprentissage par celui de Baum-Welch.

1.3.3 L'alternative des réseaux de neurones aux HMMs

L'une des alternatives à l'utilisation d'HMMs en reconnaissance est le recours à des réseaux neuronaux.

Un réseau de neurones est une interconnexion de cellules simples (neurones). Chaque cellule possède plusieurs entrées et une sortie. Le signal de sortie peut être la somme pondérée (éventuellement seuillée) des signaux collectés en entrée [Junqua and Haton, 1995].

L'utilisation de ces réseaux est largement répandue dans les domaines devant résoudre des problèmes de classification et de reconnaissance des formes (traitement d'image, de signature sonar, ...). Les réseaux de neurones (ANN, *Artificial Neural Network*) possèdent des propriétés très appréciées en RAP :

- leur apprentissage est discriminant (ils permettent d'améliorer la reconnaissance d'une classe et simultanément de rejeter les autres classes).
- ils ne nécessitent pas d'hypothèses sur les propriétés statistiques des données en entrée (contrairement aux HMMs qui les modélisent par des PDFs).

[3] cf chapitre 4 pour une description du SRAP

Dans le cas des ANNs appliqués à la reconnaissance de la parole (mot ou tout autre unité acoustique), on utilisera le plus souvent des *perceptrons multicouches*. Plus généralement, on combinera le perceptron avec un algorithme d'alignement de type DTW, les distances locales utilisées lors de la DTW étant les sorties de l'ANN [Bourlard and Morgan, 1994]. En plus de leur utilisation dans le problème de reconnaissance, les ANNs peuvent aussi servir à prétraiter le signal de parole et à extraire des paramètres discriminants. En effet, les coefficients de pondération des couches cachés d'un ANN forment une série de paramètres caractérisant l'entrée.

L'architecture d'un réseau de neurones à retard (TDNN, *Time-Delay Neural Network*) est décrite dans [Waibel *et al.*, 1989]. La particularité d'un neurone de TDNN réside dans le fait que ses entrés à un instant sont constitués de données issues de l'instant présent mais aussi du passé et du futur. L'objectif est d'intégré des schémas temporels dans l'ensemble des données que doit généraliser le réseau de neurones. Un tel réseau combine la robustesse et le pouvoir discriminant des réseaux de neurones avec une architecture invariante par rapport au temps afin de former un identifieur de phonème très performant.

1.3.4 L'alternative des réseaux bayésiens dynamiques aux HMMs

Durant ces dernières années, les réseaux bayésiens sont devenus très répandus en intelligence artificielle. Formellement, un réseau bayésien (statique) est défini par la connaissance de deux éléments : un graphe acyclique orienté S et une paramétrisation numérique Θ. Etant donné un ensemble de variables aléatoires $X = \{X_1, \ldots, X_N\}$ et $P(X)$ sa distribution de probabilité jointe, le graphe S code les dépendances conditionnelles qui existent dans cette distribution.

Dans un HMM, les observations sont supposées être gouvernées par un processus dynamique caché (la séquence des états). Les hypothèses d'indépendance associées sont telles que le processus caché est markovien d'ordre 1 et chaque observation dépend seulement de la variable cachée courante. Cette modélisation suppose donc que l'hypothèse de dépendance entre observation et état caché est valable quelles que soient les données (robustesse) et l'application. En d'autres termes, on suppose que les GMMs modélisent la distribution des observations d'une unité acoustique avec une précision équivalente quelque soit le cadre applicatif.

Un HMM est un cas particulier de réseau bayésien. Dans les travaux sur les réseaux bayésiens en RAP comme [Deviren and Daoudi, 2001], on propose de ne faire aucune hypothèse *a priori* sur les dépendances. Ces dépendances s'établissent lors d'un apprentissage sur les données. Les modèles résultants représentent la parole avec une plus grande fidélité que les HMMs alors que la charge de calcul se trouve exclusivement dans la phase d'apprentissage.

Le lecteur intéressé pourra se rapporter à la thèse de Murat Deviren sur l'utilisation des réseaux bayésiens dynamiques en RAP [Deviren, 2004]. Les réseaux bayésiens dynamiques (DBN, *Dynamic Bayesian Network*) y sont décrits en profondeur, ainsi que leur utilisation et leur apprentissage.

1.3.5 L'alternative des modèles segmentaux aux HMMs

Dans [Ostendorf *et al.*, 1996], un ensemble de modélisations alternatives regroupées sous la dénomination *modèles segmentaux* est proposé. L'objectif de ces modèles est de palier à certains défauts des HMMs : la mauvaise modélisation de la durée des unités acoustiques et l'hypothèse irréaliste d'indépendance conditionnelle entre les observations successives.

L'espace des observations utilisés par ces méthodes n'est pas l'espace des vecteurs paramétriques comme dans le cas des HMMs, mais des séquences de trames consécutives . Les modèles segmentaux sont conçus pour former une séquence optimale dans le graphe des segmentations temporelles possibles d'une phrase à reconnaître. L'objectif poursuivi en utilisant des segments plutôt que des trames, est de rendre compte de la dynamique du signal de parole de façon précise (et fondée) [Hazen *et al.*, 2002]. Les vraisemblance calculée lors de la recherche de la meilleure segmentation implique le calcul de la vraisemblance d'un phonème, et non d'une trame, sans considération pour sa durée. Dans [Glass *et al.*, 1996], il est reporté que le SRAP à base de modèles segmentaux SUMMIT du MIT testé sur la base TIMIT obtient un taux de reconnaissance supérieur à un SRAP à base de HMMs.

L'utilisation de tels modèles s'est révélée particulièrement efficace dans des applications de reconnaissance de la parole spontanée. Pour cette tâche, les variations dans la vitesse de prononciation se sont révélées être la cause majeur des erreurs de reconnaissance. Dans [Ström *et al.*, 1999] par exemple, cette méthode est appliquée à travers SUMMIT aux données collectées par l'intermédiaire de l'interface de conversation JUPITER (délivrant des informations météo par téléphone).

1.4 Conclusion

Dans ce chapitre, nous avons vu que le problème de la reconnaissance peut se résumer de façon simpliste à un problème de reconnaissance de formes acoustiques : il s'agit d'associer des portions du signal de parole à des modèles pré-entraînés.

De nombreux efforts ont été apportés pour trouver des modélisations alternatives. Cependant, la plupart des SRAP utilisent encore de nos jours la technologie des HMMs. En effet, le temps qui a été consacré à leur développement ne peut être comparé à celui qui a été accordé aux nouvelles solutions. Par conséquent, les résultats obtenus par les solutions alternatives telles que les DBNs n'atteignent pas encore les standards fixés par les performances des HMMs. De fait on choisira le plus souvent pour un SRAP cherchant à obtenir de bonnes performances (pour des fins commerciales par exemple) une implémentation utilisant des HMMs.

L'objectif de notre recherche étant de trouver une solution de compensation robuste pouvant être utilisée avec une grande variété de SRAP, nous avons choisis d'utiliser pour nos expériences un SRAP utilisant des HMMs comme modèles acoustiques. Nous verrons dans le chapitre 5 que le mécanisme de compensation que nous proposons tire parti de la façon dont sont utilisés ces modèles acoustiques dans l'algorithme de reconnaissance de Viterbi. Cependant, il faut noter que, puisque les HMMs sont un cas particulier de Réseau Bayésien, la méthode de compensation que nous proposons pourrait être adaptée à un SRAP utilisant des Réseaux Bayésiens.

Nous avons pu voir dans ce chapitre que le processus de classification est optimisé pour le cas où les données d'apprentissage et de test sont de même nature (mêmes locuteurs, même vocabulaire, même condition d'enregistrement). Cependant, un SRAP n'est que très rarement confronté à des conditions d'utilisation équivalentes aux conditions d'apprentissage. Les données de test sont soumises à une forte variabilité qui ne peut être prise en compte par le SRAP générique décrit dans ce chapitre. On dit que ce SRAP n'est pas *robuste*.

Une grande partie des recherches en RAP ont pour objectif d'améliorer la robustesse des SRAP. Dans le chapitre suivant, nous décrivons les raisons de la variabilité des données d'utilisation. De plus, nous exposons les principes de quelques méthodes développées pour augmenter

la robustesse des SRAP. Cette présentation ne forme pas une liste exhaustive mais donne une vision des grands principes utilisés en robustesse et des inspirations ayant guidé notre recherche.

2

Approches classiques en robustesse

Sommaire

2.1 Introduction

Les systèmes de reconnaissance automatique de la parole (SRAP) voient leurs performances diminuer de manière significative lorsque les environnements dans lesquels ils ont été entraînés et ceux dans lesquels ils sont utilisés diffèrent. En effet, si près de 97.8% de taux de reconnaissance peut être obtenu par un système de reconnaissance multilocuteurs (1011 mots) entraîné et testé dans un milieu calme, ce même système n'obtient pas plus de 3% de reconnaissance si on ajoute un bruit blanc gaussien (avec un RSB de 0dB) au signal de test [Siohan, 1995]. De plus, un SRAP dont les modèles acoustiques ont été entraînés dans le même milieu que le milieu de test, même bruité, donnera toujours de meilleures performances qu'un SRAP entraîné dans un milieu calme. Cependant, un système utilisant des modèles entraînés dans un milieu bruité ou simplement avec des données issues d'un seul locuteur ne peut être utilisé que dans ce milieu ou avec ce locuteur. En effet, dans le cas contraire, ses performances seraient plus faibles que celles d'un SRAP multilocuteurs entraîné en milieu calme. C'est pourquoi les SRAP actuels

utilisent pour la plupart des modèles acoustiques de parole propre entraînés à partir de données de plusieurs locuteurs, bien que d'autres solutions puissent être envisagées comme l'apprentissage multiréférentiel[4] par exemple.

2.1.1 La fonction d'environnement

Comme vu dans le chapitre précédent, le signal de parole peut être corrompu de deux manières lors de sa transmission :

- des sons, indépendants du signal de parole, peuvent venir s'y ajouter. On parlera dans ce cas de bruits additifs.
- des filtres (mécaniques ou électriques) modifient la façon dont le signal transite dans l'air ou le circuit d'enregistrement. On parlera dans ce cas de bruits convolutifs.

La combinaison de ces perturbations transforme une séquence de parole propre en une séquence de parole corrompue d'une façon difficilement modélisable. Tous les efforts dans le domaine de la reconnaissance en milieu bruité ont pour objectif de réduire la différence entre les conditions d'entraînement (la parole propre) et celles de test (la parole bruitée). Il n'est pas aisé de formaliser la modification apportée par les sources de bruit au signal de parole. Cependant, il est courant de représenter leur contribution sous la forme de la *fonction d'environnement* :

dans le domaine temporel la fonction d'environnement a pour forme :

$$y(t) = x(t) \otimes h(t) + n(t)$$

avec
- $x(t)$ le signal de parole propre
- $y(t)$ le signal bruité
- $n(t)$ le signal de bruit additif
- $h(t)$ la réponse impulsionnelle du filtre représentant les sources de bruit convolutif

dans le domaine cepstral le bruit de canal n'est plus convolué au signal propre mais il resulte que le signal propre et les sources de bruit sont liés de façon non-linéaire. On peut alors définir la fonction d'environnement g dans le domaine cepstral :

$$\mathbf{y} = \mathbf{x} + g(\mathbf{x}, \mathbf{h}, \mathbf{n})$$

où $\mathbf{x}$, $\mathbf{h}$, $\mathbf{n}$ et $\mathbf{y}$ sont les représentations de $x(t)$, $h(t)$, $n(t)$ et $y(t)$ dans ce domaine.

De nombreuses techniques de robustesse reposent sur une approximation de la fonction d'environnement[5]. Cependant, la plupart des approches simplifieront la fonction d'environnement en limitant leur domaine d'application à des bruits additifs (par exemple la soustraction spectrale) ou convolutifs (par exemple la normalisation cepstrale)[6]. D'autres encore nécessiteront des informations *a priori* précises sur la nature du bruit, sous forme de *code-book* par exemple (comme le filtrage optimum probabiliste).

[4] voir la section 2.3.4 de ce chapitre
[5] cf section 2.2.4
[6] cf section 2.2.1

2.1.2 Caractérisation du bruit

Afin d'évaluer les performances des processus de robustesse, il faut disposer d'une mesure, d'un indice qui permette de rendre compte de la difficulté que présente un environnement du point de vue de la reconnaissance.

Prenons l'exemple d'un environnement dans lequel la pollution sonore est de type additif. Le bruit à l'intérieur d'une voiture en mouvement est issu de plusieurs sources. Le son issu du moteur est un bruit périodique de basse fréquence (inférieur à 1000Hz), le roulement des pneus sur l'asphalte est un bruit aléatoire dont le spectre se trouve en dessous des 1000Hz et le vent relatif provoque un bruit de fond aléatoire de fréquence supérieure à 500Hz.

La figure 2.1 montre les spectrogrammes d'une phrase prononcée dans l'habitacle (bruyant) d'une voiture. L'axe des abscisses représente le temps, celui des ordonnées porte les fréquences et la nuance de gris code l'amplitude.

Le premier spectrogramme (a) a été obtenu grâce à l'enregistrement d'une femme par un microphone proche de la bouche. On peut donc considérer que cet enregistrement est *propre*, c'est à dire que seul le signal de parole a été capté. Le spectrogramme (c), quant à lui, a été obtenu par un microphone placé sur le rétroviseur, près du pare-brise. Par rapport au spectrogramme précédent, le signal de parole est corrompu par un bruit additif (le bruit du moteur entre autres) et, dans une moindre mesure, par un bruit convolutif (puisqu'on n'utilise pas le même microphone qu'à l'entraînement).

Le signal de bruit est visible en continu sur les basses fréquences : c'est une caractéristique du bruit à l'intérieur de l'habitacle d'une automobile. Ce sont donc les indices de parole situés dans le bas du spectre qui vont être recouverts par le spectre de bruit. On peut voir par exemple que la partie basse du spectre du phonème de transition **[H]** se confond avec un bruit de fond continu comme ici et que les passages entre unités acoustiques élémentaires sont généralement moins bien définis dans le bruit. Le spectrogramme (b) a été obtenu dans les mêmes conditions que (a) mais pour un locuteur différent (un homme). Bien que le mot prononcé soit le même, les spectrogrammes diffèrent beaucoup. On voit dans le spectrogramme (d) (locuteur masculin) que les zones de bruit recouvrent par endroits les zones de parole.

En première analyse, il n'est donc pas aisé de séparer les composantes de bruit de celles de parole. On peut par contre évaluer la puissance du bruit dans le spectre de puissance du signal. On obtient alors le rapport Signal à Bruit ou RSB (SNR, *Signal to Noise Ratio*). Les façons de l'obtenir sont très variées. La méthode la plus courante est d'évaluer le spectre de puissance dans les zones de non-parole et de parole, puis de prendre le log décimal du rapport. Les RSB s'expriment donc en décibels (dB) et décroissent lorsque la composante de bruit est plus présente :

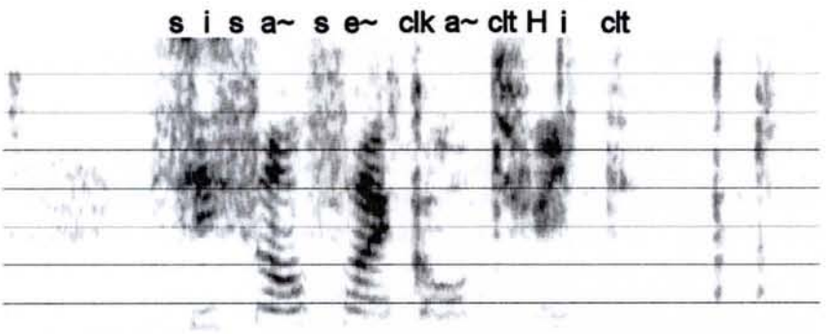
$$RSB = 10 * log10 \frac{P_s}{P_n}$$

avec :

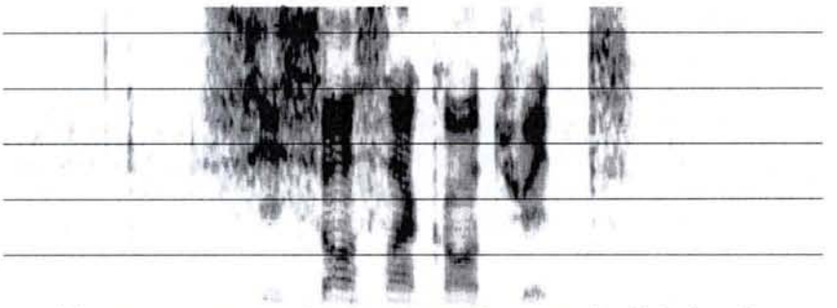
- P_s = puissance du signal
- P_n = puissance du bruit (de la non-parole)

Les zones de parole et de non parole sont déterminées en comparant l'amplitude du signal par rapport à un seuil.

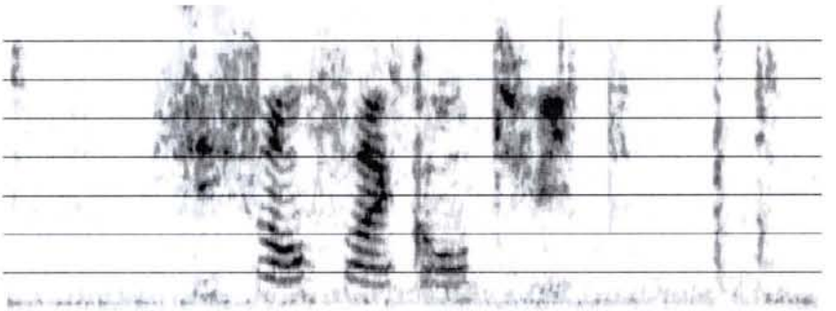
Le RSB ne constitue qu'un moyen de quantifier l'influence du bruit sur la parole. Mais il ne permet pas de qualifier complètement la difficulté qu'un environnement bruité représente du point de vue de la reconnaissance. Par exemple, il ne permet pas de rendre compte de la



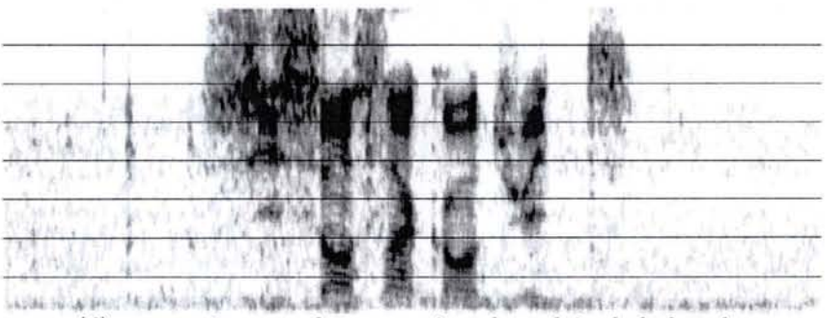
(a) prononcé par une femme, microphone proche de la bouche



(b) prononcé par une homme, microphone proche de la bouche



(c) prononcé par une femme, microphone loin de la bouche



(d) prononcé par un homme, microphone loin de la bouche

FIG. 2.1 – Spectrogrammes de "658" prononcés dans l'habitacle d'une voiture en mouvement.

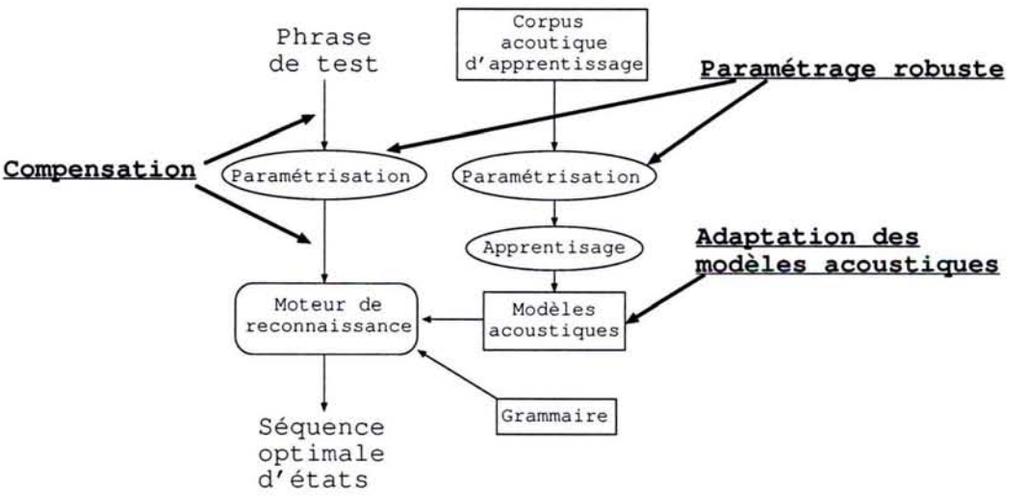


FIG. 2.2 – Points d'intervention des solutions de robustesse.

dynamique du bruit, de la façon dont il varie sur une courte période. En effet, un bruit (additif ou convolutif) peut être stationnaire, évoluer lentement ou encore inopinément.

Il est cependant possible de rendre compte des plages de spectres influencées par le bruit en déterminant des RSB par bandes de spectres. Ainsi, il est possible de voir quels types d'indices acoustiques seront corrompus par un environnement particulier. Cela est particulièrement utile dans les méthodes de compensation du type *Missing Data*, exposée en section 2.2.1.

2.1.3 Objectifs de ce chapitre

Nous présentons dans ce chapitre quelques techniques classiquement utilisées pour rendre les SRAP robustes aux distorsions (la liste n'est pas exhaustive). Il n'est pas facile de classer ces méthodes car elles font appel à des notions diverses. Nous avons choisi, dans ce document, de les classer selon leur espace d'application :

- compensation du signal bruité afin de réduire l'influence du bruit
- paramétrisation de la parole à l'aide de méthodes robustes
- adaptation des modèles acoustiques du SRAP aux nouvelles conditions acoustiques

Ce découpage, a été utilisé dans [Gong, 1995] et a souvent été repris, même s'il peut sembler artificiel car souvent certaines méthodes (comme le *Stochastic Matching*[7]) peuvent être classées dans plusieurs catégories.

Les domaines d'intervention des trois classes de méthodes sont représentés dans la figure 2.2.

Toutes les méthodes de robustesse ne sont pas évoquées ici. Nous présentons en priorité les méthodes qui ont inspiré nos travaux et les méthodes les plus classiques, éprouvées et reconnues pour améliorer le taux de reconnaissance de SRAP en milieu bruité.

Nous n'évoquerons pas en profondeur les techniques de robustesse basées sur la paramétrisation car nous pensons que ces techniques sortent du domaine d'étude de cette thèse. Nous

[7] voir une description complète du *Stochastic Matching* au chapitre 3

pouvons simplement dire que la robustesse d'un SRAP peut être fortement améliorée dans le cas où la paramétrisation du signal de parole et la mesure de similarité associée sont robustes aux variations des conditions d'environnement. La représentation du signal de parole étant supposée indépendante du bruit, un SRAP entraîné sur de la parole propre peut alors être utilisé dans un milieu calme ou bruyant. L'avantage d'une telle approche est qu'aucune hypothèse sur la nature du bruit n'est nécessaire. Ceci peut cependant constituer un handicap puisqu'on ne tire pas parti de la forme particulière du signal de bruit. La plupart des travaux portant sur ce domaine font appel, d'une part à des techniques de traitement du signal et d'estimation statistique, d'autre part à des données issues des études relatives à la perception auditive.

Les techniques de robustesse se plaçant dans le cadre du *Stochastic Matching* feront l'objet d'une étude approfondie dans le chapitre suivant (chapitre 3).

2.2 Compensation du signal bruité

Les méthodes de débruitage regroupés sous cette dénomination tentent de retrouver les données émises à partir des observations bruitées, avant l'étape de reconnaissance.

Ces techniques tirent parti de diverses informations *a priori* sur le signal de parole et sur l'environnement dans lequel il a été émis. Elles s'attachent à obtenir une *fonction de compensation* permettant de transformer la séquence d'observations bruitées en une séquence de vecteurs paramétriques proches de la séquence de parole propre.

Les méthodes évoquées dans cette section sont classées selon la nature des informations dont elles disposent pour faire la compensation. En effet, comme nous l'avons vu au début de ce chapitre, le but de la compensation est de réduire la différence existant entre espaces de test et d'entraînement. Pour réduire cet écart, une méthode de compensation doit disposer d'un système permettant de comparer les données observées pendant le test avec celles observées pendant la phase d'entraînement. Certaines méthodes devront donc disposer

- de bases de données stéréo (enregistrement simultanés dans deux environnements différents) (section 2.2.2)
- de modèles de la parole propre (section 2.2.3)
- de modèles de la fonction d'environnement (section 2.2.4)

D'autres encore (plus simples) chercheront à s'affranchir d'une telle connaissance (section 2.2.1) et donc se passeront de données de comparaisons.

2.2.1 Absence de données de comparaison

Les méthodes recensées dans cette section n'utilisent aucune information *a priori* sur les valeurs des données de test. Elles n'utilisent pas non plus de modélisation du signal de parole propre ou corrompue. Par conséquent, elles ne disposent d'aucun moyen de comparaison entre données de test et d'apprentissage.

Soustraction spectrale

La soustraction spectrale consiste à retrancher du spectre du signal de parole une estimation du spectre de bruit qui le corrompt. Cette approche fonctionne efficacement lorsque la corruption est additive et le signal de bruit est stationnaire et de faible puissance.

Dans le domaine temporel le signal $x(t)$ corrompu par un bruit additif $n(t)$ donnera une observation $y(t)$ telle que $y(t) = x(t) + n(t)$. Les transformées de Fourier $Y(\omega)$, $X(\omega)$ et $N(\omega)$ de $y(t)$, $x(t)$ et $n(t)$ sont alors liées par :

$$|Y(\omega)|^2 = |X(\omega)|^2 + |N(\omega)|^2 + X(\omega) \,.\, \bar{N}(\omega) \; + \; N(\omega) \,.\, \bar{X}(\omega) \tag{2.1}$$

où $\bar{N}(\omega)$ et $\bar{X}(\omega)$ sont les conjugués de $N(\omega)$ et $X(\omega)$.

Dans l'hypothèse où le bruit est stationnaire et non-corrélé avec la parole, on obtient une estimée du spectre de parole par :

$$|\hat{X}(\omega)|^2 = |Y(\omega)|^2 - E[|N(\omega)|^2] \tag{2.2}$$

où $E[|N(\omega)|^2]$ est obtenu en effectuant une moyenne sur les zones de non-parole. Ainsi, puisqu'il a été prouvé que la phase $\Phi_{Y(\omega)}$ du signal de parole avait peu d'influence sur la perception, on peut reconstituer une estimation du signal non-bruité :

$$\hat{x}_n = IDFT[|\hat{S}(\omega)|].exp(j.\Phi_{Y(\omega)}) \tag{2.3}$$

où $IDFT$ est l'inverse de la transformée de Fourier.

L'utilisation de cette approche, bien qu'éprouvée ([Boll, 1979] est l'un des premiers à le proposer et [Okazaki *et al.*, 2004] l'un des derniers à l'améliorer) est soumise à des conditions qui sont autant d'axes de recherche :

1. La soustraction spectrale demande un détecteur de parole/non-parole robuste. En effet, l'efficacité de la soustraction spectrale est fortement conditionnée par la qualité de l'estimation de la densité spectrale de la puissance de bruit.
2. La soustraction peut entraîner un spectre négatif, ramené à un seuil mais les sursauts dans le spectre que cette opération engendre sont à l'origine de sons purs de fréquence aléatoire : c'est le bruit musical.
3. La soustraction spectrale ne peut se faire dans le domaine cepstral où le bruit devient corrélé.

Malgré ces handicaps, la soustraction spectrale est une méthode largement implémentée dans les SRAPs, comme première étape de compensation. Elle permettent ainsi de réduire l'influence du bruit additif avant l'étape de paramétrisation. De plus, en réestimant régulièrement le spectre de bruit, il est possible de compenser pour un bruit variant lentement.

Normalisation de la moyenne cepstrale

De nombreux types de bruits additifs et convolutifs varient peu en fonction du temps en comparaison des variations du signal de parole. Filtrer les variations lentes permet donc d'améliorer les scores de reconnaissance significativement.

La manière la plus simple et la plus efficace d'enlever ces variations lentes est la normalisation cepstrale (CMN, *Cepstral Mean Normalisation* ou MCR, *Mean Cepstre Removal*). La compensation s'effectue alors comme suit. A l'apprentissage des modèles acoustiques, les séquences de cepstres d'entraînement sont normalisées par leur moyenne sur chaque phrase ([Furui, 1981], [Mokbel *et al.*, 1994]). Au cours de la reconnaissance, après l'étape de paramétrisation et avant l'étape de reconnaissance, une moyenne des cepstres est calculée itérativement à chaque instant

et est retirée au vecteur d'observation. Dans [Rahim and Juang, 1996], on précise que le calcul itératif de la moyenne temporelle $\hat{\mathbf{c}}_t$ du cepstre se fait selon :

$$\hat{\mathbf{c}}_t = \alpha \,.\, \hat{\mathbf{c}}_{t-1} \; + \; (1-\alpha) \,.\, \mathbf{c}_t$$

où α est un facteur d'oubli (compris entre 0 et 1) généralement fixé à 0.99. On parlera ici de normalisation séquentielle de la moyenne cepstrale (S-CMN, *Sequential Cepstral Mean Normalisation*). CMN nécessite que les valeurs de ces moyennes ($\hat{\mathbf{c}}_0$) soient initialisées (à partir de valeurs obtenues sur un corpus de développement bruité, par exemple) pour faciliter la convergence de la moyenne.

On remarque facilement que CMN est particulièrement bien adapté pour réduire l'influence d'un bruit convolutif (dû au canal de transmission). En effet, si l'on considère le signal de parole $x(t)$, $h(t)$ la réponse impulsionnelle du canal de transmission et $y(t)$ le signal obtenu à la sortie du canal,

$$y(t) = x(t) \otimes h(t)$$

donne, dans le domaine cepstral :

$$\mathbf{y}(i) = \mathbf{x}(i) + \mathbf{h}(i)$$

Si l'on considère que l'espérance de la variable aléatoire associée à la parole propre est nulle et que les caractéristiques du canal sont constantes, on peut dire que :

$$E[\mathbf{y}(i)] = \mathbf{w} \text{ et que } \mathbf{w} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{y}(i)$$

Il est alors possible de retirer la composante de bruit convolutif du signal dans le domaine cepstral, grâce à cette moyenne à long terme. Le processus reposant sur le calcul d'une moyenne, il ne peut compenser des variations rapides de bruit mais convient parfaitement pour une évolution lente. Par exemple, dans [Kermorvant, 1999], CMN augmente le taux de reconnaissance de nombres prononcés par téléphone de 24.6% par rapport à un SRAP n'intégrant aucun processus de compensation.

Masquage de bruit

En présence d'un bruit de fond de large bande, les régions du spectre de faible énergie sont plus affectées que les zones à forte énergie. Dans [Klatt, 1976], les coefficients à la sortie du banc de filtres qui sont inférieurs à une valeur définie par un masque sont seuillés car on considère les parties du spectre en dessous de ce seuil comme associées au bruit. On évite ainsi les variations rapides et de faibles amplitudes qui ne contiennent en fait que des informations de bruit.

Cette méthode est contrainte par la nécessité de connaître la puissance du bruit et sa nature afin d'établir un masque efficace. De plus, elle n'est plus applicable à des RSB très bas. Le lecteur se rapportera à [Varga *et al.*, 1988] pour plus d'informations.

La technique de masquage de bruit a évolué en un groupe de techniques dites *Missing Data* [Cooke *et al.*, 2001] [Morris *et al.*, 1998]. Pour les techniques *Missing Data*, les données considérées comme bruitées sont abandonnées (*marginalization*) ou estimées (*data imputation*) [Raj *et al.*, 2001]. Dans le cas de la *marginalization*, les composantes de vecteurs ou les vecteurs entiers considérés comme corrompus ne seront pas utilisés pour la reconnaissance. En ce qui

concerne la *data imputation*, elle peut être assimilée, dans une certaine mesure, à la détermination d'une séquence de vecteurs de biais compensant chaque trame de parole. Dans les deux cas, la détermination des zones de spectre que l'on soupçonne être polluées (*unreliable*) est un problème ardu. Dans [Cooke *et al.*, 2001], on propose de calculer les RSB à courte durée sur des bandes de spectres pour les déterminer. Les RSB étant calculés sur une courte durée, le processus de compensation peut fonctionner avec un bruit (additif) variant lentement.

La plupart de ces techniques se placent dans le domaine spectral et imposent donc des modèles acoustiques estimés dans ce domaine. Dans [Van Hamme, 2004] on propose une approche permettant d'utiliser *Missing Data* dans le domaine cepstral qui s'intègre facilement dans la plupart des SRAP.

Dans tous les cas, les techniques de masquage de bruit sont utilisées dans le cas d'un bruit additif dans le domaine temporel.

2.2.2 Présence de données de comparaison exactes (données stéréo)

Les méthodes de transformation d'espace cherchent à définir une fonction permettant de retrouver le signal de parole propre à partir du signal bruité. Ces approches n'utilisent aucune modélisation du bruit ou de la façon dont il interfère avec la parole. Cependant, elles nécessitent de disposer à la fois de réalisations dans le domaine bruité et dans le domaine propre. Elles doivent disposer, de surcroît, d'une relation pouvant lier les réalisations des deux domaines.

On peut distinguer deux types de méthodes. Les algorithmes appartenant au premier type doivent disposer de bases (*codebook*) d'enregistrements de référence dans les environnements de test et d'entraînement. Elles sont constituées *avant* toute phase de reconnaissance. Les approches du deuxième type n'ont pas besoin de bases réalisées au préalable mais d'un ensemble de microphones enregistrant simultanément la scène auditive.

Utilisation de bases stéréo (codebook)

La fonction de correspondance est établie une fois que l'on possède une même réalisation dans le domaine d'entraînement et le domaine d'utilisation (corpus d'adaptation). Bien évidement, on cherche le plus souvent à utiliser un corpus d'adaptation le plus réduit possible. Cette réalisation peut être une phrase, un mot, un phonème ou un petit ensemble de vecteurs. Cette approche permet donc de prendre en compte toute différence entre conditions de test et d'apprentissage, et peut être étendue pour prendre en compte la variabilité issue d'un changement de locuteur ou de microphone.

Plusieurs familles de techniques basées sur cette approche ont été développées ; parmi elles on trouvera :

- l'établissement d'une transformation linéaire qui minimise l'erreur quadratique entre la référence (propre) et la transformée de la parole bruitée [Mokbel and Chollet, 1991].
- l'utilisation d'une correspondance une à une entre les vecteurs d'un dictionnaire de référence (*codebook*) de l'espace de référence et ceux de l'espace de test. Les dictionnaires sont obtenus par Quantification Vectorielle (VQ, *Vector Quantization*) à partir de corpus de parole représentatifs issus des deux espaces acoustiques. La correspondance peut être obtenue par DTW, par exemple. Dans [Treurniet and Gong, 1994], un vecteur bruité est exprimé comme une combinaison linéaire de vecteurs de base bruités, son homologue non bruité est alors déduit en exprimant sur la base non bruitée un vecteur ayant les mêmes coordonnées.

- l'estimation des transformations plus complexes, obtenues grâce à des réseaux de neurones [Tamura and Waibel, 1988], [Yuk *et al.*, 1999].

Utilisation de champs de microphones

Les méthodes de filtrages adaptatifs sont aussi appelés méthodes ANC (*Adaptive Noise Cancellation*). Contrairement à la soustraction spectrale qui permet de filtrer le bruit d'un signal avec un seul microphone, cette méthode nécessite l'utilisation de deux microphones pour supprimer le bruit. Le premier capte la parole (bruitée) et le deuxième, le bruit de fond. Le principe consiste donc à calculer, à partir de ces deux entrées, le filtre adaptatif permettant d'estimer le bruit corrupteur et de le supprimer au signal de parole bruitée. L'intérêt de cette méthode est bien sûr qu'elle ne nécessite aucune hypothèse *a priori* sur le bruit et peut s'appliquer à des bruits non-stationnaires.

Cette approche trouve de nombreuses applications, surtout dans le domaine de l'annulation d'écho [Gänsler and Benesty, 2000] et l'estimation de filtres de Kalman adaptatifs [Buchner and Kellermann, 2002].

Cependant l'efficacité de cette solution est sujette à des contraintes fortes, notamment en ce qui concerne la disposition des microphones et leur nombre. De plus, dans certains environnements comme l'habitacle d'une voiture la correspondance entre les différentes sources n'est pas évidente.

Filtrage optimum probabiliste

Dans le filtrage optimum probabiliste (POF, *Probabilistic Optimum Filtering*), la fonction de compensation utilisée est une fonction linéaire par morceaux. Cette approche a été développée dans [Neumeyer and Weintraub, 1994]. Il y est observé que dans le domaine cepstral, la relation entre deux éléments acoustiques enregistrés au même instant, l'un dans un environnement bruité et l'autre au calme, est non-linéaire. C'est pourquoi on y propose d'utiliser une fonction de compensation non-linéaire, obtenue par concaténation de transformations linéaires.

Chaque portion linéaire (ou filtre) est obtenue par l'optimisation d'une transformation entre deux espaces vectoriels. Le premier espace contient des paramètres acoustiques non-bruités. Il est découpé en région par quantification vectorielle (VQ). Le deuxième espace contient les mêmes vecteurs, mais bruités. Il est aussi divisé en secteurs.

Lors de la compensation, les filtres sont combinés de manière probabiliste de façon à former un filtre W. Les coefficients de pondération associés à chaque filtre dépendent de la probabilité qu'a une observation d'être issue du secteur de l'espace associé au filtre.

Voici le cadre formel. Soit $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$ une séquence de vecteurs acoustiques propres. Ces vecteurs acoustiques sont corrompus par une fonction d'environnement inconnue et donne la séquence d'observations $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$. Plaçons nous à l'indice de temps n : on cherche $\hat{\mathbf{x}}_n$, l'estimée de $\mathbf{x}_n$. Les espaces vectoriels mentionnés plus haut sont fragmentés en I régions distinctes. A chacune de ces régions est associée un filtre non-causal $\mathbf{W}_i$ (qui permet de prendre en compte les corrélations du signal sur plusieurs fenêtres successives). Pour obtenir ces coefficients, on cherche à minimiser l'erreur suivante (pour le filtre correspondant à la région i) :

$$\mathbf{e}_{n,i} = \mathbf{x}_n - \hat{\mathbf{x}}_{n,i} = \mathbf{x}_n - \mathbf{W}_i^T \mathbf{Y}_n$$

où

- $\mathbf{e}_{n,i}$ est l'erreur entre le vecteur acoustique de original $\mathbf{x}$ et son estimée $\hat{\mathbf{x}}$ pour région i
- $\mathbf{W}_i^T = [\mathbf{A}_{i,-p} \ \ldots \ \mathbf{A}_{i,-1} \ \mathbf{A}_{i,0} \ \mathbf{A}_{i,1}; \ldots \ \mathbf{A}_{i,p} \ \mathbf{b}_i]$
- $\mathbf{Y}_n^T = [y_{n-p}^T \cdots y_{n-1}^T y_n^T y_{n+1}^T \cdots y_{n+p}^T 1]$

On obtient la transformation globale et l'estimée du signal propre par :

$$\hat{\mathbf{x}}_n = \left[\sum_{i=0}^{I-1} \mathbf{W}_i^T \, p(g_i|\mathbf{y}_n) \right] Y_n^T$$

où $p(g_i|\mathbf{y}_n)$ est la probabilité que x_n appartienne à la i-ème région sachant que le vecteur d'observation à l'indice de temps n est y_n.

Ce traitement peut se faire dans divers espaces, autres que l'espace cepstral. Dans [Neumeyer and Weintraub, 1994], le taux de reconnaissance sur les phrases de test du *Wall Street Journal* prononcé par téléphone suit une augmentation relative de 34% grâce à l'utilisation de POF dans le domaine cepstral. Cependant, le nombre de paramètres qu'il est nécessaire d'estimer pour obtenir la correspondance est très grand.

Enfin, on ne peut pas prévoir dans quel milieu bruité l'application va être utilisée (ou simplement quel micro va être employé). Une transformation valable pour un bruit ne l'est pas nécessairement pour un autre. Par conséquent, une série de transformations/filtres "génériques" va être entraînée (pour divers bruits, divers RSB) et l'une d'entre elles sera choisie au moment opportun, en temps réel ([Neumeyer and Weintraub, 1994], [Neumeyer and Weintraub, 1995]).

En conclusion, la POF ne peut être utilisée que dans le cas où l'on dispose de données stéréo, ce qui limite son cadre applicatif à des SRAP fonctionnant dans des environnements acoustiques bien définis. Cependant, POF permet de compenser aussi bien pour des bruits additifs que convolutifs. De plus, cette méthode met en avant l'intérêt d'une fonction de compensation non-linéaire, approchée par une fonction linéaire par morceaux.

2.2.3 Présence de données de comparaison modélisées

Cette catégorie d'approches met en évidence l'objectif de la plupart des méthodes de compensations : disposer d'un moyen de comparer des données similaires dans les espaces d'entraînement et de test afin d'élaborer une fonction permettant d'associer à tous point de l'un, un point de l'autre. Cet objectif peut être atteint de plusieurs manières. L'approche la plus directe est de disposer comme à la section précédente d'enregistrements stéréo. Une approche plus élaborée est de modéliser la parole dans l'espace d'entraînement, comme ici, et comme dans la méthode du *Stochastic Matching*[8].

Egalisation par histogramme

L'égalisation par histogramme (*histogram equalisation*) est une technique de normalisation non-linéaire. Elle modifie les paramètres acoustiques de test de sorte que leur densité de probabilité cumulée (CDF, *cumulative density function*) corresponde à la CDF des données d'entraînement (CDF *cible*). Ici, la correspondance entre les réalisations du domaine propre et celles du domaine bruité n'est pas directe, à l'opposé des techniques présenté à la section précédente. Les CDFs sont obtenues par recensement des données de test (ou d'entraînement). On les nomme

[8] voir chapitre 3

souvent *histogrammes*. Du fait qu'il faille calculer les CDFs, cette méthode n'est efficace que pour les bruits stationnaires.

Cette technique a été utilisée en RAP robuste à la fois pour normaliser les données dans le domaine spectral ([Hilger and Ney, 2001], [Moleau *et al.*, 2001], [Moleau *et al.*, 2003a], [Moleau *et al.*, 2003b]) et le domaine cepstral ([Obushi and Stern, 2003], [Segura *et al.*, 2002a], [de la Torre *et al.*, 2002]).

Dans [Moleau *et al.*, 2003b] et [Korkmazsky *et al.*, 2004], on n'utilise plus une seule CDF *cible* mais un ensemble de CDFs. Dans [Moleau *et al.*, 2003b], deux CDFs sont utilisés : l'une modélisant la parole et l'autre, les zones de silence. Dans [Korkmazsky *et al.*, 2004], une CDF par classes phonétiques est créée. Dans les deux cas, un histogramme *cible*, adapté à la phrase de test est construit par interpolation linéaire des histogrammes de classe. Les coefficients de la combinaison linéaire dépendent de la phrase de test. Dans le cas de [Moleau *et al.*, 2003b], par exemple, on associe un coefficient α à l'histogramme de parole et le coefficient $(1 - \alpha)$ à l'histogramme de silence. Le coefficient α correspond à l'estimation de la proportion de trames de parole dans la séquence de test. Cette estimation se fait sur les phrases ultérieurement prononcées par le locuteur [Moleau *et al.*, 2003b] ou sur des données collectées sur la phrase de test même [Korkmazsky *et al.*, 2004].

Les techniques d'égalisation par histogramme donnent de bons résultats en milieu bruité (de façon convolutive et additive) et peuvent être cumulées avec d'autres méthodes pour augmenter la robustesse de la RAP. Par exemple, dans [Segura *et al.*, 2002a] on propose de les utiliser après la soustraction spectrale. Dans [Korkmazsky *et al.*, 2004], il est montré que l'on peut réduire le taux d'erreur de reconnaissance en phonèmes de 49.52% à 44.85%, sur des données collectées dans l'habitacle d'une voiture. Toutefois, dans [Moleau *et al.*, 2001], il est précisé que si la normalisation est utilisée dans le domaine spectral, il est aussi possible de compenser pour des différences entre le canal d'entraînement et celui de test.

Estimation par maximum de vraisemblance d'un biais dans le signal

[Rahim and Juang, 1996] propose une approche qui, comme dans [Sankar and Lee, 1996], utilise une méthode itérative basée sur l'estimation par maximum de vraisemblance. Le but est, là encore, de minimiser les effets d'un environnement contaminant le signal propre. Il utilise un modèle de parole (en fait une collection de modèles) simple pour mesurer ces effets, par l'intermédiaire de la vraisemblance.

Il est important de noter ici que :
- Le biais calculé est enlevé au signal entre la phase de paramétrisation et celle de reconnaissance. Cette approche peut être utilisée pour tout type de paramétrisation.
- Le modèle de parole utilisé pour son calcul n'est pas lié aux modèles acoustiques. D'une manière générale le processus de compensation est indépendant du processus de reconnaissance.

La figure 2.3 représente le mécanisme de compensation dans le processus de reconnaissance.

Le but est ici aussi de maximiser la vraisemblance par rapport au dictionnaire de modèles de parole $\Lambda = \{\lambda_1, \ldots, \lambda_I\}$ de la séquence d'observations $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$.

$$p_X(\mathbf{X}|\Lambda) = \prod_{t=1}^{T} \max_i p(\mathbf{x}_t|\lambda_i)$$

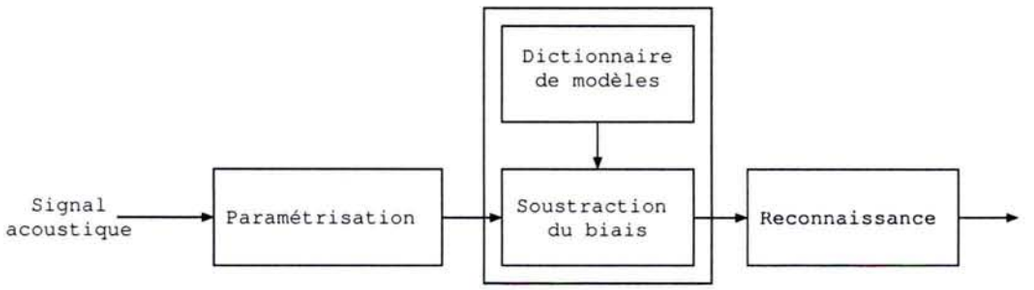


FIG. 2.3 – Soustraction du biais dans la méthode de Rahim.

Si on pose comme hypothèse que la séquence de parole est corrompue par un biais additif,

$$\mathbf{y}_t = \mathbf{x}_t + \mathbf{b}$$

alors, la probabilité de la séquence d'observations corrompues $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$ a pour probabilité :

$$p(\mathbf{Y}|\mathbf{b}, \Lambda) = p_X(\mathbf{Y} - \mathbf{b}|\Lambda)$$

La vraisemblance peut s'exprimer en fonction du biais $\mathbf{b}$:

$$p(\mathbf{Y}|\mathbf{b}, \Lambda) = \prod_t \max_i p(\mathbf{y}_t - \mathbf{b}|\lambda_i) \tag{2.4}$$

et dans ce cas, l'estimée au sens ML du biais correcteur $\mathbf{b}$, $\hat{\mathbf{b}}_{ML}$ est :

$$\hat{\mathbf{b}}_{ML} = \arg\max_{\mathbf{b}} p(\mathbf{Y}|\mathbf{b}, \Lambda)$$

Dans le cas où les modèles λ_i sont de type monogaussien,

$$p(\mathbf{y}_t - \mathbf{b}, \lambda_i) = A_i \, exp\left\{-\frac{1}{2}\left[(\mathbf{y}_t - \mathbf{b} - \mu_i)^T \mathbf{\Sigma}_i^{-1} (\mathbf{y}_t - \mathbf{b} - \mu_i)\right]\right\}$$

où

- A_i est une constante.
- μ_i est la moyenne du modèle (monogaussien) λ_i
- $\mathbf{\Sigma}_i$ est la matrice de covariance du modèle (monogaussien) λ_i

Sous l'hypothèse d'indépendance des dimensions, $\mathbf{\Sigma}_i$ est diagonale. Dans la suite, nous développons les calculs dans le cas mono-dimensionnel :

$$p(y_t - b, \lambda_i) = A_i \, exp\left\{-\frac{(y_t - b - \mu_i)^2}{2}\right\}$$

Alors, la valeur optimale du biais peut être obtenue par EM.

1. Phase d'estimation : l'estimée z_t à l'instant t est la moyenne du modèle λ_{i_t} :

$$\mathbf{z}_t = \mu_{i_t}$$

où l'indice optimal i_t à l'instant t est défini par :

$$i_t = \arg\max_j p(y_t - b|\lambda_j)$$

Dans ce cas, la fonction de vraisemblance (équation 2.4) devient :

$$p(Y|b,\Lambda) = A\ exp\left\{-\sum_{t=1}^{T}\frac{(y_t - b - z_t)^2}{2}\right\} \tag{2.5}$$

2. Phase de maximisation : la dérivée partielle de la vraisemblance (équation 2.5) par rapport à b en z_t est anulée :

$$\frac{\partial p(Y|b,\Lambda)}{\partial b} = K'\sum_{t=1}^{T}(y_t - b - z_t)exp\left\{-\sum_{t=1}^{T}\frac{(y_t - b - z_t)^2}{2}\right\}$$

on obtient alors l'estimée $\hat{b}$ du biais par :

$$\sum_{t=1}^{T}(y_t - \hat{b} - z_t)exp\left\{-\sum_{t=1}^{T}\frac{(y_t - \hat{b} - z_t)^2}{2}\right\} = 0$$

soit :

$$\hat{b} = \frac{\sum_{t=1}^{T}(y_t - z_t)}{T}$$

Plusieurs itérations des phases d'estimation et de maximisation assurent une convergence vers une valeur de biais maximisant (localement) la vraisemblance. Le biais est alors retiré de la séquence d'observations. La séquence d'observations, ainsi débruitée est ensuite fournie au SRAP pour être reconnue.

[Rahim and Juang, 1996] propose de réestimer les paramètres $\lambda_i = (\mu_i, \Sigma_i)$ toutes les n itération de EM, par VQ. Cette étape est très lourde en calcul. Pour l'évaluation de son algorithme, [Rahim and Juang, 1996] utilise 4 cycles de 20 itérations EM suivis par une segmentation VQ avant de ne plus observer de convergence dans la valeur du biais.

On peut voir que le calcul du biais correctif est inadapté à un bruit variant au cours de la phrase à reconnaître. Cette approche est donc intéressante pour normaliser un bruit de canal mais pas dans le cas d'un bruit additif non-stationnaire.

Une estimation séquentielle du biais peut être obtenue afin de faire une compensation synchrone à la trame :

$$\hat{\mathbf{b}}_t = \frac{t-1}{t}\hat{\mathbf{b}}_{t-1} + \frac{1}{t}(\mathbf{y}_t - \mathbf{z}_t)$$

et le vecteur corrigé à l'instant t est :

$$\hat{\mathbf{x}}_t = \mathbf{y}_t - \hat{\mathbf{b}}_t$$

Les méthodes exposées dans [Rahim and Juang, 1996] sont testées pour la reconnaissance de chaînes de chiffres enregistrées sur des lignes téléphoniques. Il y est montré que les performances obtenues par l'estimation séquentielle de biais proposée sont comparables à celles observées avec la méthode exposée plus haut (non-synchrone à la trame). Dans ce même cadre applicatif, les résultats obtenus sont comparables à ceux obtenus par normalisation cepstrale séquentielle (S-CMN)[9].

[9] voir la section 2.2.1

2.2.4 Présence d'une modélisation de la fonction d'environnement

Ces méthodes sont parfois catégorisées sous la dénomination *Model Based* car elles font appel à une modélisation de l'interaction de l'environnement acoustique avec les vecteurs d'observation.

Normalisation cepstrale par dictionnaire (CDCN)

CDCN, comme son nom l'indique, est appliqué dans le domaine cepstral. Il compense à la fois pour la distorsion convolutive et pour la distorsion additive.

Lors de la phase d'entraînement du SRAP, un modèle de parole général (un GMM) est construit en plus des modèles acoustiques (HMMs) destinés au processus de reconnaissance. Ce modèle est utilisé pendant la phase de test pour estimer les composantes convolutives et additives de la fonction d'environnement. Les cepstres sont alors compensés grâce à ces estimées et la séquence de cepstres obtenue est utilisée pour la reconnaissance.

Le principe est le suivant. Intéressons nous à la fonction d'environnement, dans le domaine spectral : le signal de parole $x(t)$ passe par un filtre convolutif $h(t)$. Au signal résultant s'ajoute un bruit additif $n(t)$ pour donner un signal corrompu $y(t)$.

$$y(t) = x(t) \otimes h(t) + n(t)$$

Dans le domaine cepstral, on obtient donc :

$$\mathbf{y} = \mathbf{x} + \mathbf{h} + \mathbf{r}(\mathbf{x}, \mathbf{n}, \mathbf{h}) \tag{2.6}$$

et

$$\mathbf{r}(\mathbf{x}, \mathbf{n}, \mathbf{h}) = IDFT\left\{log\left(1 + e^{DFT(\mathbf{n}-\mathbf{h}-\mathbf{x})}\right)\right\}$$

Avec $\mathbf{x}$, $\mathbf{y}$, $\mathbf{n}$ et $\mathbf{h}$: les équivalents cepstraux de $x(t)$, $y(t)$, $n(t)$ et $h(t)$ et où $IDFT$ représente l'inverse de la fonction de Fourier discrète. La méthode CDCN propose ensuite de modéliser le signal de parole par un GMM de K gaussiennes (de moyennes $\mathbf{c}_k$ et matrice de covariance $\mathbf{C}_k$) :

$$p(\mathbf{x}) = \sum_{k=0}^{M-1} p_k \mathcal{N}(\mathbf{x}; \mathbf{c}_k; \mathbf{C}_k) \tag{2.7}$$

Cette modélisation est obtenue durant la phase d'apprentissage, par Quantification Vectorielle par exemple.

Sous l'hypothèse que le RSB ne soit pas trop faible, il est montré dans [Acero and Stern, 1990] que l'on peut obtenir une approximation $\hat{\mathbf{x}}_{MMSE}$ de l'estimée MMSE de x qui s'exprime sous la forme :

$$\hat{\mathbf{x}}_{MMSE} = \sum_{k=0}^{M-1} p_k \, f_k \, \hat{\mathbf{x}}_k$$

avec

$$\hat{\mathbf{x}}_k = \mathbf{y} - \mathbf{h} - \mathbf{r}(\mathbf{c}_k, \mathbf{n}, \mathbf{h})$$

et f_k, un terme dépendant de $\hat{\mathbf{x}}_k$. $\mathbf{r}(\mathbf{c}_k, \mathbf{n}, \mathbf{h})$ ne dépend plus des observations mais des moyennes du modèle de parole : c'est pourquoi cette méthode est dite *codeword-dependent*, c'est-à-dire dépendant de centroïdes prédéterminés. A la fin de cette étape, on a donc obtenu une estimation

d'un vecteur de cepstre débruité $\hat{\mathbf{x}}_{MMSE}$ à partir du vecteur d'observation $\mathbf{y}$ et des composantes de bruit $\mathbf{h}$ et $\mathbf{n}$.

Comme aucune information *a priori* n'est disponible sur $\mathbf{h}$ et $\mathbf{n}$, CDCN utilise une approximation de type ML de ces composantes :

$$(\hat{\mathbf{n}}_{ML}, \hat{\mathbf{h}}_{ML}) = \arg\max_{\mathbf{n},\mathbf{h}} p(\mathbf{y}_0, \ldots, \mathbf{y}_T | \mathbf{h}, \mathbf{n})$$

Grâce à l'équation 2.6 reliant $\mathbf{x}$ à $\mathbf{y}$, $\mathbf{n}$ et $\mathbf{h}$, il est possible d'exprimer $p(\mathbf{y}_0, \ldots, \mathbf{y}_T | \mathbf{h}, \mathbf{n})$ en fonction des paramètres de la modélisation de $\mathbf{x}$ donnée en équation 2.7. La relation obtenue n'étant pas linéaire, il n'est pas facile d'obtenir directement les valeurs de $\mathbf{h}$ et $\mathbf{n}$ maximisant le vraisemblance. Cependant, en adoptant une approche EM, on obtient des estimées des composantes de bruit convolutif et additif.

Grâce à cette approche, le SRAP peut s'adapter à de nouveaux locuteurs, microphones, et de nouveaux environnements sans avoir à collecter d'information sur ces conditions. Par exemple, il est montré en [Acero and Stern, 1990] que CDCN permet d'augmenter considérablement les performances du système SPHINX lorsque les modèles acoustiques sont entraînés dans le calme et que le test s'effectue dans un environnement de bureau, avec un autre microphone.

Dans [Acero and Stern, 1990], une expérience est menée pour une tâche de reconnaissance de séries de lettres, nombres et de mots de contrôle dans un environnement de bureau. Les performances obtenues par CDCN sur de la parole enregistrée par un microphone placé sur le bureau (*far-talk*) par des modèles entraînés sur de la parole enregistrée sur un micro-casque (*close-talk*) sont comparables à celles obtenus lorsque l'entraînement et le test se font sur de la parole *far-talk*.

L'avantage de CDCN est qu'il ne pose aucune hypothèse sur la nature de l'environnement acoustique. Par contre, il est nécessaire de connaître l'influence des composantes du bruit sur la parole. Cela revient à disposer d'une approximation de la fonction d'environnement et d'une modélisation de la parole propre. De plus, l'efficacité de CDCN décroît lorsque le RSB décroît car l'estimation des paramètres de bruit devient plus ardue. Enfin, CDCN n'est pas adapté pour des bruits non-stationaires.

Le temps de calcul associé à CDCN a été jugé trop important pour le concevoir dans une application temps-réel. Des approximations sont donc nécessaires. Ces modifications ont abouti à un ensemble de méthodes décrites dans [Liu *et al.*, 1994], que nous nous contentons d'évoquer ici :

SDCN (*Snr Dependent Cepstral Normalization*) où la fonction de compensation, issue d'un *code-book* dépend du niveau de bruit.

Fixed CDCN , plus rapide que CDCN mais reposant sur la collecte de données stéréo

Phone-DCN , où la segmentation de l'espace cepstral ne se fait plus par VQ mais par classe de phonèmes.

Approximation de la fonction d'environnement par un développement de Taylor

L'approximation de la fonction d'environnement par un développement de Taylor (VTS, *Vector Taylor Series*) est une méthode de compensation qui repose sur une approximation en série de Taylor de la fonction d'environnement ([Moreno *et al.*, 1996], [Moreno, 1996], [Segura *et al.*, 2002b]). Pour cette raison, on peut dire que VTS offre une réponse analytique là où d'autres algorithmes proposent des approches basées sur l'empirisme. En effet, dans les

méthodes de compensation exposées jusqu'à présent dans ce document, la relation entre les données bruitées et une représentation des données propres est observée alors que dans VTS, cette relation est calculée.

Le deuxième point fort de cette méthode est la réduction du volume d'informations nécessaires à l'entraînement. Il se limite à la seule phrase à reconnaître.

Reprenons la représentation du vecteur bruité $\mathbf{y}$ par rapport au vecteur original non bruité $\mathbf{x}$, la composante de bruit additif $\mathbf{n}$ et la composance du bruit convolutif $\mathbf{h}$ dans le domaine log-spectral.

$$\mathbf{y} = \mathbf{x} + f(\mathbf{x}, \mathbf{h}, \mathbf{n})$$

avec f, la fonction d'environnement :

$$f(\mathbf{x}, \mathbf{q}, \mathbf{n}) = \mathbf{h} + \log\left(I + e^{\mathbf{n}-\mathbf{x}-\mathbf{q}}\right)$$

où

- $\mathbf{n}$ est la représentation dans le domaine cepstral du bruit additif.
- $\mathbf{q}$ est un paramètre inconnu représentant l'effet du bruit convolutif dans le domaine log-spectral.

L'algorithme VTS propose simplement de donner une approximation de f par un développement en série de Taylor.

$$\mathbf{y} = \mathbf{x} + f(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0) + \frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)(\mathbf{x}-\mathbf{x}_0) + \frac{\partial f}{\partial \mathbf{n}}(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)(\mathbf{n}-\mathbf{n}_0) + \frac{\partial f}{\partial \mathbf{q}}(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)(\mathbf{q}-\mathbf{q}_0) + \dots$$

où

- $f(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)$ est la valeur de la fonction d'environnement au point de fonctionnement $(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)$
- $\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)(\mathbf{x} - \mathbf{x}_0)$ représente la matrice des dérivées de la fonction d'environnement par rapport au vecteur $\mathbf{x}$), au point de fonctionnement $(\mathbf{x}_0, \mathbf{q}_0, \mathbf{n}_0)$

Le développement est tronqué au premier ordre, ce qui est une bonne approximation si on considère que $\mathbf{x}$ a une distribution gaussienne :

$$p(\mathbf{x}) = \sum_{k=0}^{K-1} w_k \mathcal{N}(\mathbf{x}; \mu_k^x, \mathbf{\Sigma}_k^x)$$

Si de plus on considère que $\mathbf{n}$ et $\mathbf{y}$ ont aussi des distributions gaussiennes,

$$p(\mathbf{n}) = \mathcal{N}(\mathbf{n}; \mu^n, \mathbf{\Sigma}^n)$$

et

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}; \mu^y, \mathbf{\Sigma}^y)$$

on peut obtenir les paramètres $\mu^y, \mathbf{\Sigma}^y$ et q à partir de la décomposition en série de Taylor par un algorithme proche de l'Estimation-Maximisation.

L'estimée $\hat{\mathbf{x}}$ du vecteur acoustique $\mathbf{x}$ est obtenu par MMSE. Lorsqu'on se limite à un développement d'ordre 0, on a :

$$\hat{\mathbf{x}} = \mathbf{y} - \sum_{k=0}^{K-1} p(k|\mathbf{y}) f(\mu_k^x, \mathbf{q}, \mu^n)$$

La compensation n'est pas parfaite : l'estimée $\hat{\mathbf{x}}$ n'est pas égale à $\mathbf{x}$ et la différence entre ces deux valeurs est mise en évidence dans [Segura *et al.*, 2002b]. Dans cette publication, on propose de réduire ce résidu par un filtrage linéaire par un procédé proche de l'égalisation par histogrammes. L'amélioration du taux de reconnaissance qu'apporte cette techniques par rapport à l'utilisation de VTS seul est particulièrement remarquable pour les faibles RSB (inférieurs à 10dB) car c'est dans cette zone que VTS est le moins performant. De plus, il y est montré que l'on peut obtenir une amélioration allant jusqu'à 33% du taux de reconnaissance moyen sur Aurora2 (bruit additifs stationnaires), par rapport à un SRAP n'intégrant aucun processus de compensation.

Dans [Kristjanson *et al.*, 2001], la densité de distribution de $\mathbf{n}$ est une somme pondérée de gaussiennes, les paramètres de chaque gaussienne étant déterminés en appliquant individuellement la méthode décrite plus haut pour chaque composante. Dans cette publication, l'approche est combinée avec la composition de modèles acoustiques[10]. L'introduction de données dynamiques dans la composition de modèles autorise un bon taux de reconnaissance pour un système grand vocabulaire et des données corrompues par un bruit additif non-stationnaire. On remarquera que des développements récents tendent à retarder l'estimation de la fonction de bruitage : dans [Jiang and Wang, 2004], ce n'est pas cette fonction que l'on estime, mais l'intégrale résultant de l'estimation MMSE de la donnée propre. L'intégrale est estimée grâce à une approche numérique. Dans ce cas et lorsque l'on teste cette nouvelle approche sur des phrases corrompues par un bruit blanc, on peut obtenir une réduction du taux d'erreur en phrase de 6.4% par rapport à l'approche VTS proposée par [Moreno *et al.*, 1996].

Approximation de la fonction d'environnement par décomposition polynômiale

L'approximation de la fonction d'environnement par décomposition polynomiale (VPS, *Vector Polynomial approximationS*) est une amélioration de VTS. Dans cette évolution, on n'utilise pas le même ordre de développement pour approximer les paramètres de la densité de probabilité du bruit et approximer la fonction d'environnement.

Considérons à nouveau la fonction d'environnement :

$$\mathbf{y} = \mathbf{x} + f(\mathbf{x} - \mathbf{n} - \mathbf{q})$$

soit

$$\mathbf{y} = \mathbf{x} + f(\nu) \text{ et } \nu = \mathbf{x} - \mathbf{n} - \mathbf{q}$$

f est strictement monotone, avec des asymptotes à $f(\nu) = 0$ en $-\infty$ et $f(\nu) = \nu$ en $+\infty$. Par conséquent, sa dérivée première (par rapport à ν) peut être considérée comme une fonction de densité. On remarque de plus que la dérivée seconde de f est en forme de cloche. VPS fait le choix d'approximer cette dérivée seconde par une fonction triangulaire. En intégrant deux fois, on obtient une approximation de la fonction d'environnement par un polynôme d'ordre 3. Par contre, la densité de probabilité du bruit,

$$p(\mathbf{n}) = \mathcal{N}(\mathbf{n}; \mu^n, \mathbf{\Sigma}^n)$$

est approximée par une distribution uniforme de même moyenne et variance [Raj *et al.*, 1997][Raj *et al.*, 1996].

[10] voir section 2.3 de ce chapitre pour la décomposition de modèles

On peut ensuite linéariser cette fonction :

$$f(\mathbf{x} - \mathbf{n} - \mathbf{q}) = A_k(\mathbf{x} - \mathbf{n} - \mathbf{q}) + B_k$$

en choisissant A_k et B_k qui minimisent l'erreur quadratique :

$$E(A_k(n - x - h) + B_k - f(\mathbf{x} - \mathbf{n} - \mathbf{q}))^2$$

Ceci donne un polynôme d'ordre 6 en $\mu_{x,k}$, k étant une des M composantes gaussiennes utilisées pour modéliser les vecteurs de parole. Une fois les paramètres A_k et B_k approximés, on peut retrouver les paramètres de la densité de $\mathbf{y}$, puis l'approximation du vecteur acoustique débruité. D'après [Raj *et al.*, 1996], cette méthode surpasse VTS et SNR-DCN, à tous RSB, lorsqu'un bruit blanc est ajouté au signal de parole propre.

Les méthodes VTS et VPS font partie, comme POF[11] des méthodes de compensation non-linéaires et confirment les avantages à tirer de ce genre d'approche.

2.3 Compensation des modèles en présence de bruit

Jusqu'à présent, nous avons vu des méthodes permettant de modifier les données bruitées pour en enlever les composantes dues au bruit. Dans les méthodes que nous allons présenter dans cette section, les données bruitées sont fournies au processus de reconnaissance sans modification et ce sont les modèles acoustiques du système qui vont être modifiés afin de se conformer aux conditions de test.

Pour ces méthodes, l'espace dans lequel s'effectue la compensation est l'espace des modèles acoustiques. Pourtant, les idées sous-jacentes aux approches exposées ici ne sont pas radicalement différentes de celles exposées dans la section précédente. Par conséquent, de nombreux algorithmes de compensation peuvent s'effectuer aussi bien sur les paramètres acoustiques que sur les paramètres des modèles acoustiques. Nous verrons par exemple dans le chapitre suivant que la méthode de *Stochastic Matching* peut tout aussi bien être utilisée pour la compensation des paramètres des modèles acoustiques et des paramètres acoustiques.

Il existe cependant des méthodes de compensation (on parlera aussi d'*adaptation*) spécifiques à l'espace des modèles. Ce sont ces méthodes que nous présentons dans cette section.

2.3.1 Composition/Combinaison de modèles

Cette catégorie de techniques exploite la capacité qu'ont les HMMs à modéliser la variabilité temporelle et spectrale d'un signal.

Comme nous l'avons vu, le signal de parole bruitée est la combinaison d'un signal de parole propre et d'un signal de bruit. Or ces deux signaux peuvent être modélisés par des HMMs. La composition de modèles propose donc de conduire simultanément une reconnaissance de ces deux signaux par des HMMs. Un modèle de bruit rendra compte de la vraisemblance d'une observation de bruit, un modèle de parole rendra compte de la vraisemblance de l'observation de parole au même instant. La combinaison des deux modèles en un nouveau modèle permet d'obtenir la vraisemblance d'une observation bruitée. Un simple décodage de Vitrebi permet d'établir un chemin (composite) optimum dans les deux modèles [Varga and Moore, 1990][Varga and Moore, 1991].

[11] voir section 2.2.2 pour une description de POF

Une nouvelle étape est franchie avec la PMC (*Parallel Model Combination*) [M.Gales and S.Young, 1992] [Gales, 1995]. Cette méthode permet de construire un HMM de parole bruitée à partir d'un HMM de parole propre et d'un HMM de bruit. Les paramètres du nouveau modèle s'expriment facilement à partir de ceux des modèles de parole et de bruit, selon une approximation décrite dans [Gales, 1995].

La figure 2.4 représente la combinaison d'un modèle de parole propre avec un modèle de bruit.

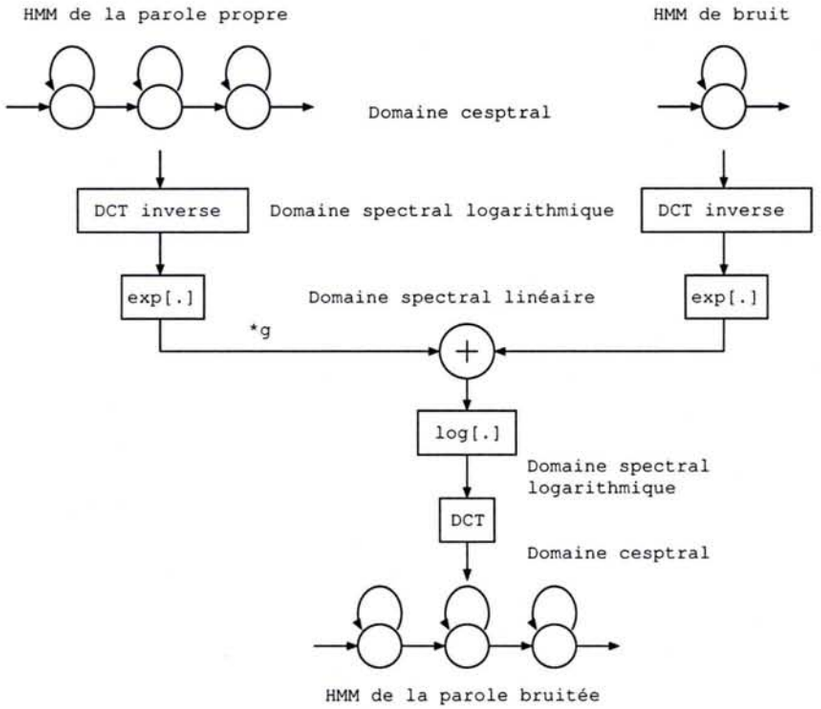


FIG. 2.4 – Principe de la composition de modèles (PMC).

La possibilité d'appliquer cette méthode pour prendre en compte des bruits non stationnaires est un atout important. En effet, il est possible de réestimer les modèles de bruits pendant les pauses de parole. Cela permet de prendre en compte les changements dans un environnement acoustique variant lentement. Dans l'implantation utilisée lors de nos expériences[12] les modèles acoustiques sont combinés à des HMMs estimés sur les premières trames de la phrase de test. Les expériences reportées dans la thèse de M.J.F.Gales ([Gales, 1995]) montrent qu'il est possible de faire chuter le taux d'erreur de 50% à 10% pour la reconnaissance petit vocabulaire sur la base NOISEX-92, pour des phrases corrompues à la fois par convolution et par bruit additif (RSB à 0dB) en n'utilisant que les 2 premiers mots pour adapter les modèles au nouvel environnement.

La difficulté principale de la composition de modèles est la détermination de la fonction de distribution de la probabilité (PDF, *Probability Distribution Function*) de la parole bruitée à partir des PDFs de parole propre et de bruit. L'approximation la plus courante est de considérer

[12]implantation faite par F.Chaffard, Dialoca

que le signal de parole bruité est égal au maximum du signal de parole et de la composante du bruit, dans le domaine cepstral [M.Gales and S.Young, 1992][Varga and Moore, 1991].

2.3.2 Filtrage par état

Plusieurs méthodes mettent à profit le fait que les HMMs découpent automatiquement le signal de parole en segments quasi-stationnaires, correspondant aux états des modèles. Dans [Beatie and Young, 1991], on propose par exemple un lissage des observations commandé par la séquence des états lors de la reconnaissance. Le but de cette méthode est d'obtenir un filtre optimal afin de convertir la parole corrompue en parole propre. Mais au lieu de modifier chaque vecteur paramétrique, on altère par un biais les moyennes de leur distribution dans les modèles acoustiques exprimés dans le domaine cepstral. En effet, dans le domaine cepstral, l'action d'un filtre de Wiener est additive.

On trouvera dans [Beatie and Young, 1992] les dérivations menant au calcul de ces filtres dépendants des états. Les filtres dépendent des états puisqu'ils utilisent le rapport entre les moyennes des états et les moyennes des observations attribuées à ces états.

L'intérêt de cette approche est qu'elle ne nécessite pas de conversion des modèles dans le domaine cepstral, contrairement à PMC. Seules les moyennes des modèles sont altérées, et les autres paramètres restent inchangés.

Cette technique donne des taux de reconnaissance très proches de ceux obtenus lorsque l'entraînement se fait dans les mêmes conditions que le test, pour la reconnaissance de chiffres dans un environnement automobile simulé avec un RSB de 0dB. Dans ces conditions, la soustraction spectrale est surpassée.

2.3.3 Adaptation des modèles de Markov cachés

Il a souvent été observé qu'un SRAP devant être utilisé dans un environnement de test connu et ne variant pas, atteint de meilleurs performances si ses modèles acoustiques sont entraînés dans l'environnement de test. Cette remarque tient même si l'environnement de test est bruité. Par conséquent, de nombreuses méthodes de robustesse ont pour objectif de modifier les modèles acoustiques propres en des modèles acoustiques proches de ceux qu'on aurait obtenu s'ils avaient été entraînés avec les conditions de test. C'est l'*adaptation* de modèles.

Les premières approches consistaient à faire subir une transformation linéaire aux paramètres des PDF des modèles acoustiques, la transformation étant apprise à partir de données *stéréo* [Mokbel, 1992].

La plupart des autres méthodes d'adaptation se basent sur le réentraînement des modèles à partir de corpus restreints représentatifs des conditions de test. Les paramètres des PDFs des nouveaux modèles (bruités) sont estimés à partir des paramètres des modèles propres.

Les deux méthodes principales pour atteindre cet objectif sont :

- la régression linéaire par maximum de vraisemblance (MLLR, *Maximum Likelihood Linear Regression*) [Leggetter and Woodland, 1995] : moyennes des PDFs propres sont transformées par une fonction simple pour founir les moyennes des PDFs bruitées. Cette méthode est l'une des méthodes les plus utilisées (et l'une des plus déclinée) parmi les méthodes d'adaptation des modèles acoustiques pour l'adaptation au locuteur.

- les méthodes *maximum a posteriori* (MAP)[Gauvain and Lee, 1994] : l'estimation au MAP de la transformation ou sa restriction a une structure donnée assurent une structure probabiliste stable des modèles acoustiques lorsque peu de données d'adaptation sont disponibles.

Les deux méthodes peuvent être combinées.

Comme montré dans [Mokbel, 2001], le choix d'une méthode d'adaptation particulière reposera sur la quantité de données d'adaptation à disposition. Dans cette publication, des expériences ont été menées sur la reconnaissance de chiffres prononcés sur des téléphones cellulaires (les modèles acoustique étant entraînés sur de la parole enregistrée sur téléphone fixe, 1000 locuteurs de toutes les régions de la France prononçant les chiffres). Le corpus d'adaptation est constitué par des phrases prononcées sur des téléphones cellulaires (1300 locuteurs, en voiture, à l'extérieur, ...) L'auteur montre que l'adaptation MLLR donne un taux d'erreur en mot de 1.9% pour un corpus d'adaptation 16 fois plus petit que le corpus d'entraînement. Pour le même corpus d'adaptation, MAP donne 2.5%. Inversement, pour un corpus d'adaptation de même taille que le corpus d'entraînement, c'est MAP qui donne les résultats les plus intéressants.

Le volume de données d'adaptation peut être réduit si les modèles acoustiques utilisés sont déjà "proches" des conditions de test. C'est pourquoi de nombreux SRAP utilisant l'adaptation comme méthode de robustesse disposent de collections des modèles entraînés dans des environnements bruités afin de les utiliser comme point de départ de l'adaptation. Dans [Zhang *et al.*, 2004], un arbre de modèles entraînés sur des données bruitées de différents types et à différents RSB est utilisé pour fournir une base à l'estimation de modèles adaptés aux conditions de test. L'arbre est parcouru à la recherche d'un modèle dont les données d'entraînement sont les plus proches des données de test. Ce modèle est alors adapté par MLLR.

Ces méthodes de robustesse peuvent se montrer très efficaces dans le cas où les environnements de test varient peu. En effet, l'effort à fournir pour modifier les modèles est souvent lourd et nécessite beaucoup de données d'adaptation comparativement au volume de données utilisé pour la reconnaissance. Nous verrons plus tard que pour débruiter une phrase, le *Stochastic Matching* n'utilise que les données qu'elle contient.

2.3.4 Apprentissage multiréférentiel

Une stratégie possible pour la reconnaissance de la parole dans le bruit consiste à entraîner les modèles dans le bruit. Ainsi, les différences entre conditions d'entraînement et de test seraient totalement éliminées.

Il n'est pas possible de prévoir quelles seront les conditions de bruit lors de la reconnaissance, aussi plusieurs méthodes reposent sur l'utilisation de collections des modèles entraînés sur une variété d'environnements différents. Une telle approche est également valable pour prendre en compte la variabilité du signal de parole provoquée par la variation du mode d'expression. Cependant cela nécessite une grande base d'entraînement contenant tout style d'élocution (criée, chuchottée, stressée, ...) qui est difficile à réunir. Les bases nécessaires à l'entraînement des modèles mulitréférentiels sont donc de préférence artificielles. Cependant, dans [Zhang *et al.*, 2004], il est signalé qu'il ne suffit pas de faire des modèles pour différents types de bruits mais aussi d'en créer pour différents RSB. En effet, des données bruitées à une certain RSB sont parfois plus proches de données corrompues par le même bruit à un RSB différent que de données corrompues par un bruit différent mais à un RSB équivalent. Comme les modèles appris sur une classe trop précise de locuteurs ou de bruits ont tendance à être moins discriminants, on cherchera

à regrouper les données d'apprentissage en larges classes. Par exemple regrouper les données d'apprentissage en classes de locuteurs, distinguer la parole sur de la musique, ...

Dans [Siohan *et al.*, 1995] et [Siohan, 1995], on rapporte que certaines méthodes de compensation de modèles permettent d'obtenir des taux de reconnaissance supérieurs à ceux obtenus lorsque test et apprentissage se font dans le même environnement. Aussi, on peut voir que les méthodes d'apprentissage multiréférentiel ne sont pas la panacée, quand bien même nous disposerions d'un ensemble d'entraînement très varié.

2.4 Conclusion

Nous avons vu quelques méthodes utilisées pour augmenter la robustesse des Systèmes de Reconnaissance Automatique de la Parole. Pour la plupart de ces méthodes, il est nécessaire de disposer d'une correspondance entre les données de test et les données d'entraînement. Certaines d'entre elles requièrent l'usage de bases stéréo, d'autres d'une modélisation des données de test et d'autres encore d'une modélisation de la fonction d'environnement. La collecte des données nécessaires à cette mise en correspondance est toujours un facteur limitant de la méthode car cette collecte suppose toujours une hypothèse *a priori* sur la nature de l'environnement ou le signal de parole : les bases de données stéréo doivent être collectées avant la phase de reconnaissance. Les méthodes de compensation utilisant ces bases ne fonctionnent donc pas correctement dans un milieu variant souvent et rapidement. Les bases de modèles de parole doivent, comme les bases stéréo, être élaborées avant la phase de test. De plus, elles doivent pouvoir produire un modèle de la parole prononcée dans l'environnement de test pour être efficace. Ceci suppose donc de limiter leur utilisation à des environnements connus et variant lentement.

Une méthode de compensation utilisée dans un environnement ne se conformant pas aux hypothèses *a priori* qu'elle s'est fixée n'est plus efficace. C'est pourquoi nous nous sommes efforçés dans nos recherches d'obtenir un processus de robustesse posant le minimum d'hypothèses sur la nature de l'environnement acoustique de test.

Le chapitre suivant porte sur une catégorie particulière d'algorithmes de robustesse que l'on appelle *Stochastic Matching*. Ces algorithmes permettent de compenser l'influence de la fonction d'environnement en se basant sur la comparaison de la séquence de test avec une modélisation des signaux utilisés pour l'entraînement. Le *Stochastic Matching* nécessite aussi la mise en correspondance entre données de test et d'entraînement pour développer une stratégie de compensation. Cette comparaison est stochastique et est fournie par l'étape de reconnaissance. Cette méthode s'affranchit donc de la création de bases stéréo ou de modèles et peut effectuer la compensation d'une phrase sans disposer d'information *a priori*.

Nous verrons que plusieurs approches ont été proposées et nous en exposerons les détails. Nous verrons aussi comment elles aboutissent sous certaines conditions à des implantations très semblables.

3

Stochastic Matching

Sommaire

Dans ce chapitre, nous avons regroupé un ensemble de techniques particulières de robustesse pouvant être associées à la dénomination *Stochastic Matching*. Comme les méthodes de compensation exposées au chapitre précédent, les méthodes de compensation par *Stochastic Matching* utilisent une mise en correspondance entre données de test et données d'entraînement. Ici, cette mise en correspondance est assurée par l'appariement stochastique de la séquence des observations bruitées avec une séquence de modèles. A partir de cet appariement, il est possible de mettre en place des processus de compensation de la séquence des observations bruitées.

3.1 Introduction

Comme nous l'avons vu dans le chapitre précédent, les systèmes de reconnaissance automatique de la parole voient leurs performances diminuer de manière significative lorsque les environnements dans lesquels ils ont été entraînés et ceux dans lesquels ils sont utilisés diffèrent. La différence entre les deux milieux est due à des sources de bruits extérieurs qui s'ajoutent au signal de parole, ainsi qu'à des variations dans le canal de transmission[13]. Ces perturbations extérieures au signal de parole le modifient d'une façon difficilement modélisable. Pourtant, le signal de parole, s'il n'était pas corrompu, serait reconnu par le SRAP car celui-ci utilise des modèles acoustiques entraînés sur de la parole propre. Mais l'influence des perturbations sur le

[13] voir le chapitre précédent

signal de parole introduit un écart entre le signal de parole corrompu et les modèles acoustiques utilisés par le SRAP.

Cet écart (ou *acoustic mismatch*) peut être réduit en transformant les données de test afin que leurs caractéristiques correspondent à celles des données d'entraînement (et donc aux modèles acoustiques). La forme fonctionnelle de la transformation dépendra d'une connaissance *a priori* sur l'influence de l'environnement acoustique sur la parole. Les méthodes de robustesse que l'on peut regrouper sous le nom de *Stochastic Matching* vont tirer parti d'une certaine modélisation (ou structure) de la parole, du bruit et de leur interaction [Surendran *et al.*, 1996]. Les paramètres des fonctions de transformation peuvent être estimés en maximisant la vraisemblance des vecteurs paramétriques en fonction de ces modèles. On peut donc dire que les méthodes de l'approche *Stochastic Matching* sont de type *model-based* comme les méthodes CDCN, VTS et VPS vues au chapitre précédent.

L'ensemble des paramètres étant généralement petit (moins nombreux que les paramètres des modèles acoustiques), leur estimation ne nécessite pas d'ensemble d'entraînement étendu. Cette adaptation de structure peut généralement se faire sur une phrase ou un petit ensemble de phrases.

Considérons la séquence de parole propre $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$. Le but de la reconnaissance est d'obtenir une séquence de mots[14] $W = \{W_1, \ldots, W_L\}$ rapportant l'information contenue dans $\mathbf{X}$. Or $\mathbf{X}$ est modifiée par l'environnement acoustique en une séquence $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$. On cherche à retrouver la séquence originale $\mathbf{X}$ à partir de $\mathbf{Y}$ par : $\hat{\mathbf{X}} = F_\nu(\mathbf{Y})$. Où ν est un ensemble de paramètres décrivant la transformation.

De même, dans le domaine des modèles, on peut considérer la fonction $\Lambda_Y = G_\eta(\Lambda_X)$ qui transforme les modèles acoustiques appris sur des données propres (Λ_X) en modèles acoustiques adaptés à l'environnement acoustique dans lequel est généré $\mathbf{Y}$.

Dans les deux cas, les méthodes de *Stochastic Matching* cherchent à faire décroître l'écart entre $\mathbf{Y}$ et Λ_X en déterminant les paramètres ν ou η et la séquence de mots W qui maximisent conjointement la vraisemblance de la séquence d'observations et de la séquence de mots. Cela se traduit, par la définition, dans l'espace des observations :

$$(\nu', W') = \arg\max_{(\nu, W)} p(\mathbf{Y}, W|\nu, \Lambda_X) \tag{3.1}$$

et dans l'espace des modèles :

$$(\eta', W') = \arg\max_{(\eta, W)} p(\mathbf{Y}, W|\eta, \Lambda_X) \tag{3.2}$$

Ce chapitre a pour but de présenter les contributions les plus importantes apportées dans le cadre du *Stochastic Matching*. L'algorithme de compensation proposé dans cette thèse s'appuie sur les développements qui y sont présentés. La première partie sera consacrée à l'approche par maximisation de la vraisemblance développée par Sankar et Lee. La seconde partie portera sur des approches moins populaires mais qui permettent d'élargir le cadre d'application du *Stochastic Matching*..

[14] ou de phonèmes ou d'unités acoustiques

3.2 Approche ML

[Sankar and Lee, 1995] présente une approche *Maximum Likelihood* du *Stochastic Matching*. Dans cette technique, on utilise l'ensemble des modèles acoustiques à la fois pour décoder la séquence des mots prononcés et pour estimer les paramètres d'une fonction de compensation. Cette approche du *Stochastic Matching* est intuitivement satisfaisante. En effet, le but étant de réduire l'écart entre les données de test $\mathbf{Y}$ et d'entraînement $\mathbf{X}$ pour augmenter la reconnaissance par des modèles Λ_X estimés sur $\mathbf{X}$, il est logique d'utiliser Λ_X pour calculer les paramètres de compensation (ou d'adaptation). Les paramètres de ces fonctions sont estimés en utilisant un algorithme d'Estimation-Maximisation (EM).

Dans la suite, nous donnerons les dérivations menant au calcul d'une fonction de compensation paramétrique simple dans l'espace des observations, telles qu'elles sont présentées dans [Sankar and Lee, 1995].

L'algorithme d'Estimation-Maximization [Dempster *et al.*, 1977] permet d'augmenter itérativement la vraisemblance.

$$(\nu', W') = \arg\max_{(\nu,W)} p(\mathbf{Y}, W|\nu, \Lambda_X)$$

ce qui correspond à

$$(\nu', W') = \arg\max_{(\nu,W)} p(\mathbf{Y}|W, \nu, \Lambda_X)P(W)$$

Pour cela, on fixe à tour de rôle ν et W comme montré dans la figure 3.2. Ce qui correspond à l'algorithme suivant :

1. en gardant ν fixe, maximiser sur W (opération classique de détermination de séquence optimale d'état)
2. en gardant W fixe, maximiser ν.

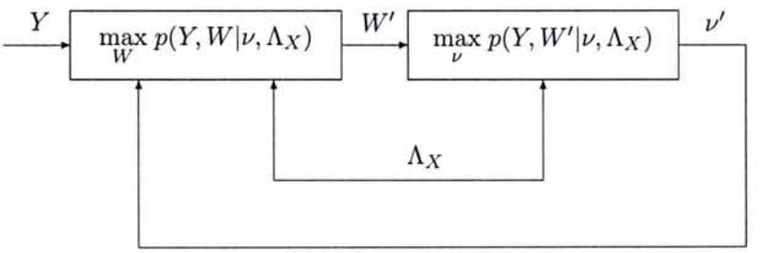


FIG. 3.1 – Procédure itérative pour maximiser conjointement la vraisemblance sur ν et W dans l'approche ML du *Stochastic Matching*.

La détermination d'une séquence de mots W maximisant la vraisemblance par rapport à une séquence d'observations et un ensemble de modèle est un problème classique de la reconnaissance et peut être résolu par l'utilisation de méthodes ML comme l'algorithme de Viterbi. Le reste de la discussion portera sur la maximisation par rapport à ν, la détermination de la séquence optimale de mots ayant été présentée dans le chapitre 1 :

$$\nu' = \arg\max_{\nu} p(\mathbf{Y}|\nu, \Lambda_X, W)$$

L'approche du *Stochastic Matching* considérée s'applique dans le cas de SRAP utilisant des HMMs continus. Comme vu au chapitre 1, ces HMMs contiennent des états et à chaque état est associé une distribution de probabilité définie par un mélange de composantes gaussiennes.

Considérons donc pour les modèles Λ_X[15],

- l'ensemble des états des modèles acoustiques : $\{s_1, s_2, \ldots, \}$
- l'ensemble des composantes gaussiennes de ces états : $\{c_1, c_2, \ldots\}$

Introduisons de plus les ensembles :

- $S = \{S_1, \ldots, S_p, \ldots\}$, l'ensemble de toutes les séquences d'états possibles pour Λ_X :

$$S_p = \{q_p(1), \ldots, q_p(T)\}$$

- $C = \{C_1, \ldots C_q, \ldots\}$, l'ensemble de toute les séquences de composantes gaussiennes des modèles acoustiques.

$$C_q = \{c_q(1), \ldots, c_q(T)\}$$

Nous avons pu voir, en étudiant le principe du décodage de Viterbi dans le chapitre 1 qu'une hypothèse de reconnaissance est représentée sous la forme d'une séquence d'états S_p. De même il est possible de représenter cette solution sous la forme d'une séquence de composantes gaussiennes C_q appartenant aux états.

La figure 3.2-(a) est une illustration de l'utilisation des notations introduites.

Sur cette figure est représentée une séquence d'états S_p issue de S. Dans ce cas,

$$S_p = \{q_p(1), \ldots, q_p(5)\}$$

avec

$$q_p(1) = s_3,\ q_p(2) = s_2,\ q_p(3) = s_1,\ q_p(4) = s_4,\ q_p(5) = s_5,$$

De même, la figure 3.2-(b) donne une illustration d'une séquence de composantes gaussiennes. Les composantes c_1, c_2 et c_3 sont issues de l'état s_1, etc. Sur cette figure est représentée une séquence de composantes gaussiennes C_q issue de C. Ici,

$$C_q = \{c_q(1), \ldots, c_q(5)\}$$

avec

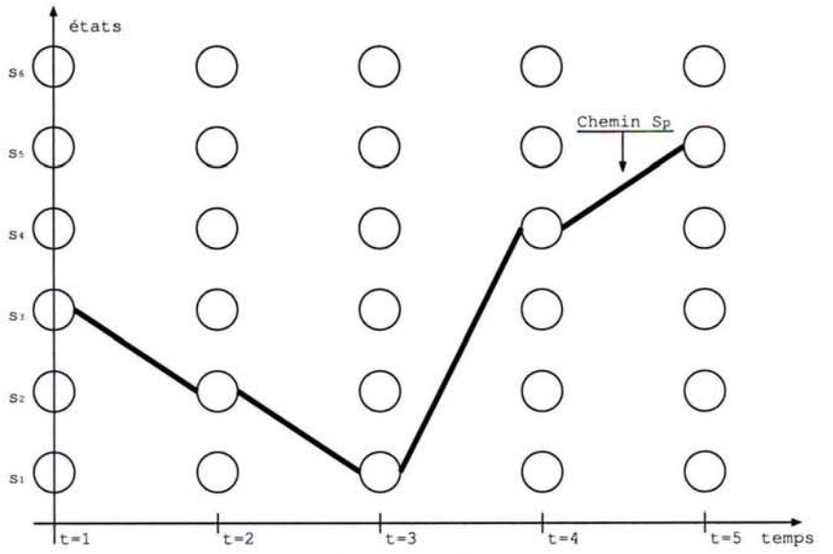
$$c_p(1) = c_3,\ c_q(2) = c_6,\ c_q(3) = c_3,\ c_q(4) = c_8,\ c_q(5) = c_5,$$

Par conséquent, la détermination de ν' passe par :

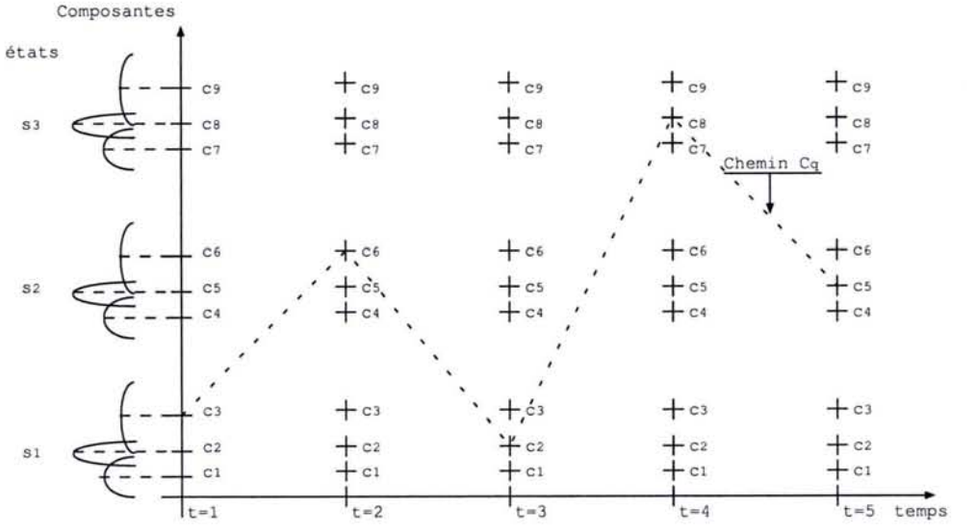
$$\nu' = \arg\max_{\nu} p(\mathbf{Y}, S, C|\nu, \Lambda_X) \tag{3.3}$$

$$= \arg\max_{\nu} \sum_{S_p \in S} \sum_{C_q \in C} p(\mathbf{Y}, S_p, C_q|\nu, \Lambda_X) \tag{3.4}$$

[15] voir la section 1.3.1 pour une description des notations



(a) Séquence d'états.



(b) Séquence de composantes gaussiennes.

FIG. 3.2 – Illustration des séquences S et C introduites par *Stochastic Matching*.

3.2.1 La fonction auxiliaire

La maximisation de la vraisemblance par rapport à ν :

$$\nu' = \arg\max_{\nu} p(\mathbf{Y}|\nu, \Lambda_X)$$

fait intervenir la fonction auxiliaire :

$$Q(\nu'|\nu) = E\left[log\ p(\mathbf{Y}, S, C|\nu', \Lambda_X)|\mathbf{Y}, \nu, \Lambda_X\right] \tag{3.5}$$

qui peut être ré-écrite ainsi :

$$Q(\nu'|\nu) = \sum_{S_p \in S} \sum_{C_q \in C} p(\mathbf{Y}, S_p, C_q|\nu, \Lambda_X) log\ p(\mathbf{Y}, S_p, C_q|\nu', \Lambda_X) \tag{3.6}$$

En effet il a été montré ([Dempster *et al.*, 1977]) que, si l'on obtient un ensemble de paramètres ν' tel que $Q(\nu'|\nu) \geq Q(\nu|\nu)$ alors on a la relation entre les vraisemblances :

$$log\left(p(\mathbf{Y}|\nu', \Lambda_X)\right) \geq log\left(p(\mathbf{Y}|\nu, \Lambda_X)\right)$$

Ceci s'explique par le fait que :

$$log\left(p(\mathbf{Y}|\nu', \Lambda_X)\right) = Q(\nu'|\nu) - H(\nu'|\nu)$$

avec

$$H(\nu'|\nu) = \sum_{S_p \in S} \sum_{C_q \in C} log\left(p(\mathbf{Y}|S_p, C_q, \nu', \Lambda_X)\right) p(\mathbf{Y}|S_p, C_q, \nu, \Lambda_X)$$

Or d'après l'inégalité de Jensen [Dempster *et al.*, 1977], on a toujours $H(\nu|\nu) \geq H(\nu'|\nu)$

3.2.2 Estimation-Maximisation

L'étape d'estimation consiste donc à établir la fonction $Q(\nu'|\nu)$ et l'étape de Maximisation consiste à trouver :

$$\nu'' = \arg\max_{\nu'} Q(\nu'|\nu)$$

D'après la construction de la fonction auxiliaire on garantit une augmentation de la vraisemblance en itérant les étapes d'estimation et de maximisation .

A priori, F_ν peut être une fonction qui transforme une séquence de I vecteurs d'observation en une séquence de J vecteurs ($I \geq J$). Pour la suite, on suppose plutôt que F_ν transforme chaque vecteur acoustique $\mathbf{y}_t$ en un vecteur compensé $\mathbf{x}_t$ exactement. Dans ce cas, la relation entre $\mathbf{x}_t$ et $\mathbf{y}_t$, vecteurs paramétriques de taille D :

$$\mathbf{x}_t = (x_{t,1}, \ldots, x_{t,D})^T \text{ et } \mathbf{y}_t = (y_{t,1}, \ldots, y_{t,D})^T$$

est

$$\mathbf{x}_t = f_\nu(\mathbf{y}_t)$$

Nous utiliserons cette représentation de la fonction de compensation dans la suite des dérivations.

Supposons que Λ_X soit composé de HMMs. La probabilité de $\mathbf{x}_t$ sachant que l'état d'émission s soit d'indice i est :

$$p(\mathbf{x}_t|s=i,\Lambda_X) = \sum_{j=1}^{M} w_{i,j} p(\mathbf{x}_t|s=i,c=j,\Lambda_X)$$

avec

$$p(\mathbf{x}_t|s=i,c=j,\Lambda_X) = \mathcal{N}(\mathbf{x}_t;\mu_{i,j};\mathbf{\Sigma}_{i,j})$$

où

- M : nombre de composantes gaussiennes dans chaque état de Λ_X,
- $w_{i,j}$: poids de la composante j dans l'état i,
- $\mathcal{N}$: distribution normale

$$\mathcal{N}(\mathbf{x};\mu_{i,j};\mathbf{\Sigma}_{i,j}) = \frac{1}{(2\pi)^{\frac{D}{2}}|\mathbf{\Sigma}_{i,j}|^{\frac{1}{2}}} exp\left(-\frac{1}{2}(\mathbf{x}_t-\mu_{i,j})^T\mathbf{\Sigma}_{i,j}^{-1}(\mathbf{x}_t-\mu_{i,j})\right)$$

- $\mu_{i,j}, \mathbf{\Sigma}_{i,j}$: moyenne et matrice de covariance de cette composante.

Dans ce cas, on peut exprimer la probabilité de l'observation bruitée $\mathbf{y}_t$ en fonction de l'état i et de la composante j par :

$$p(\mathbf{y}_t|s=i,c=j,\nu',\Lambda_X) = \frac{p(f_{\nu'}(\mathbf{y}_t)|s=i,c=j,\nu',\Lambda_X)}{|J_{\nu'}(\mathbf{y}_t)|}$$

avec $J_{\nu'}(\mathbf{y}_t)$, la matrice Jacobienne $(D \times D)$, dont la valeur aux coordonnées $(k,l) \in (D \times D)$ est :

$$J_{\nu',k,l} = \frac{\partial y_{t,k}}{\partial f_{\nu',l}(\mathbf{y}_t)} \text{ avec } (k,l) \in (D \times D)$$

où $f_{\nu',l}(\mathbf{y}_t)$ est la l-ième composante de $f_{\nu'}(\mathbf{y}_t)$:

$$f_{\nu'}(\mathbf{y}_t) = (f_{\nu',1}(\mathbf{y}_t),\ldots,f_{\nu',D}(\mathbf{y}_t))^T$$

Or la fonction auxiliaire (équation 3.6) peut s'exprimer de la façon suivante :

$$Q(\nu'|\nu) = \sum_{S_p\in S}\sum_{C_q\in C} p(\mathbf{Y},S_p,C_q|\nu,\Lambda_X) log\{ \prod_{t=1}^{T} a_{q_p(t-1),q_p(t)} w_{q_p(t),c_q(t)} p(\mathbf{y}_t|s=q_p(t),c=c_q(t),\Lambda_X)\} \quad (3.7)$$

Ce qui donne :

$$Q(\nu'|\nu) = \sum_{S_p\in S}\sum_{C_q\in C} p(\mathbf{Y},S_p,C_q|\nu,\Lambda_X)\sum_{t=1}^{T}\{ log(a_{q_p(t-1),q_p(t)}) + log(w_{q_p(t),c_q(t)}) + log(\mathcal{N}(p(f_{\nu'}(\mathbf{y}_t)|s=s_p(t),c=c_q(t),\Lambda_X)) - log(|J_{\nu'}(\mathbf{y}_t)|)\} \quad (3.8)$$

Il est possible de la décomposer en :

$$\begin{aligned} Q(\nu'|\nu) &= \sum_{n=1}^{N} p(\mathbf{Y}, q(1) = n|\nu, \Lambda_X) log(a_{\pi_n}) \\ &+ \sum_{t=2}^{T}\sum_{n=1}^{N}\sum_{l=1}^{N} p(\mathbf{Y}, q(t) = n, q(t-1) = l|\nu, \Lambda_X) log(a_{l,n}) \\ &+ \sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} p(\mathbf{Y}, q(t) = n, c(t) = m|\nu, \Lambda_X) log(w_{n,m}) \\ &+ \sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} p(\mathbf{Y}, q(t) = n, c(t) = m|\nu, \Lambda_X) log(p(f_{\nu'}(\mathbf{y}_t)|s = n, c = m, \Lambda_X)) \\ &- \sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} p(\mathbf{Y}, q(t) = n, c(t) = m|\nu, \Lambda_X) log(|J_{\nu'}(\mathbf{y}_t)|) \end{aligned} \tag{3.9}$$

avec la probabilité jointe de $\mathbf{Y}$ et de la composante m du n-ème état produisant $\mathbf{y}_t$ à la date t.

$$\gamma_t(n, m) = p(\mathbf{Y}, q(t) = n, c(t) = m|\nu, \Lambda_X)$$

Cette probabilité peut être calculée par l'algorithme avant/arrière par :

$$\gamma_t(n, m) = \alpha_t(n)\beta_t(n)\frac{w_{n,m}\mathcal{N}(f_\nu(\mathbf{y}_t); \mu_{n,m}; \mathbf{\Sigma}_{n,m})}{\sum_{j=1}^{M} w_{n,j}\mathcal{N}(f_\nu(\mathbf{y}_t); \mu_{n,j}; \mathbf{\Sigma}_{n,j})}$$

avec

- $\alpha_t(n)$ la probabilité ***avant*** (équation 1.3.2 en section 1.3.2)
- $\beta_t(n)$ la probabilité ***arrière*** (équation 1.2 en section 1.3.2)

Si on fait abstraction des termes ne dépendant pas de ν', on obtient donc :

$$\begin{aligned} &Q(\nu'|\nu) = \\ &\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} \gamma_t(n, m)\left\{-\tfrac{1}{2}(f_{\nu'}(\mathbf{y_t}) - \mu_{n,m})^T\mathbf{\Sigma}_{n,m}^{-1}(f_{\nu'}(\mathbf{y_t}) - \mu_{n,m}) - log\,|J_{\nu'}(\mathbf{y_t})|\right\} \end{aligned} \tag{3.10}$$

Enfin, pour maximiser $Q(\nu'|\nu)$, on égale sa dérivée partielle selon ν' à zéro, ce qui donne :

$$\frac{\partial}{\partial\nu'}\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} \gamma_t(n, m)\left\{-\frac{1}{2}(f_{\nu'}(\mathbf{y_t}) - \mu_{n,m})^T\mathbf{\Sigma}_{n,m}^{-1}(f_{\nu'}(\mathbf{y_t}) - \mu_{n,m}) - log\,|J_{\nu'}(\mathbf{y_t})|\right\} = 0$$

3.2.3 Une fonction de compensation simple

Supposons que la fonction f_ν agisse indépendamment sur chaque dimension et que les matrices de covariance utilisées sont diagonales. On peut écrire

$$x_{t,d} = f_{\nu,d}(y_{t,d}), \text{ pour toute dimension } d \in \{1, \ldots, D\}$$

Par la suite, on développera les calculs pour une seule dimension d. Par souci de clarté, nous abandonnerons l'indice d se rapportant à la dimension.

Considérons la fonction simple :

$$f_\nu(y_t) = a \,.\, y_t + b$$

(ici le paramètre de la fonction de compensation est $\nu = (a, b)$). Alors l'équation auxiliaire peut s'écrire sous la forme :

$$Q(a', b'|a, b) = \sum_{t,n,m}^{T,N,M} \gamma_t(n,m) \left[-\frac{(a' y_t + b' - \mu_{n,m})^2}{2\sigma_{n,m}^2} + log\ a' \right]$$

Afin de maximiser la fonction auxiliaire, on annule ses gradients par rapport à a' et b'. On obtient respectivement :

$$\sum_{t,n,m}^{T,N,M} \gamma_t(n,m) \left[\frac{1}{a'} - \frac{(a' y_t + b' - \mu_{n,m}) y_t}{\sigma_{n,m}^2} \right] = 0$$

et

$$\sum_{t,n,m}^{T,N,M} \gamma_t(n,m) \left[\frac{a' y_t + b' - \mu_{n,m}}{\sigma_{n,m}^2} \right] = 0$$

qui donnent explicitement les valeurs de a' et b'.

Si on considère la transformation particulière $f_\nu(\mathbf{y}_t) = \mathbf{y}_t + \mathbf{b}$, l'estimation itérative d'un biais fixe $\mathbf{b} = (b_1, \ldots, b_D)^T$ de dimension D (la dimension des vecteurs paramétriques) aura pour formule :

$$b_d = -\frac{\sum_{t,n,m}^{T,N,M} \gamma_t(n,m) \frac{y_{t,d} - \mu_{n,m,d}}{\sigma_{n,m,d}^2}}{\sum_{t,n,m}^{T,N,M} \frac{\gamma_t(n,m)}{\sigma_{n,m,d}^2}}$$

Nous aboutissons donc à l'expression d'un biais correctif dans le domaine cepstral. La détermination de ce biais dépend de la connaissance de statistiques issues du processus de reconnaissance ($\gamma_t(n,m)$). Ces statistiques sont déterminées par une première passe de reconnaissance sur la phrase de test. Comme précisé dans [Sankar and Lee, 1996], il est important de voir que l'hypothèse de reconnaissance obtenue lors de cette première passe guide la détermination du biais. Par conséquent, une mauvaise hypothèse peut mener à une très mauvaise estimation du biais.

3.3 Autres approches du *Stochastic Matching*

Bien que [Sankar and Lee, 1996] soit l'une des premières publications à utiliser le terme *Stochastic Matching*, des méthodes de robustesse utilisent des approches similaires sans faire mention de cette appellation.

Ces méthodes utilisent la classification des données de test en classes acoustiques obtenue par le SRAP pour développer la fonction de compensation. En cela, elles sont semblables à l'approche ML développée dans [Sankar and Lee, 1996] et aboutissent sous certaines conditions à un même algorithme de compensation.

3.3.1 Approche par l'optimisation de l'information de Kullback-Leibler

Cette approche est développée dans [Delphin-Poulat *et al.*, 1998], où un algorithme d'estimation récursive des paramètres d'une fonction de compensation est proposé. L'approche proposée est générale dans le sens où la forme de la fonction de compensation n'est pas imposée (elle peut être non-linéaire). Cependant, les dérivations pour une fonction de compensation plus élaborée qu'une fonction affine sont complexes.

Dans le domaine temporel, la fonction d'environnement peut s'écrire de la façon suivante :

$$y(t) = x(t) \otimes h(t) + n(t) \tag{3.11}$$

où

- $\otimes$ est l'opérateur de convolution,
- h est le filtre de canal,
- n le bruit additif.

Dans le domaine cepstral, la relation devient :

$$\mathbf{y} = \mathbf{x} - g(x, n, h) \approx \mathbf{x} - g(\mathbf{y})$$

où

- $\mathbf{y}$ et $\mathbf{x}$ sont les équivalent cepstraux de y et x
- $g(x, n, h)$ est une fonction non-linéaire dont l'expression régulière est très complexe et souvent approximée par $g(\mathbf{y})$.

Le but de l'algorithme de compensation est donc de trouver la transformation f telle que :

$$f(\mathbf{y}) \approx \mathbf{x}$$

Ici, on considère f comme étant une fonction de forme connue et définie par un ensemble de paramètres θ_0.

L'information de Kullback-Leibler entre deux densités de probabilité p_{θ_0} et p_{θ_1} d'une même variable aléatoire $\mathbf{Y}$ est définie par :

$$K(\theta_0, \theta_1) = \int_{\mathbf{Y}} log \frac{p_{\theta_0}(y)}{p_{\theta_1}(y)} p_{\theta_0}(y) dy$$

L'information s'annule quand les deux densités sont égales [Basseville and Nikiforov, 1993]. Dans le cas d'un processus aléatoire, l'information de Kullback-Leibler contenue dans une séquence de taille t ($\mathbf{Y}_t = \{y_1, \ldots, y_t\}$) est :

$$K_t(\theta_0, \theta_1) = \frac{1}{t} \int log \frac{p_{\theta_0}(\mathbf{Y}_t)}{p_{\theta_1}(\mathbf{Y}_t)} p_{\theta_0}(\mathbf{Y}_t) d\mathbf{Y}_t$$

ou encore :

$$K_t(\theta_0, \theta_1) = \frac{1}{t} \sum_{\tau=1}^{t} \int log \frac{p_{\theta_0}(\mathbf{Y}_{\tau-1})}{p_{\theta_1}(\mathbf{Y}_{\tau-1})} p_{\theta_0}(\mathbf{Y}_{\tau-1}) d\mathbf{Y}_{\tau-1}$$

Avec :

$$K(\theta_0, \theta_1) = \lim_{t \to +\infty} K_t(\theta_0, \theta_1)$$

Dans le cadre de l'algorithme, on cherchera donc à minimiser :

$$J(\theta) = K_t(\theta_0, \theta_1) = A\ E\left[log\left(p(\mathbf{Y}_t|\theta)\right)|\theta^0\right]$$

- θ_0 est le paramètre optimum.
- A est une constante indépendante de θ et de $\mathbf{Y}$.

On cherche à déterminer θ de sorte que l'information de Kullback-Leibler $J(\theta)$ soit la plus proche de 0.

Il est possible d'approcher cette valeur optimale de façon incrémentale par intermédiaire de la fonction auxiliaire :

$$\begin{aligned} \theta_{t+1} &= \arg\max_{\theta} Q_{t+1}(\Theta_t, \theta) \\ Q_{t+1}(\Theta_t, \theta) &= \sum_{\tau=1}^{t+1} \mathcal{L}_{\tau|t+1}(\Theta_{\tau-1}) \end{aligned}$$

avec $\Theta_t = (\theta_0, \ldots, \theta_t)$.

Considérons les modèles acoustiques comme étant des HMMs à N états, chaque état d'indice n étant caractérisé par un mélange de M Gaussiennes de moyenne $\mu_{n,k}$ et variance $\sigma_{n,k}$ pondérées par le facteur $w_{n,k}$ $(k = 1, \ldots, M)$.

La fonction auxiliaire est définie à partir de l'expression de la vraisemblance suivante.

$$\begin{aligned} \mathcal{L}_{\tau|t+1}(\Theta_{\tau-1}) &= \log(|f'_\theta(y_\tau)|) - \\ \tfrac{1}{2}\textstyle\sum_{n=1}^{N}\sum_{k=1}^{M} \gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)\frac{(f_\theta(y_\tau)-\mu_{n,k})^2}{\sigma^2_{n,k}} \end{aligned}$$

Dans laquelle

- $f'_\theta(y_\tau)$ est la dérivée partielle de la fonction de compensation par rapport à l'observation à l'instant τ, y_τ
- $\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)$ est la probabilité que le τ-iéme état de la séquence optimale d'états globale s_τ soit celui d'indice n et que sa principale composante gaussienne g_τ soit celle d'indice k, sachant la séquence partielle d'observations Y_{t+1} et la séquence des estimations précédentes des paramètres : $\Theta_{\tau-1}$.

$$\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k) = P(s_\tau|Y_{t+1}, \Theta_{\tau-1})$$

Alors,

$$\theta_{t+1} = \theta_t + \frac{S(\theta_t, y_{t+1})}{I_{t+1}(\theta_t)}$$

avec

$$S(\theta_t, y_{t+1}) = \frac{\partial \mathcal{L}_{t+1|t+1}(\Theta_t, \theta)}{\partial \theta}|_{\theta=\theta_t}$$

et

$$I_{t+1}(\theta_t) = -\frac{\partial^2 Q_{t+1}(\Theta_t, \theta)}{\partial^2 \theta}|_{\theta=\theta_t}$$

Considérons une fonction de compensation simple $f_B(y_{t+1}) = y_{t+1} + b_t$. Dans ce cas, $\theta_t = b_t$ et $\Theta = B$, de plus :

$$\mathcal{L}_{\tau|t+1}(B_{\tau-1}) = -\frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{M} \gamma_{\tau|t+1,B_{\tau-1}}(n,k)\frac{(y_\tau + b_t - \mu_{n,k})^2}{\sigma^2_{n,k}}$$

et

$$\frac{\partial \mathcal{L}_{t+1|t+1}(B_t, b)}{\partial b}|_{b=b_t} = -\sum_{n=1}^{N}\sum_{k=1}^{M} \gamma_{t|t+1,B_t}(n,k)\frac{(y_{t+1} + b_t - \mu_{n,k})}{\sigma^2_{n,k}}$$

et

$$\frac{\partial^2 \mathcal{L}_{\tau|t+1}(B_t, b)}{\partial b^2}|_{b=b_t} = -\sum_{n=1}^{N}\sum_{k=1}^{M}\gamma_{\tau|t+1,B_{\tau-1}}(n,k)\frac{1}{\sigma_{n,k}^2}$$

donc

$$S(b_t, y_{t+1}) = -\sum_{n=1}^{N}\sum_{k=1}^{M}\gamma_{t+1|t+1,B_t}(n,k)\frac{(y_{t+1}+b_t-\mu_{n,k})}{\sigma_{n,k}^2}$$

et

$$I_{t+1}(b_t) = \sum_{\tau=1}^{t+1}\sum_{n=1}^{N}\sum_{k=1}^{M}\gamma_{\tau|t+1,B_{\tau-1}}(n,k)\frac{1}{\sigma_{n,k}^2}$$

Alors, la séquence de paramètres de biais $B_t = \{b_0, \ldots, b_t\}$ peut être estimée grâce à la séquence optimale d'états donnée par l'algorithme de Viterbi :

$$b_{t+1} = b_t - \frac{\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{t+1|t+1,B_t}(n,k)\frac{y_{t+1}+b_t-\mu_{n,k}}{\sigma_{n,k}^2}}{\sum_{\tau=1}^{t+1}\sum_{n=1}^{N}\sum_{k=1}^{K}\frac{\gamma_{\tau|t+1,B_{\tau-1}}(n,k)}{\sigma_{n,k}^2}} \tag{3.12}$$

où

$$\gamma_{\tau|t+1,B_{\tau-1}}(n,k) = p(s_\tau = n, g_\tau = k|Y_{t+1}, B_{\tau-1})$$

Les probabilités $\gamma_{\tau|t+1,B_{\tau-1}}(n,k)$ sont indisponibles en cours de reconnaissance puisqu'il est nécessaire de connaître la séquence complète des états pour les calculer.

On remarque que l'expression obtenue pour le calcul du biais est très proche de celle obtenu lors de l'approche ML du *Stochastic Matching* décrit dans [Sankar and Lee, 1996]. Dans [Delphin-Poulat *et al.*, 1998], le lecteur intéressé trouvera aussi les expressions des paramètres d'une fonction de compensation affine

$$f_B(y_{t+1}) = a_t\, y_{t+1} + b_t$$

Les expériences menées par les auteurs ne permettent pas de dire pour quel type de conditions acoustiques cet algorithme est le plus adapté. En effet, elles ont été menées sur des bases de test collectées sur téléphone portable, en milieu extérieur (les modèles acoustiques étant entraînés sur de la parole propre, sur le réseau téléphonique fixe). Par conséquent, on ne peut savoir si cette méthode compense le bruit additif ou la transformation de canal.

3.3.2 Estimation Séquentielle d'un biais

Dans [Afify, 1999] est proposé un algorithme séquentiel de calcul de biais additif qui offre la possibilité de suivre l'évolution du bruit pour une phrase de test. Cette méthode est basée sur la minimisation de l'erreur prédictive (le lecteur intéressé se rapportera à [Ford and Moore, 1998] pour une description de l'erreur prédictive).

Dans ces travaux, la comparaison est faite avec l'algorithme présenté à la section précédente. En effet, les travaux de [Delphin-Poulat *et al.*, 1998] constitueraient une approximation de ceux présentés dans [Afify, 1999]. Il y est montré que si le processus de reconnaissance utilise l'hypothèse de Viterbi selon laquelle on peut simplifier la somme des probabilités de tous les chemins par le maximum des probabilités sur tous les chemins, alors les deux approches sont équivalentes.

L'algorithme de compensation par biais proposé suppose connue la séquence optimale des états. Cette séquence peut être fournie (***supervised compensation***) ou obtenue lors d'une première passe de reconnaissance (***unsupervised***).

L'approche proposée repose sur la minimisation de l'erreur de prédiction récursive (RPE, *Recursive Prediction Error*). Dans le cas où les états des HMMs $\Lambda = \{\lambda_1, \ldots, \lambda_N\}$ sont mono-Gaussiens, l'expression de la RPE est :

$$\begin{aligned} V_t(b) &= \frac{1}{2}\sum_{\tau=1}^{t}\sum_{i=1}^{N}\gamma_{\tau|\tau}(i)\frac{(y_\tau + b - \mu_i)^2}{\sigma_i^2} \\ &= \sum_{\tau=1}^{t} l_\tau(b) \end{aligned} \tag{3.13}$$

où

- t est l'instant courant,
- N est le nombre de HMMs
- μ_i et σ_i^2 sont les paramètres de la densité de probabilité gaussienne de l'état d'indice i (moyenne et variance)
- $\gamma_{t|t}(i)$ est l'estimée de probabilité d'état : $\gamma_{t|t}(i) = p(s_t = i|y_1, \ldots, y_t, \Lambda, b_1, \ldots, b_{t-1})$

L'algorithme proposé réduit récursivement la RPE et permet de récupérer le biais b optimal :

$$b_t = b_{t-1} - \frac{\frac{\partial l_t(b)}{\partial b|_{b=b_{t-1}}}}{\frac{\partial^2 V_t(b)}{\partial b^2|_{b=\{b_1,\ldots,b_{t-1}\}}}}$$

où le dénominateur

$$\partial^2 V_t(b)/\partial b^2|_{b=\{b_1,\ldots,b_{t-1}\}}$$

peut être approximé par

$$\frac{1}{f_t} = \frac{\eta}{f_{t-1}} + \sum_{i=1}^{N}\frac{\gamma_{\tau|\tau}(i)}{\sigma_i^2}$$

où η est un facteur d'oubli compris entre 0 et 1 [Ford and Moore, 1998].

Avec l'approximation de Viterbi, on simplifie la somme sur tous les états par le maximum obtenu à l'instant τ.

$$\begin{aligned} \hat{\gamma}_{\tau|\tau}(i) &= 1, \text{ si } i = s_\tau = \arg\max_j \gamma_{\tau|\tau}(j) \\ &= 0 \text{ sinon.} \end{aligned} \tag{3.14}$$

Sous cette condition, on obtient la séquence de biais :

$$b_t = b_{t-1} - f_t.\frac{y_t + b_{t-1} - \mu_{s_t}}{\sigma_{s_t}^2}$$

avec

$$\frac{1}{f_t} = \frac{\eta}{f_{t-1}} + \frac{1}{\sigma_{s_t}^2}$$

où

- η est un facteur d'oubli à fixer expérimentalement, dont la valeur est comprise entre 0 et 1.
- s_τ est l'indice de l'état dominant à l'instant τ

$$s_\tau = \arg\max_j \gamma_{\tau|\tau}(j)$$

Dans [Afify, 1999] il est fait remarquer que cette récursion peut être mise en relation avec celle donnée dans la section précédente. Mais, contrairement à cette dernière, la récursion proposée dans [Afify, 1999] offre une charge calculatoire très réduite.

Plusieurs techniques non supervisées ont été proposées :
- effectuer une première passe par le reconnnaisseur pour déterminer un chemin possible (que l'on fournira dans un deuxième temps au processus de calcul de biais),
- calculer un biais pour chaque modèle possible et modifier la séquence de test par autant de biais,
- calculer un biais pour chaque modèle, en faire une moyenne et transformer la séquence de test grâce à ce biais.

Mais aucune de ces méthodes ne permet de donner un résultat de reconnaissance supérieur à celui obtenu par la compensation supervisée. L'algorithme est testé avec un système de reconnaissance de nombres prononcés en arabe (vocabulaire de taille 12). Les modèles acoustiques sont indépendants du locuteur. On cherche donc à tester la possibilité pour l'algorithme de compensation d'augmenter la robustesse du système par rapport à la variabilité inter-locuteurs. Pour ce SRAP, le taux de reconnaissance en mots isolés est de 93% sans compensation, de 96,8% en mode supervisé et ne dépasse pas 92,4% pour les modes non-supervisés proposés.

Il est intéressant de voir que le mode non-supervisé, dans ce cadre expérimental, n'améliore pas le taux de reconnaissance. L'auteur ne précise pas si ce comportement se reproduit lorsque la variation entre données de test et d'entraînement n'est plus due à une variabilité inter-locuteur mais à une variabilité dans l'environnement acoustique (bruit de fond...). On peut cependant conclure que le mode supervisé permet de combattre efficacement la variabilité inter-locuteur.

3.3.3 Estimation récursive d'un bruit non-stationnaire

Dans [Deng *et al.*, 2001] et [Deng *et al.*, 2003], un algorithme itératif permettant d'estimer exclusivement la composante additive du bruit est proposé. Cette estimée est utilisée pour compenser la séquence de signal par l'intermédaire d'une fonction non-linéaire qui est en fait la décomposition en série de Taylor ou premier ordre de la fonction de corruption.

Le signal corrompu $\mathbf{y}$ peut s'exprimer à partir du signal original $\mathbf{x}$ et de l'influence du bruit additif $\mathbf{n}$:

$$\mathbf{y} \approx \mathbf{x} + g(\mathbf{n} - \mathbf{x})$$

avec

$$g(\mathbf{z}) = \mathbf{Cos}\ log\ [\mathbf{I} + exp(\mathbf{Cos}^T \mathbf{z})]$$

où **Cos** est la matrice de transformation discrète en cosinus, utilisée pour formuler la transformation de Fourier discrète (DFT, *Discret Fourier Transform*).

$$\mathbf{Cos} \approx DFT$$

Le signal de parole $\mathbf{x}$ est quant à lui modélisé par un mélange de Gaussiennes multi-dimensionnelles :

$$p(\mathbf{x}) = \sum_{m=1}^{M} c_m \mathcal{N}(\mathbf{x}; \mu_m^x, \mathbf{\Sigma}_m^x)$$

La décomposition en Série de Taylor au point de fonctionnement μ_0^x et $\mathbf{n}_0$ donne :

$$\mathbf{y} \approx \mathbf{x} + g(\mathbf{n}_0 - \mu_0^x) + \mathbf{G}(\mathbf{n}_0 - \mu_0^x)(\mathbf{x} - \mu_0^x) + [\mathbf{I} - \mathbf{G}(\mathbf{n}_0 - \mu_0^x)](\mathbf{n} - \mathbf{n}_0) \tag{3.15}$$

avec

$$\mathbf{G}(\mathbf{z}) = \mathbf{I} - \mathbf{Cos}\ diag(\frac{\mathbf{I}}{\mathbf{I} + exp(\mathbf{Cos}^T \mathbf{z})})\mathbf{Cos}^T$$

Dans [Deng *et al.*, 2001] et [Deng *et al.*, 2003], on effectue une estimation de la composante de bruit $\mathbf{n}$, de façon récursive, par Estimation-Maximisation. Cette approximation sera utilisée comme point de fonctionnement dans le développement de Taylor de $\mathbf{y}$ et permettra donc d'obtenir une bonne approximation du signal original $\mathbf{x}$. L'estimation de $\mathbf{n}$ passe par l'optimisation d'une fonction objectif :

$$\mathbf{n}_{t+1} = \arg\max_{\mathbf{n}} \mathbf{Q}_{t+1}(\mathbf{n})$$

avec

$$\mathbf{Q}_{t+1}(\mathbf{n}) = E(log\ p(\mathbf{y}_1, \ldots, \mathbf{y}_{t+1}|\mathbf{n})|\mathbf{y}_1, \ldots, \mathbf{y}_{t+1}, \mathbf{n}_1, \ldots, \mathbf{n}_t)$$

qui se simplifie en :

$$\mathbf{Q}_{t+1}(\mathbf{n}) = \sum_{\tau=1}^{t+1} \sum_{m=1}^{M} \gamma_\tau(m)\ log\ p(\mathbf{y}_\tau|m, \mathbf{n}) + const.$$

où $\gamma_\tau(m) = p(m|\mathbf{y}_\tau, \mathbf{n}_{\tau-1})$. peut être calculé à partir de la décompostion en série de Taylor de la fonction d'environnement (équation 3.15).

Les détails des itérations en 4 étapes nécessaires à l'estimation de la valeur de $\mathbf{n}$ sont dans [Deng *et al.*, 2003]. Cette méthode permet d'obtenir une estimation de la composante additive du bruit pour chaque trame.

Dans [Deng *et al.*, 2001], on fait le choix d'utiliser cette estimée pour élaborer une base stéréo artificielle nécessaire à l'algorithme de compensation SPLICE (*Stereo-based Piecewise LInear Compensation for Environments*). Cependant, une telle estimée pourrait servir de base à de nombreux processus de compensation.

Dans [Deng *et al.*, 2001], on présente les détails de l'implémentation de l'algorithme dans un SRAP et les résultats obtenus sur Aurora2. On observe que l'estimation de la composante de bruit par cette approche EM est plus précise que son estimation MMSE par une approche numérique. Les deux approches permettent une amélioration du taux de reconnaissance et peuvent être combinées avec CMN.

3.4 Conclusion

Nous venons de voir une approche particulière de la robustesse : le *Stochastic matching*. Les algorithmes utilisant ce paradigme permettent de compenser l'influence de la fonction d'environnement en se basant sur la comparaison de la séquence de test avec une modélisation des signaux utilisés pour l'entraînement.

Ainsi, comme les méthodes exposées au chapitre précédents, ces algorithmes nécessitent la mise en correspondance entre données de test et d'entraînement pour développer une stratégie de compensation. Ici, cette mise en correspondance est stochastique et est fournie par l'étape de reconnaissance. Les algorithmes de compensation basés sur le *Stochastic Matching* se caractérisent donc par leur cohérence avec le processus de reconnaissance.

Le chapitre 5 propose une nouvelle approche originale du *Stochastic Matching* qui permet à la fois d'exploiter cette cohérence et d'effectuer une compensation en temps-réel.

Le chapitre suivant décrit le système de reconnaissance et les bases de test que nous avons utilisé pour tester les propriétés de cet algorithme et comparer ses performances avec celles obtenues par des méthodes de robustesse classiques.

4

Présentation du système de reconnaissance et du cadre applicatif

Sommaire

Dans ce chapitre, nous présentons en détails la configuration du système de reconnaissance et des bases de test utilisés pour l'ensemble de nos expérimentations.

4.1 Système de reconnaissance ESPERE

Nous avons utilisé le système de reconnaissance ESPERE (*Engine for SPEech REcognition*) [Fohr *et al.*, 2000].

Il s'agit d'un SRAP à vocabulaire moyen développé au LORIA. Ce système est composé principalement de trois modules : un module de traitement acoustique, un module d'apprentissage et un moteur de reconnaissance (cf figure 4.1).

Le module de traitement acoustique s'occupe de la paramétrisation du signal avant le traitement par le module de reconnaissance. Il est surtout utilisé pour le codage sous forme de MFFCs[16] et est entièrement paramétrable. Il est possible par exemple de modifier la taille des fenêtres d'analyse, leur recouvrement, le nombre de filtres, le nombre de paramètres...

Le module d'apprentissage utilise l'algorithme de Baum-Welch pour entraîner des modèles acoustiques HMMs continus. C'est-à-dire que les probabilités d'émission sont commandées

[16] voir section 4.3

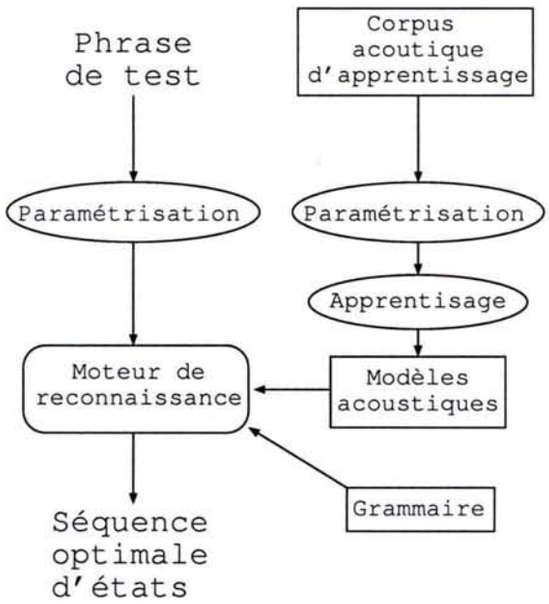


FIG. 4.1 – Schéma général du système ESPERE.

par des fonctions de densité de probabilité (PDF, *Probability Density Function*). Ces PDFs sont des mélanges de Gaussiennes. Le nombre de Gaussiennes participant à une PDF est modulable et est incrémenté de manière itérative (par dédoublement) durant la phase d'entraînement.

Le moteur de reconnaissance met en application un algorithme synchrone à un seul passage. Il nécessite l'ensemble des HMMs appris comme décrit précédemment et une grammaire. La grammaire présente les couples de mots qui peuvent se suivre (bigrammes). Il est possible d'associer à chaque couple une probabilité.

L'implémentation d'ESPERE contient plus de 2000 lignes de code C++ et il fonctionne aussi bien sous les systèmes d'exploitation Linux et Windows.

4.2 Bases de test

Afin d'évaluer les performances des différentes approches de la robustesse étudiés dans nos travaux de recherche, nous avons utilisé comme bases de test des enregistrements effectués dans des voitures en mouvement. En effet, l'habitacle d'une voiture est un milieu riche en interférences sonores (bruit du moteur, bruits extérieurs au véhicule, réverbération de la voix dans l'espace confiné, ...).

4.2.1 VODIS

Cette base de données a été enregistrée par 200 locuteurs francophones (98 femmes et 102 hommes) alors qu'ils conduisaient. Les locuteurs ont répété les phrases en français prononcées par

le co-pilote. Les enregistrements ont été faits à 11025Hz. Pour chaque séance d'enregistrement, deux microphones ont été utilisés pour obtenir ces données stéréo sur deux canaux :

canal *close-talk* : un microphone *SENNHEISER HME 1410 K* ou *SHURE MS10 A*, proche de la bouche (micro-casque), fournit un enregistrement que l'on considère exempt de bruit additif (RSB=21dB en moyenne).

canal *far-talk* : un microphone *AKG Q400 II* placé sur le pare soleil fournit un enregistrement de la voix corrompue par les bruits de l'habitacle (RSB=11dB en moyenne).

a. Trois types de voitures ont été utilisées à cet effet :
   - VW Passat GL TDI (diesel, 17 locuteurs)
   - Peugeot 406 (essence, 68 locuteurs)
   - Renault Mégane (essence, 115 locuteurs)

b. Plusieurs conditions de conduite ont été considérées :
   - conduite sur autoroute et route, à plusieurs allures
   - conduite avec fenêtre ouverte (37 locuteurs) ou fermée (163 locuteurs)
   - conduite avec la radio allumée (48 locuteurs) ou éteinte (152 locuteurs)
   - conduite avec la climatisation (55 locuteurs) éteinte ou allumée (145 locuteurs)
   - conduite par temps de pluie ou par temps sec

c. L'ensemble des phrases peut se diviser en plusieurs exercices :
   - ***lettres*** : chaque locuteur a prononcé les 38 lettres
   - ***chiffres*** : chaque locuteur a prononcé 5 séquences de 4 chiffres
   - ***commande*** : chaque locuteur a prononcé :
     - mettre + ACRONYM
     - appeler + NAME
     - appeler le + NUMERO
   - ***nombre*** : chaque locuteur a prononcé 29 nombres inférieurs à 100.
   - ***nombres100à15000*** : chaque locuteur a prononcé 20 nombres compris entre 15000 et 100 choisis pour leur équilibre phonétique.
   - ***épelés*** : chaque locuteur a épelé 5 noms.
   - ***spontané*** : chaque locuteur a prononcé 5 phrases spontanées.
   - ***phonétique*** : chaque locuteur a prononcé 5 phrases choisies pour leur équilibre phonétique.
   - ***téléphone*** : chaque locuteur a prononcé 5 numéros de téléphone (10 chiffres).
   - ***mot-clé*** : chaque locuteur a prononcé 70 mots clés parmi une liste de 104 mots.

Dans nos expériences, on utilisera en particulier la reconnaissance en mots pour les tâches *nombres100à15000* et *téléphone* ou la reconnaissance en phonème sur l'ensemble de la base de test.

4.2.2 Aurora3

La base de données Aurora3 a été développée pour ETSI (European Telecommunications Standard Institut) à des fins d'évaluation des systèmes de paramétrisation. Cette base existe en Finnois, Italien, Espagnol, Allemand et Danois. La base finlandaise étant la plus étendue, nous l'avons choisie pour nos expériences. 200 locuteurs ont participé à l'élaboration de cette base. C'est une base à petit vocabulaire puisque les phrases sont des séquences de chiffres. Les enregistrements ont été faits dans l'habitacle d'une automobile dans des conditions variées.

La fréquence d'échantillonnage est de 8kHz. Tous les enregistrements sont en stéréo : deux microphones ont été utilisés, l'un proche de la bouche (*close-talk*) et l'autre éloigné (main libre, *far-talk*). Encore une fois, les données *close-talk* ne sont pas exemptes de bruit mais seront considérées comme les données *propres*.

Plusieurs conditions de conduite ont été recensées :

- à l'arrêt, moteur en marche (condition : *Quiet*, 259 phrases prononcées par 59 hommes et 291 phrases prononcées par 61 femmes)
- conduite lente, en ville ou sur petite route
  - fenêtre ouverte (condition : *Low noise*, 207 phrases prononcées par 87 hommes et 206 phrases prononcées par 83 femmes)
  - fenêtre fermée (condition : *Low noise*, 206 phrases prononcées par 87 hommes et 206 phrases prononcées par 83 femmes)
- conduite rapide, sur une bonne route
  - musique de fond (condition : *High noise*, 207 phrases prononcées par 84 hommes et 206 phrases prononcées par 82 femmes)
  - sans musique de fond (condition : *High noise*, 206 phrases prononcées par 84 hommes et 206 phrases prononcées par 82 femmes)

4.3 Paramétrisation utilisée pour les expérimentations

Dans l'ensemble des expériences que nous avons réalisées, nous avons utilisé une paramétrisation de types MFCC (*Mel Frequency Cepstral Coeficients*). Ce codage est le plus répandu dans les SRAP actuels.

L'analyse menant aux paramètres MFCC est dite *homomorphique* c'est-à-dire que la représentation dans le temps des paramètres spectraux et celle des paramètres cepstraux ont la même forme. Ces cœfficients sont considérés comme robustes à une grande gamme de variabilité. D'une part, ils assurent une séparation entre les deux composantes du signal vocal : la fonction d'excitation (caractérisée par le *pitch*) et la fonction de transfert du conduit vocal. La contribution du conduit vocal se retrouve dans les basses *quéfrences* (premiers coefficients cepstraux) alors que la contribution de l'excitation se localise dans les *quéfrences* élevées. Un *liftrage* passe-bas permettra donc de séparer ces deux composantes : c'est pourquoi on ne retient généralement que les premiers coefficients cepstraux pour le codage. D'autre part, ils sont très peu sensibles à la puissance acoustique du signal analysé. En effet, cette puissance est concentré dans le premier coefficient cepstral. L'adjonction des dérivées premières et secondes par rapport au temps des coefficients cepstraux rend ces derniers encore plus résistants aux fluctuations dues au locuteur ou à l'environnement (comme évoqué dans [Haton, 2002]).

La figure 4.2 illustre les étapes du codage du signal (analogique) de la parole en coefficients MFCC.

Les paramètres utilisés pour le codage sont aussi présents sur cette figure. La fréquence d'échantillonnage pour les deux bases de test étant différente, les tailles de fenêtre varient aussi. Le but est toujours d'obtenir une analyse sur des portions du signal où il peut être considéré comme stationnaire (10ms). Les fenêtres de Hamming sont utilisées pour permettre une analyse sur une portion du signal sans effet de bord. Les fenêtres sont glissantes : ce sont des fenêtres de largeur de 30ms, se recouvrant sur 20ms.

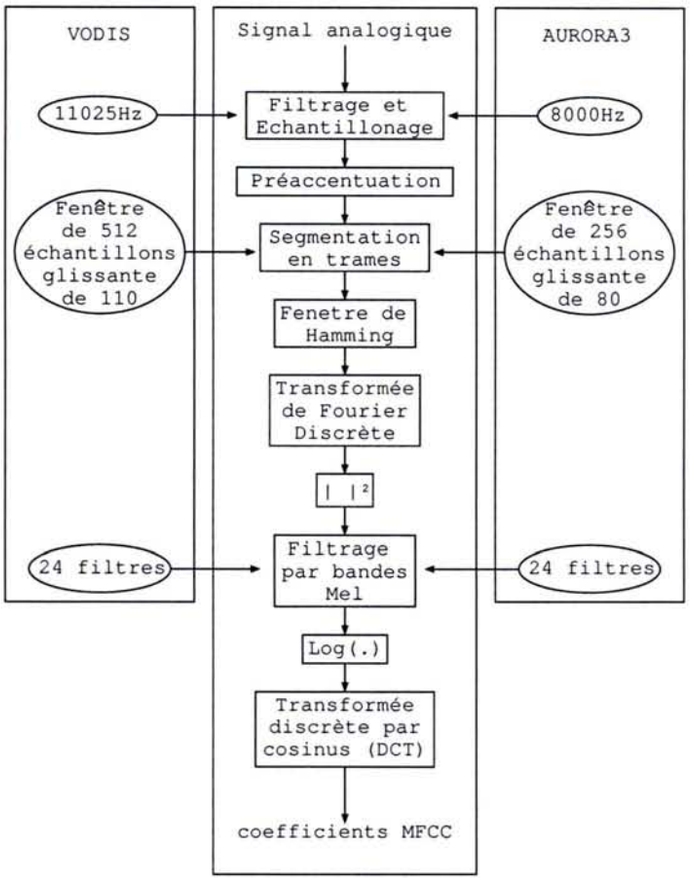


FIG. 4.2 – Le codage MFCC.

On passe au domaine spectral en appliquant la transformée de Fourier, l'implantation étant du type FFT (*Fast Fourier Transform*). Cette FFT et son inverse font intervenir une matrice de valeurs de cosinus, c'est pourquoi on parle aussi parfois de transformation en Cosinus discret. Le spectre de sortie est alors filtré par un banc de filtres répartis sur l'échelle Mel :

$$Mel(f) = \frac{1000}{Log(2)}(1 + \frac{f}{1000})$$

Seuls les 12 premiers coefficients cepstraux sont gardés, car toutes les informations nécessaires sont dans les basses *quéfrences*. On calcule par la suite les dérivées premières (Δ) et secondes ($\Delta\Delta$) sur une fenêtre de 5 trames de cepstre :

$$\begin{aligned} x_t^{k+12} &= \Delta x_t^k = x_{t+2}^k + x_{t+1}^k - x_{t-1}^k - x_{t-2}^k,\ 1 \leq k \leq 12 \\ x_t^{k+24} &= \Delta\Delta x_t^k = \Delta x_{t+1}^k - \Delta x_{t-1}^k,\ 1 \leq k \leq 12 \end{aligned} \tag{4.1}$$

4.4 Modélisation acoustique et résultats de référence

4.4.1 VODIS

Les corpus de test et d'entraînement

Chaque locuteur ayant fait un nombre équivalent d'enregistrements, dans des conditions homogènes, nous avons décidé de séparer la base VODIS en une base de test et une base d'entraînement selon les locuteurs. Ainsi, nous nous assurons que les locuteurs ayant participé à l'entraînement des modèles acoustiques sont différents de ceux ayant participé à la base de test. Les modèles acoustiques obtenus sont donc totalement indépendants du locuteur. On distinguera donc :

- le corpus de test, constitué de toutes les phrases des locuteurs 001 002 003 004 005 006 007 009 010 012 014 015 016 017 025 029 031 032 035 036 037 038 039 040 041 043 045 046 047 128 136 149 150 177 178 179 162 164 187 200. On distinguera le corpus de test *close-talk* et le corpus de test *far-talk*.
- le corpus d'entraînement, constitué de toutes les phrases *close-talk* des locuteurs restants.

Les modèles acoustiques

Nous avons choisi de modéliser les phonèmes par des HMMs de type *Bakis* à 3 états. La liste des phonèmes retenus pour la modélisation est visible dans le tableau 4.1.

Chaque état comporte une densité de probabilité d'émission sous forme de GMM de 8 gaussiennes à matrice de covariance diagonale.

Les grammaires

De toutes les tâches proposées par la base VODIS, nous en avons utilisé 3 plus particulièrement :

- la reconnaissance en mots de *nombres100à15000*,

nom	exemple
i	*dix* [**vcl d i s**]
y	*rue* [**R y**]
u	*douze* [**vcl d u z 6**]
e	*trouver* [**cl t R u v e**]
E	*sept* [**s E cl t**]
E/	*zero* [**z E/ R o**]
6	*quatre* [**cl k a cl t R 6**]
2	*deux* [**vcl d 2**]
9	*neuf* [**n 9 f**]
o	*au* [**o**]
O	*quatorze* [**cl k a cl t O R z 6**]
O/	*vélodrome* [**v E/ l O/ vcl d R O/ m**]
a	*quatre* [**cl k a cl t R 6**]
e~	*vingt* [**v e~ **]
9~	*un* [**9~ **]
a~	*trente* [**cl t R a~ cl t 6**]
o~	*onze* [**o~ z 6**]
cl p	*proche* [**cl p R O S**]
vcl b	*boulevard* [**vcl b u l v a R**]
cl t	*trois* [**cl t R w a**]
vcl d	*deux* [**vcl d 2**]
cl k	*quinze* [**cl k e~ z 6**]
vcl g	*gare* [**vcl g a R**]
f	*neuf* [**n 9 f**]
v	*vingt* [**v e~**]
s	*six* [**s i s**]
z	*douze* [**vcl d u z 6**]
S	*proche* [**p R O S**]
Z	*bonjour* [**vcl b o~ Z u R**]
m	*mille* [**m i l**]
n	*neuf* [**n 9 f**]
J	*union* [**y J o~ **]
l	*mille* [**m i l**]
R	*zero* [**z E/ R o**]
j	*station* [**s cl t a s j o~ **]
H	*huit* [**H i cl t**]
w	*trois* [**cl t R w a**]
silence	
!	bruit de bouche

TAB. 4.1 – Description des modèles acoustiques utilisés pour les tests sur la base VODIS.

- la tâche *téléphone*,
- la reconnaissance des phonèmes sur toute la base de test.

Pour effectuer la reconnaissance en mots de *téléphone*, nous avons utilisé une grammaire permettant la reconnaissance de 5 séries d'unités numériques. Une unité numérique est un nombre (de 10 à 99) ou un couple de chiffres commençant par "zéro". Les chiffres et les nombres sont transcrits comme une succession de phonèmes, plusieurs transcriptions étant autorisées.

Pour effectuer la reconnaissance en mots de *nombres100à15000*, nous avons utilisé une grammaire permettant la reconnaissance des nombres présentés de diverses manières usuelles. Par exemple, le nombre 1254 peut être présenté sous la forme "mil deux cent cinquante quatre" ou "douze cent cinquante quatre". Le SRAP ne peut donc fournir comme réponse qu'une séquence bien formée du point de vue de la grammaire française (on ne peut obtenir à la sortie du SRAP : "quarante-douze"). Là encore, chiffres et nombres sont représentés comme une suite de phonèmes.

Pour la reconnaissance en phonème, aucune grammaire n'a été utilisée : tout phonème peut succéder à un autre, sans restriction de longueur.

Les taux de reconnaissance

Les tableaux 4.2 et 4.3 donnent les résultats de la reconnaissance[17] sur VODIS pour différentes tâches et exercices. Pour chaque valeur, les intervalles de confiance[18] (*int. conf*) sont donnés. Ces valeurs nous serviront de référence par la suite.

(%)	Insertion	Délétion	Substitution	*Acc*	*int. conf.*
close-talk	9.2	14.0	21.6	55.2	± 0.3
far-talk	7.8	28.8	26.3	37.2	± 0.3

TAB. 4.2 – Résultats de la reconnaissance en phonèmes pour ESPERE sans mécanisme de compensation sur la totalité des phrases de test *close-talk* et *far-talk* de VODIS.

(%)	*close-talk*	*far-talk*
nombres100à15000	89.9 (± 1.2)	63.5 (± 2.0)
téléphone	95.2 (± 0.9)	78.6 (± 1.6)

TAB. 4.3 – Taux de reconnaissance en mot (%) pour ESPERE sans mécanisme de compensation sur les phrases de test *far-talk* des tâches *nombres100à15000* et *téléphone*.

4.4.2 Aurora3

ETSI a divisé la base Aurora3 en ensembles d'entraînement et de test afin de définir 3 tâches spécifiques de reconnaissance.

[17] une description du taux de reconnaissance est fournie en annexe A
[18] une description de l'intervalle de confiance est fournie en annexe A

Les corpus de test et d'entraînement

- Higly Mismatch (HM) : les modèles acoustiques sont entraînés sur de la parole *close-talk* (toutes conditions). Le test s'effectue sur *far-talk* dans des conditions *Low noise* et *High noise.*
- Medium Mismatch (MM) : les modèles acoustiques sont entraînés dans un milieu bruité différent du milieu de test : l'entraînement se fait sur les données *far-talk* dans les conditions *Low noise.* Le test se fait sur les données *far-talk* dans les conditions *High noise.*
- Well Matched (WM) : les modèles acoustiques sont entraînés dans le même environnement que celui du test, c'est-à-dire en *far-talk* et dans toutes les conditions.

Les modèles acoustiques

Ici, nous avons utilisé des modèles de mots car le vocabulaire de la base est très réduit (chiffres). La liste des mots reconnus est contenue dans le tableau 4.4.

Finnois	Français
nolla	zéro
yksi	un
kaksi	deux
kolme	trois
nelja	quatre
viisi	cinq
kuusi	six
seitseman	sept
kahdeksan	huit
yhdeksan	neuf

TAB. 4.4 – Modèles acoustiques de mots utilisés pour la reconnaissance sur la base en finnois d'Aurora3.

Ces modèles sont constitués de 16 états, caractérisés par 3 gaussiennes de matrice de covariance diagonale.

La grammaire utilisée

La grammaire utilisée autorise la succession de mots sans limite de nombres ni ordre particulier.

Les taux de reconnaissance

Les taux de reconnaissance obtenus par ESPERE, grâce à ces modèles, sont recensés dans le tableau 4.5.

(%)	Insertion	Délétion	Substitution	*Acc*	*int. conf.*
HM	13.1	26.6	14.2	46.2	± 1.9
MM	6.8	12.4	5.9	75.0	± 2.3
WM	4.1	4.4	2.7	88.8	± 0.7

TAB. 4.5 – Taux de reconnaissance (en mot) , d'insertion, de délétion et de substitution sur les trois tâches Aurora3 (Finlandais) pour ESPERE sans mécanisme de compensation.

4.5 Conclusion

Le système de reconnaissance ESPERE constitue une base de recherche très intéressante car il est facile d'y intégrer de nouveaux modules, de nouvelles fonctions. L'implémentation du processus de compensation que nous exposerons au chapitre 5 a été grandement facilité par sa simplicité et sa lisibilité.

Les qualités de simplicité et de lisibilité se retrouvent dans la conception et la forme de la base VODIS. De plus cette base est très complète et bien élaborée. Son utilisation dans nos recherches s'est donc imposée.

Aurora3, comme VODIS est aussi une base de test enregistrée dans un milieu automobile. Le formalisme de la base Aurora3 est plus complexe mais son usage est très répandu. Nous l'avons utilisé afin de confirmer les observations que avons formulées sur VODIS.

5

Approche personnelle d'un algorithme de compensation temps réel basé sur le *Stochastic Matching*

Sommaire

L'objectif de notre recherche est d'obtenir un processus de compensation permettant de rendre facilement un SRAP robuste à une grande variété d'environnements. Pour cela, ce processus doit être capable de se passer d'hypothèses *a priori* sur la nature du bruit qui corrompt le signal de parole et, *a fortiori*, de toute supervision. Cet objectif est atteint par les techniques de *Stochastic Matching*. De plus il doit être capable d'être réactif à cet environnement, c'est-à-dire qu'il doit

pouvoir adapter rapidement sa technique de compensation dans l'éventualité de changements dans les conditions de test.

Dans les travaux originaux sur le *Stochastic Matching*, les paramètres d'une fonction de compensation sont estimés afin de maximiser la vraisemblance d'une séquence de parole en fonction de l'ensemble des modèles acoustiques. Ces paramètres sont obtenus à l'issue de plusieurs étapes d'Estimation-Maximisation pour lesquels il est nécessaire de connaître la séquence optimale des modèles acoustiques modélisant la séquence de vecteurs de parole observée. L'aspect le plus intéressant du *Stochastic Matching* est qu'il ne nécessite, à première vue, aucune information sur la nature ou le niveau de bruit ambiant. En effet, il est possible d'effectuer le débruitage d'une phrase de test sans disposer d'autres informations *a priori*.

Nous avons vu dans le chapitre 2 que les méthodes de compensation peuvent se diviser en deux sous-ensembles : les algorithmes peuvent être *temps-réel* (*on-line*) ou en *temps-différé* (*off-line*). Les algorithmes *temps-réel*, ou synchrones à la trame, sont particulièrement intéressants lorsqu'il s'agit de contrer l'influence d'une source de bruit lentement variable [Deng *et al.*, 2001]. Des algorithmes *temps-différé* permettraient d'obtenir un résultat similaire pour de telles sources de bruit mais les calculs qu'ils impliquent rendent leur utilisation incompatible avec une implantation dans un SRAP dans un cadre applicatif (SRAP embarqué dans une voiture par exemple). La technique développée durant nos recherches et exposée dans ce chapitre est totalement synchrone à la trame : les paramètres de la fonction de compensation sont réactualisés à l'arrivée de chaque observation acoustique, en parallèle avec le processus de reconnaissance.

Un algorithme de compensation utilisant le cadre formel du *Stochastic Matching* tout en étant synchrone à la trame soulève un problème ardu. En effet, dans ce cas, les statistiques (probabilités *avant* et *arrière*) nécessaires dans la méthode *Stochastic Matching* exposée au chapitre 3, ne sont pas disponibles au moment de compenser une trame de parole (toute la séquence de parole n'a pas encore été observée). Or ces statistiques sont fournies par la séquence optimale des modèles acoustiques sur la séquence complète des observations. Afin de nous départir de cet écueil, nous avons envisagé d'approximer ces statistiques manquantes par les seules disponibles lors de la compensation : les probabilités *avant*. A l'aide de cette approximation, il est alors possible d'estimer les paramètres d'une fonction de compensation affine permettant de compenser la parole à chaque instant. Ces paramètres sont réactualisés à chaque trame. De plus, ils sont obtenus de façon incrémentale, leur estimation est donc d'autant plus fine que le nombre d'observations émises de façon consécutive dans un même environnement acoustique est important. Par conséquent, cet algorithme *temps-réel* effectue une compensation en parallèle avec le processus de reconnaissance et ne nécessite pas de connaissances *a priori* sur la nature du bruit.

Ce chapitre s'articule comme suit. Dans un premier temps, nous partirons du cadre théorique du *Stochastic Matching* et exposerons nos hypothèses avant d'aboutir à une présentation générale de notre algorithme de compensation (sections 5.1 et 5.1.1). Puis, nous exposerons les propriétés et les performances de cet algorithme par quelques expériences simples (section 5.2). Nous évaluerons ensuite notre approche en la comparant avec des méthodes classiquement utilisées en robustesse (section 5.4). Enfin, nous confirmerons nos conclusions sur les performances de notre algorithme en effectuant une série d'expériences sur une nouvelle base de test (section 5.5).

5.1 Compensation synchrone à la trame

Dans l'ensemble de nos recherches, nous avons utilisé un système de reconnaissance basé sur le paradigme des modèles de Markov cachés (HMM). Ces modèles comportent N états et utilisent

des matrices de covariance diagonales pour leur densités de probabilité d'émission. Chaque état n est caractérisé par un mélange de K fonctions de probabilité gaussiennes de moyenne $\mu_{(n,k)}$, de variance $\sigma_{(n,k)}$ et de poids $w_{(n,k)}$.

Le cadre de travail du *Stochastic Matching* suppose l'utilisation d'une distance entre l'observation bruitée et une séquence d'états. Pour une estimation *temps-différé* de la fonction f, cette distance est calculée en utilisant la séquence optimale d'états [Surendran *et al.*, 1996]. Mais pour une méthode de compensation synchrone à la trame, seules des séquences partielles et hypothétiques sont disponibles. Dans ce cas, deux solutions peuvent être envisagées :

- estimer les statistiques nécessaires sur de courtes fenêtres d'observations, comme dans [Delphin-Poulat *et al.*, 1998],
- approximer ces statistiques par celles fournies lors du décodage de Viterbi, les probabilités *avant* dans notre cas.

L'idée principale de notre méthode est la suivante. L'hypothèse posée est qu'à chaque instant lors de l'alignement de Viterbi, l'état associé à la plus grande probabilité *avant* donne une bonne modélisation du vecteur de cepstre observé y. Les paramètres de la fonction f sont alors estimés afin de maximiser la vraisemblance de $f(y)$ en fonction de ces états. La fonction de compensation est calculée pour chaque trame et la confiance dans la justesse de ces paramètres augmente au cours de la reconnaissance de la séquence.

L'hypothèse classique de considérer les matrices de covariances des modèles acoustiques diagonales permet de considérer les dimensions cepstrales comme des dimensions indépendantes du point de vue de la compensation. Plus concrètement, à chaque instant, chaque dimension d'un vecteur d'observation sera transformé par une fonction dont les paramètres auront été calculés indépendamment des autres dimensions. Par conséquent, dans la suite des dérivations, nous ne considérerons que le cas unidimensionnel. Ainsi, pour disposer de l'ensemble du processus de compensation, il suffit d'appliquer la méthode décrite ci-après à chaque dimension du vecteur d'observation.

5.1.1 Compensation par simple biais

Supposons que le vecteur de signal propre dans le domaine temporel $x(t)$ soit distordu par l'environnement pour donner le signal bruité de $y(t)$. La distorsion peut être modélisée ainsi :

$$y(t) = x(t) \otimes h(t) + n(t) \tag{5.1}$$

où

- $\otimes$ est l'opérateur de convolution,
- $h(t)$ représente la réponse impulsionnelle du filtre de canal et
- $n(t)$ le bruit additif.

Dans le domaine cepstral, cette équation devient :

$$\mathbf{y} = \mathbf{x} + g(x(t), n(t), h(t))$$

où $g(x(t), n(t), h(t))$ est une fonction non-linéaire sans expression analytique simple.

De manière générale, les expressions exactes de $x(t)$, $n(t)$ et $h(t)$ sont inconnues. En pratique, on utilise une approximation de cette fonction ne dépendant que des observations $\mathbf{y}$.

$$\mathbf{y} \approx \mathbf{x} + g(\mathbf{y})$$

Le but de la compensation est donc de trouver une fonction $f(\mathbf{y})$ qui approche $\mathbf{x}$: $f(\mathbf{y}) \approx \mathbf{x}$.

Soit θ l'ensemble des paramètres de la transformation $f_\theta(y)$ de l'espace des observations de test vers celui d'entraînement. Il a été démontré dans [Delphin-Poulat *et al.*, 1998] que le paramètre θ optimisant l'information de Kullback-Leibler $J(\theta) = E\{log(p(Y_t|\theta))\}$ pouvait être approché par une séquence $\{\theta_i\}$ tendant à maximiser la fonction auxiliaire Q suivante :

$$\begin{aligned} \theta_{t+1} &= \arg\max_{\theta} Q_{t+1}(\Theta_t, \theta) \\ Q_{t+1}(\Theta_t, \theta) &= \sum_{\tau=1}^{t+1} L_{\tau|t+1}(\Theta_{\tau-1}) \end{aligned}$$

avec $\Theta_t = (\theta_0, \ldots, \theta_t)$. La fonction auxiliaire est définie à partir de l'expression de la vraisemblance suivante.

$$L_{\tau|t+1}(\Theta_{\tau-1}) = \log(|f'_\theta(y_\tau)|) - \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)\frac{(f_\theta(y_\tau) - \mu_{(n,k)})^2}{\sigma^2_{(n,k)}}$$

Dans laquelle

- les modèles acoustiques utilisés sont des *HMM*s à N états caractérisés par des densités de probabilité d'émission de K gaussiennes de moyenne $\mu_{(n,k)}$ et variance $\sigma_{(n,k)}$ et de poids $w_{(n,k)}$,
- $f'_\theta(y_\tau)$ est la dérivée partielle de la fonction de compensation par rapport à l'observation à l'instant τ, y_τ,
- $\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)$ est la probabilité que le τ-iéme état de la séquence optimale d'états globale s_τ soit celui d'indice n et que sa principale composante gaussienne g_τ soit celle d'indice k, sachant la séquence partielle d'observations et
- $Y_{t+1} = \{y_1, \ldots, y_{t+1}\}$ la séquence des estimations précédentes $\Theta_{\tau-1}$.

La recherche du maximum de cette vraisemblance par rapport aux paramètres de la transformation mène aux expressions suivantes :

Considérons une fonction de compensation simple :

$$f_B(y_{t+1}) = y_{t+1} + b_t$$

Alors, la séquence de paramètres de biais $B_t = \{b_0, \ldots, b_t\}$ peut être estimée grâce à la séquence optimale d'états donnée par l'algorithme de Viterbi :

$$b_{t+1} = b_t - \frac{\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{t+1|t+1,B_t}(n,k)\frac{y_{t+1}+b_t-\mu_{(n,k)}}{\sigma^2_{(n,k)}}}{\sum_{\tau=1}^{t+1}\sum_{n=1}^{N}\sum_{k=1}^{K}\frac{\gamma_{\tau|t+1,B_{\tau-1}}(n,k)}{\sigma^2_{(n,k)}}} \tag{5.2}$$

où

$$\gamma_{\tau|t+1,B_{\tau-1}}(n,k) = p(s_\tau = n, g_\tau = k|Y_{t+1}, B_{\tau-1})$$

Les probabilités $\gamma_{\tau|t+1,B_{\tau-1}}(n,k)$ sont indisponibles lors de l'alignement de Viterbi puisqu'il est nécessaire de connaître la séquence complète des états pour la calculer.

Dans notre algorithme, nous avons fait l'hypothèse que la probabilité *avant*

$$\alpha_{\tau|B_{\tau-1}}(n,k) = p(f_B(Y_\tau), s_\tau = n, g_\tau = k|B_{\tau-1})$$

pouvait être utilisée en place de γ dans l'équation 5.2

En utilisant cette hypothèse, la nouvelle expression du biais est :

$$b_{t+1} = b_t - \frac{\sum_{n=1}^{N}\sum_{k=1}^{K} \alpha_{t+1|B_t}(n,k)\frac{y_{t+1}+b_t-\mu_{(n,k)}}{\sigma^2_{(n,k)}}}{\sum_{\tau=1}^{t+1}\sum_{n=1}^{N}\sum_{k=1}^{K}\frac{\alpha_{\tau|B_{\tau-1}}(n,k)}{\sigma^2_{(n,k)}}} \tag{5.3}$$

Cette expression se simplifie si l'on suppose que la somme sur l'ensemble des états possibles et de leur composante gaussienne au temps τ peut être approximée par la seule contribution du couple (n,k) qui maximise $\alpha_{\tau|B_{\tau-1}}(n,k)$. Dans ce cas, l'équation devient :

$$b_{t+1} = b_t - \frac{\frac{y_{t+1}+b_t-\mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}}{\sum_{\tau=1}^{t+1}\frac{1}{\sigma^2_{(n,k)_\tau}}} \tag{5.4}$$

5.1.2 Algorithme

Dans la méthode proposée, le calcul du biais à l'instant t ne nécessite pas la séquence globale et optimale des états mais plutôt la séquence des états optimaux : à chaque trame d'instant t, l'état le plus probable au sens de la probabilité *avant* est utilisé pour ré-estimer les paramètres de la transformation de compensation. Ainsi, l'algorithme d'estimation de ces paramètres suit les étapes suivantes :

1. initialiser : $b_0 := 0$, $t := 1$
2. à l'instant t, calculer, pour tout (n,k)

   $$\alpha_{t|B_{t-1}}(n,k) := p(y_t + b_{t-1}, s_t = n, g_t = k)$$

   de l'alignement de Viterbi pour obtenir le couple $(n,k)_t$ tel que

   $$(n,k)_t = \arg\max_{(n,k)} \alpha_{t|B_{t-1}}(n,k)$$

3. à l'instant t, calculer b_t grâce aux paramètres de $(n,k)_t$, selon l'équation (5.4)

   $$b_t = b_{t-1} - \frac{\frac{y_t+b_{t-1}-\mu_{(n,k)_t}}{\sigma^2_{(n,k)_t}}}{\sum_{\tau=1}^{t}\frac{1}{\sigma^2_{(n,k)_\tau}}}$$

4. $t := t+1$

5.1.3 Compensation par fonction affine

De nombreux algorithmes utilisés en robustesse utilisent des fonctions de compensation de forme affine. L'approche VPS[19], par exemple, utilise une telle fonction. De même, l'approche

[19] cf section 2.2.4 du chapitre 2 pour une description de VPS

proposée dans [Delphin-Poulat *et al.*, 1998][20] permet de dériver les paramètres d'une fonction de compensation affine dans le cadre du *Stochastic Matching*.

Les dérivations menées dans la section précédente peuvent aboutir à l'estimation en temps réel de paramètres d'une fonction de compensation affine. Nous avons vu en section 5.1.1 que, dans le domaine cepstral, la distorsion entre le cepstre émis et le cepstre effectivement traité par le système de reconnaissance pouvait se traduire sous la forme :

$$y = x - g(x(t), n(t), h(t))$$

où $g(x(t), n(t), h(t))$ est une fonction non-linéaire sans expression analytique simple dépendant du bruit de canal $h(t)$, du bruit ambiant $n(t)$ et du signal de parole $x(t)$. Le but de la compensation est de trouver une fonction $f(y)$ qui approche x : $f(y) \approx x$. Dans la section précédente, nous avons dérivé les calculs et exposé les approximations qui ont mené à l'établissement d'une fonction f simple, sous forme d'un biais additif. Ces dérivations peuvent aussi s'effectuer en supposant que f est une fonction affine $f_t(y_{t+1}) = a_t\ y_{t+1} + b_t$. Cette fonction de compensation sera appelée *Affine* dans la suite.

Le résultat final des dérivations[21] donne, comme expression de a_t et b_t :

$$\begin{aligned}
a_{t+1} &= a_t - \frac{(m_{t+1} + E_{t+1} + \frac{t+1}{a_t^2} - (y_{t+1} - \frac{1}{a_t})D_{t+1})}{\delta_{t+1}} \\
b_{t+1} &= b_t - \frac{1}{\delta_{t+1}} \left(m_{t+1}(y_{t+1}C_{t+1} - D_{t+1}) - \frac{1}{a_t}C_{t+1} \right) \\
\delta_{t+1} &= \frac{1}{C_{t+1}\left(E_{t+1} + \frac{t+1}{a_t^2}\right) - (D_{t+1})^2} \\
E_{t+1} &= \sum_{\tau=1}^{t+1} \frac{y_\tau^2}{\sigma^2_{(n,k)_\tau}}, \quad D_{t+1} = \sum_{\tau=1}^{t+1} \frac{y_\tau}{\sigma^2_{(n,k)_\tau}} \\
C_{t+1} &= \sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}}, \quad m_{t+1} = \frac{a_t y_{t+1} + b_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}
\end{aligned} \tag{5.5}$$

5.2 Propriétés de notre approche

Cette section est consacrée à la présentation des performances expérimentales de notre algorithme de compensation. L'indicateur principal des performances d'un tel algorithme est bien entendu le taux de reconnaissance (en mots ou phonèmes) atteint par le SRAP l'utilisant. C'est pourquoi nous avons testé notre algorithme sur des bases variées, dans des environnements naturels et artificiels.

Afin d'étudier avec plus de précision les propriétés de notre algorithme, nous avons créé une série d'expériences constructives qui nous permettra de :

- vérifier par l'étude des log-vraisemblances calculées lors de la reconnaissance que les phrases compensées ont effectivement été projetées dans l'espace des modèles acoustiques, comme décrit dans le paradigme du *Stochastic Matching*.

[20] cf section 3.3.1 du chapitre 3

[21] les détails de ces dérivations sont visibles en annexe B

- mettre en évidence une relation entre la durée des phrases de test et l'efficacité de notre méthode de compensation.
- introduire un mécanisme simple pour faciliter la compensation en milieu acoustique variant lentement.
- affirmer que, pour notre algorithme, une fonction de compensation affine est plus efficace qu'un simple biais.
- vérifier que notre processus ne dégrade pas les performances du reconnaisseur en milieu non-bruité.

5.2.1 Augmentation de la vraisemblance finale

Le *Stochastic Matching* a pour but de rapprocher la séquence d'observations acoustiques de la séquence optimum des états. Afin de mettre en évidence la conservation de cette propriété par notre méthode, nous avons fait l'expérience suivante : nous avons comparé les log-vraisemblances des chemins optimums obtenus par ESPERE seul (*Référence*) et ESPERE intégrant notre méthode de compensation (*Biais*). Nous avons restreint la base de test aux seules phrases reconnues (reconnaissance en mots) à la fois par *Référence* et *Biais*. Pour ces phrases, les séquences d'états fournies par les systèmes sont les mêmes. On peut donc dire, dans ce cas précis, que les séquences optimums d'états obtenus par les deux systèmes sont égales. Dans la figure 5.1 sont représentés les couples de log-vraisemblances obtenues par *Biais* et par *Référence* pour les phrases de test bien reconnues par les deux systèmes de reconnaissance. Les valeurs représentées dans cette figure sont les logarithmes des probabilités *avant* cumulées et non normalisées. Sur l'axe des abscisses sont reportées les valeurs finales obtenues à la fin de la reconnaissance d'une phrase par un système n'incluant pas de système de compensation (*Référence*). Sur l'axe des ordonnés sont reportées les mêmes valeurs, obtenues sur les mêmes phrases, mais cette fois avec un système utilisant notre algorithme (*Biais*). Par conséquent, la différence entre les valeurs des log-vraisemblances

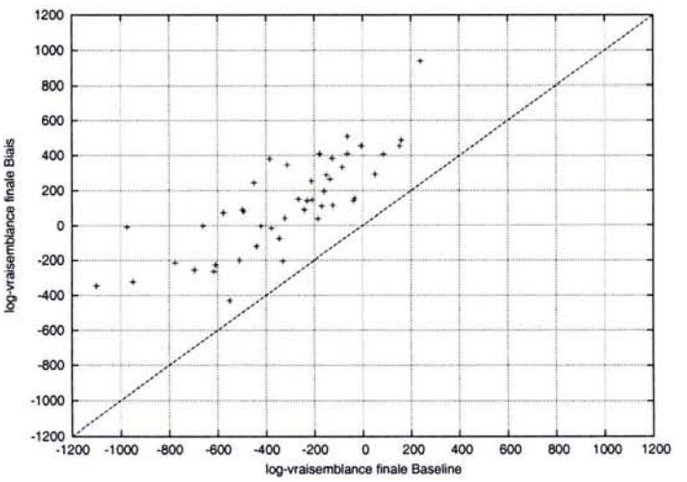


FIG. 5.1 – Log-vraisemblance finale de phrases bien reconnues par *Biais*.

finales n'est due qu'aux différences des valeurs des probabilités d'émission. En effet, comme vu à la section 1.3.2 du chapitre 1, l'expression de la vraisemblance finale s'exprime en fonction des

probabilités d'émission b_{q_j} et de transition $a_{q_i q_j}$ comme suit (voir notations au chapitre 1) :

$$P^* = \max_{i \in \{1,\ldots,N\}} [\delta_T(i)]$$

avec

$$\delta_1(i) = \pi_i b_i(\mathbf{o}_1) \text{ et } \Phi_1(i) = 0$$

et par récurrence :

$$\delta_t(j) = \max_{i \in \{1,\ldots,N\}} [\delta_{t-1}(i) a_{ij}] b_j(\mathbf{o}_t)$$

Or, on considère dans le cas de notre expérience que les chemins optimums obtenus par *Référence* et *Biais* sont égaux. Les contributions des probabilités de transitions sont alors les mêmes et les vraisemblances finales ne diffèrent que par la contribution des probabilités d'émission. Dans le cas de *Référence* ces probabilités d'émission sont calculées à partir des observations $\mathbf{o}_t$. Dans le cas de *Biais*, elles sont calculées à partir des observations compensées .

La figure 5.1 montre donc que, pour la totalité des phrases de test retenues, les log-vraisemblances finales obtenues par *Biais* sont supérieures à celles obtenues par *Référence*. Cette expérience nous amène à conclure que, sur une phrase de test, les probabilités d'émission sont supérieures dans le cas *Biais*. Par conséquent, la séquence des observations acoustiques transformées a bien été rapprochée de la séquence des états (optimums au sens de la probabilité *avant*).

La figure 5.2 permet d'illustrer cette conclusion. Pour cette expérience, nous avons utilisé deux phrases du corpus de test VODIS *far-talk* bien reconnues à la fois par *Référence* et *Biais*. Lors de la reconnaissance par *Référence* et *Biais*, le maximum (sur tout les états) de la probabilité *avant*, a été relevé à chaque trame de temps [22]. Chaque valeur des deux séquences ainsi formées représente donc la cumulation des probabilités d'émission et de transition pour une séquence partielle d'états. La figure 5.2 permet donc de suivre l'évolution de ces valeurs au cours de la reconnaissance, pour *Référence* et *Biais*. On peut noter que la séquence générée par *Biais* est toujours supérieure à celle générée par *Référence* et ce dès les premières trames. Cet écart, comme nous l'avons vu, s'explique par des valeurs plus élevées des probabilités d'émission. On peut donc en conclure que le processus de compensation *Biais* rapproche effectivement les observations transformées des modèles acoustiques.

5.2.2 Taux de reconnaissance en fonction de la durée des phrases de test.

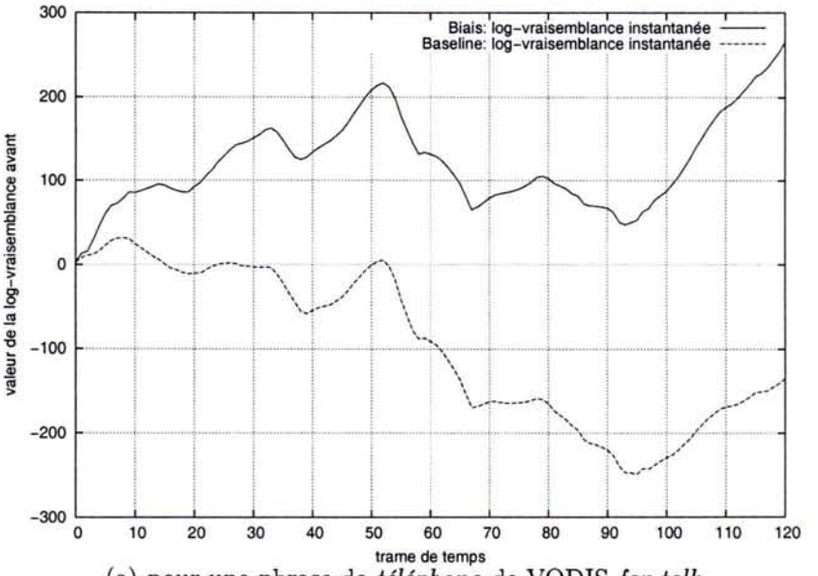
La figure 5.3 représente des taux de reconnaissance en phonèmes sur toute la base de test *far-talk* de VODIS. Les phrases de test de cette base ont été classées par durée. Ainsi, 5 sous-groupes de tailles comparables ont pu être formés :

- les phrases de durée inférieure à 1.5 secondes
- les phrases de durée comprise entre 1.5 et 2 secondes
- les phrases de durée comprise entre 2 et 2.5 secondes
- les phrases de durée comprise entre 2.5 et 4 secondes
- les phrases de durée supérieure à 4 secondes

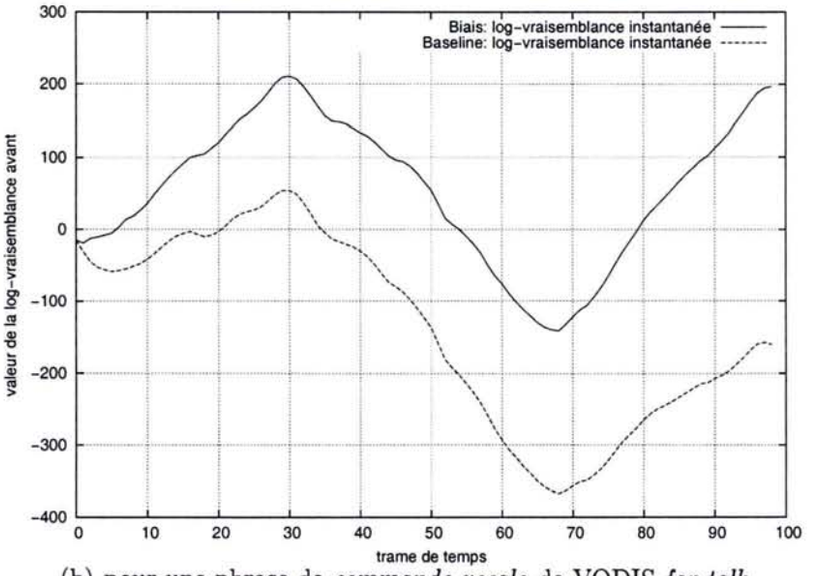
Pour chaque sous-groupe, les taux de reconnaissance en phonèmes ont été rapportés. Les intervalles de confiance[23] sont représentes en pointillés. Dans tout les cas de figure, la reconnaissance

[22] les valeurs reportées sur le graphique sont les log() des valeurs des probabilités *avant* non pondérées

[23] cf calcul en annexe A



(a) pour une phrase de *téléphone* de VODIS *far-talk*



(b) pour une phrase de *commande vocale* de VODIS *far-talk*

FIG. 5.2 – Evolution du maximum instantané de la Log-vraisemblance pour des phrases bien reconnues par *Biais*.

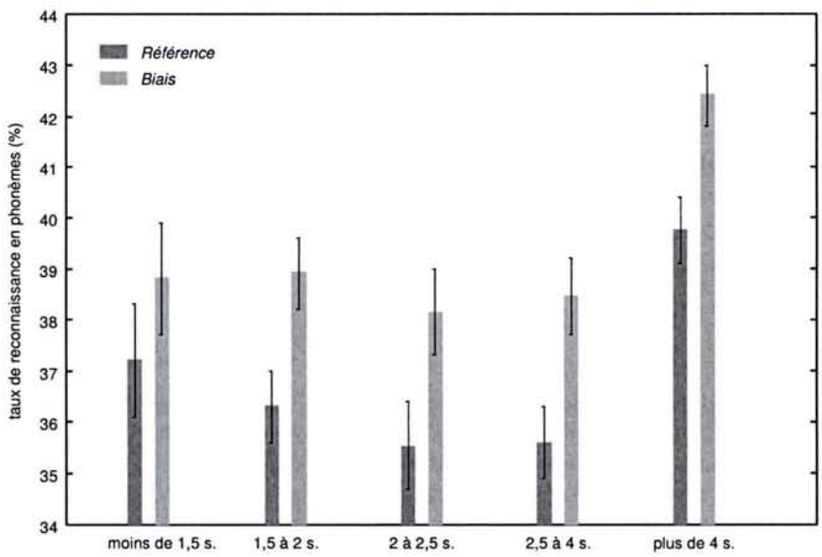


FIG. 5.3 – Comparaison des taux de reconnaissance en phonèmes sur toute la base VODIS *far-talk* entre *Référence* et *Biais*.

est améliorée par notre méthode.

En ce qui concerne la forme générale de la figure, les taux de reconnaissance décroissent puis croissent à mesure que la durée des phrases augmente. Ceci s'explique par le fait que le système de reconnaissance, et en particulier le paramètre appelé *Language Weight* qui permet de pondéré l'influence du modèle de langage durant la phase de reconnaissance ont été établis afin que le taux de reconnaissance soit maximal pour toute les durées de phrases confondues. Or, le *Language Weight* optimum n'est pas le même selon la durée de la phrase à reconnaître. C'est pourquoi les taux de reconnaissance de *Référence* pour des tailles de phrase différentes ne sont pas tous égaux.

5.2.3 Un facteur d'oubli pour des environnements variant lentement

Pour l'expérience que décrit la figure 5.4, nous avons introduit un facteur d'oubli *ff* sur le deuxième terme de l'équation 5.4. Ce qui donne :

$$b_{t+1} = b_t - ff * \frac{\frac{y_{t+1}+b_t-\mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}}{\sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}}} \tag{5.6}$$

La valeur de *ff*, inférieure à 1, modère l'influence du terme se rapportant à l'instant t par rapport au terme se rapportant au passé (le biais calculé à $t-1$). Par conséquent, plus le facteur d'oubli est faible, moins la composante relative à la distance entre l'observation et l'état le plus probable influe sur le calcul du biais. Nous obtenons ainsi des variations moins brusques du biais. Ceci se traduit donc par de meilleurs résultats de la méthode de compensation sur les phrases courtes, lorsque le facteur d'oubli est suffisamment bas.

La figure 5.4 illustre les remarques précédentes. Les taux de reconnaissance en phonèmes sur

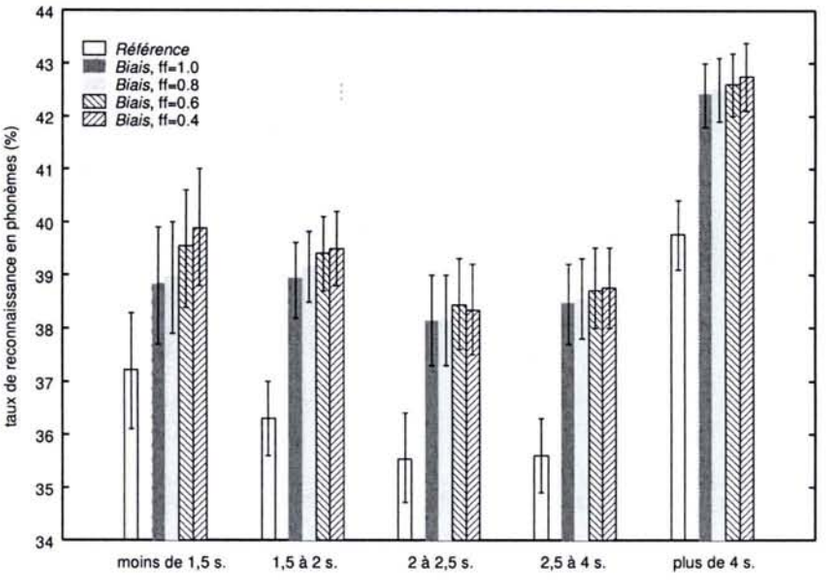


FIG. 5.4 – Comparaison des taux de reconnaissance en phonèmes sur toute la base VODIS *far-talk* entre *Référence* et *Biais* en fonction du facteur d'oubli.

les 5 groupes de test y sont reportés. Pour chaque groupe, la colonne de gauche représente le score obtenu par ESPERE sans système de compensation et les méthodes de droite, les scores obtenus par ESPERE avec notre méthode de compensation et un facteur d'oubli décroissant (de gauche à droite : 1.0, 0.8, 0.6 et 0.4). L'influence bénéfique du facteur d'oubli est particulièrement remarquable lorsque l'on effectue le test sur des phrases de durée inférieure à 1.5 secondes. Dans ce cas, lorsqu'on utilise un facteur d'oubli de 0.4, le taux de reconnaissance est significativement supérieur à celui obtenu sans méthode de compensation.

Nous verrons à la section 5.4.1 que la pondération introduite par ce facteur d'oubli permet à notre algorithme d'obtenir de très bons résultats dans des environnements variant rapidement.

5.2.4 Amélioration apportée par une fonction de compensation affine

La figure 5.5 représente les taux de reconnaissance en phonèmes pour la partie *far-talk* de toute la base de test VODIS. On y a représenté à la fois :

- les résultats obtenus par le système de reconnaissance sans méthode de compensation (*Référence*),
- les meilleurs résultats obtenus par notre méthode utilisant un biais comme fonction de compensation (*Biais*) en faisant varier le facteur d'oubli (ff)
- les meilleurs résultats obtenus par notre méthode utilisant une fonction affine comme fonction de compensation (*Affine*) en faisant varier le facteur d'oubli

Sur cette figure, on peut constater que l'utilisation d'une fonction affine comme fonction de compensation est toujours bénéfique. L'amélioration apportée est surtout notable dans la reconnaissance des phrases de courte durée (de moins de 2 secondes). Le facteur d'oubli peut aussi

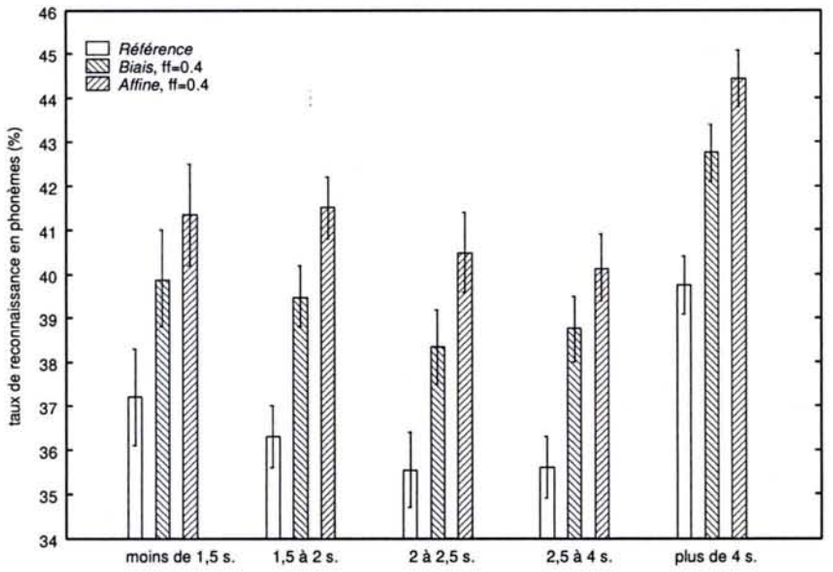


FIG. 5.5 – Comparaison des taux de reconnaissance en phonèmes sur toute la base de test VODIS *far-talk* entre *Référence*, *Biais* et *Affine*.

être utilisé lorsque la fonction de compensation est affine. Dans ce cas, l'équation 5.6 devient :

$$\begin{aligned} a_{t+1} &= a_t - ff * \frac{(m_{t+1}+E_{t+1}+\frac{t+1}{a_t^2}-(y_{t+1}-\frac{1}{a_t})D_{t+1})}{\delta_{t+1}} \\ b_{t+1} &= b_t - ff * \frac{1}{\delta_{t+1}}\left(m_{t+1}(y_{t+1}C_{t+1} - D_{t+1}) - \frac{1}{a_t}C_{t+1}\right) \end{aligned}$$

La figure 5.6 donne l'évolution du taux de reconnaissance en phonèmes sur toute la base de test VODIS *far-talk*.

On peut voir sur cette figure que le facteur d'oubli permet d'augmenter les scores pour un environnement évoluant lentement, comme dans le cas de la fonction *Biais*. Ici aussi, l'influence bénéfique du facteur d'oubli est particulièrement remarquable lorsque l'on effectue le test sur des phrases de durée courte.

5.2.5 Amélioration de la reconnaissance sur parole propre

La figure 5.7 présente les taux de reconnaissance en phonèmes pour toute la base de test VODIS *close-talk* (c'est-à-dire sans bruit). Là encore, on compare les résultats obtenus par la méthode proposée (implémentant une fonction de compensation biais ou affine) et ceux obtenus par le système de reconnaissance sans compensation. Cette figure nous permet d'observer que notre méthode de compensation ne dégrade pas les performances dans le cas où le bruit est faible (voir nul, comme c'est le cas ici). Au contraire, on obtient une amélioration par rapport aux résultats de base, surtout pour les phrases de courte durée. Il existe toujours une amélioration du taux de reconnaissance lorsqu'on utilise une fonction affine comme fonction de compensation, mais la différence n'est plus significative. On remarque par ailleurs que les taux de reconnaissance

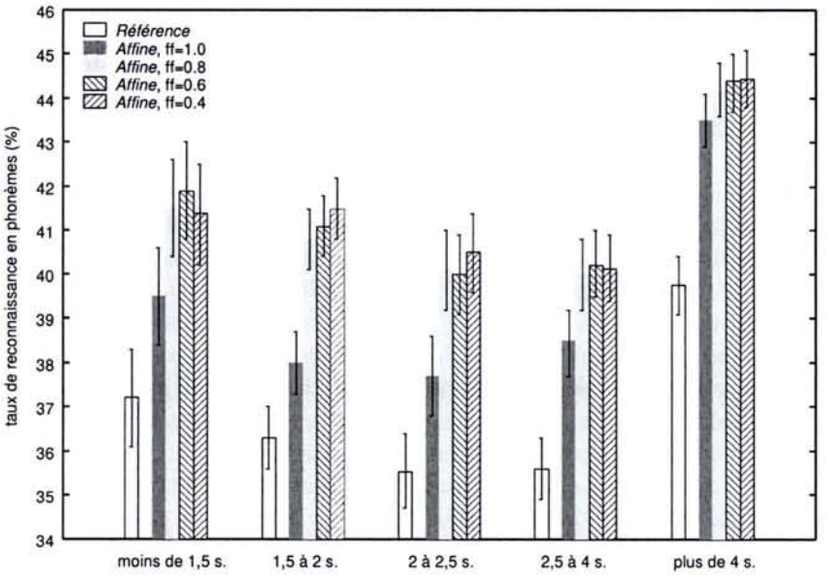


FIG. 5.6 – Comparaison des taux de reconnaissance en phonèmes sur toute la base de test VODIS *far-talk* entre *Référence* et *Affine* en fonction du facteur d'oubli.

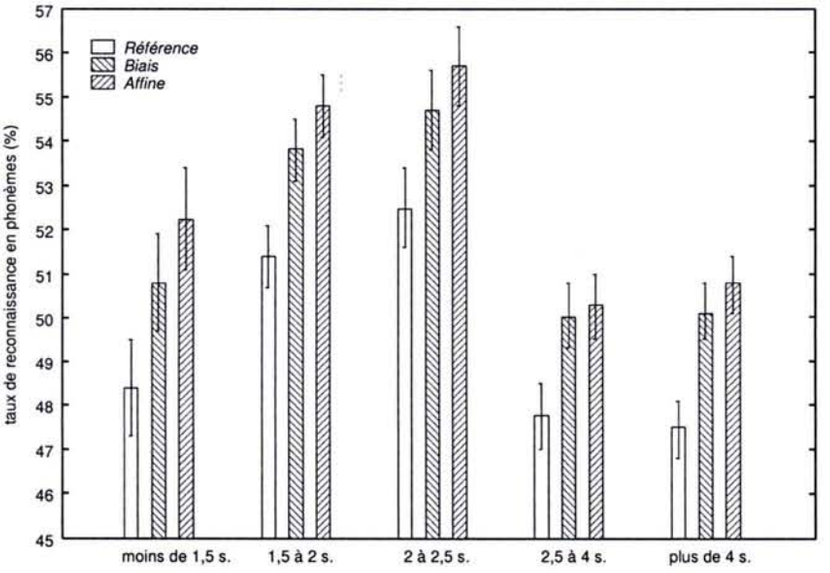


FIG. 5.7 – Comparaison des taux de reconnaissance en phonèmes sur toute la base de test VODIS *close-talk* entre *Référence* et *Biais*.

ne varient pas lorsque l'on fait varier le facteur d'oubli. Ceci est logique puisque l'environnement acoustique varie peu (pas) pendant la durée de la phrase.

Cette expérience permet de mettre en évidence un autre point fort de notre méthode. En effet, ici les données de test ne sont pas corrompues par un bruit ambiant et pourtant l'algorithme que nous proposons améliore le taux de reconnaissance. Dans l'expérience que nous venons de proposer, les données de test et d'entraînement ont été collectées dans les mêmes conditions acoustiques. Par conséquent, notre algorithme ne compense pas pour la variabilité d'environnement acoustique. Cependant, il existe une différence notable entre les ensembles d'entraînement et de test : les locuteurs auxquels on a fait appel sont différents d'un ensemble à l'autre. La variabilité inter-locuteurs se situe à plusieurs niveaux du processus de phonation (la différence entre les appareils phonatoire des locuteurs, leur accent, leur âge, ...). Cette variabilité se traduit, au niveau du processus de reconnaissance, par une distance non-nulle entre la réalisation acoustique d'un locuteur de test et le modèle acoustique de cette réalisation entraîné sur un corpus de locuteur excluant le locuteur de test. Grâce à l'expérience que nous venons de présenter nous pouvons dire que notre algorithme (comme tout algorithme basé sur le *Stochastic Matching*) permet de réduire cette distance. Nous pouvons donc conclure que l'algorithme que nous proposons permet de compenser en partie la variabilité inter-locuteurs.

5.3 Convergence et initialisation de l'algorithme de compensation

En section 5.2.2, nous avons mesuré les taux de reconnaissance en phonèmes obtenus par notre algorithme (*Biais*) et un SRAP n'intégrant pas de processus de compensation (*Référence*) sur des phrases de durées croissante. Les mesures des améliorations relatives en fonction de la durée des phrases de test sont représentées dans le tableau 5.1.

	0s à 1.5s	1.5s à 2s	2s à 2.5s	2.5s à 4s	plus de 4s
Acc. Référence	37.2	36.3	35.5	35.6	39.8
Acc. Biais	38.8	38.9	38.1	38.5	42.4
amélioration relative	4.3	7.16	7.3	8.1	6.5
réduction relative du taux d'erreur	2.5	4.1	4.0	4.6	4.3

TAB. 5.1 – Taux de reconnaissance en phonèmes (%) sur toute la base VODIS *far-talk* obtenu par *Biais* selon la durée des phrases de test.

On remarque que l'amélioration apportée est d'autant plus importante que les phrases sont longues. Ceci peut s'expliquer par le fait que l'estimation des paramètres de la fonction de compensation est récursive. L'estimation est d'autant plus précise que les données à disposition pour faire cette estimation sont nombreuses. Or ces données sont fournies par les phrases à compenser/reconnaître. Par conséquent la compensation sur des phrases longues est plus aisée.

Ceci met en évidence le phénomène de convergence auquel est soumis le biais. En effet, comme nous le verrons dans cette section, les valeurs du biais oscillent dans une première phase puis convergent vers une valeur stable (alors que l'influence du deuxième terme de l'équation 5.4 diminue). De fait, la compensation n'est effective que sur des phrases de durée suffisamment longue. Sur des phrases trop courtes, les valeurs du biais ne sont pas stables. On peut dire qu'elles ne reflètent pas correctement l'état de l'environnement acoustique.

5.3.1 Convergence des paramètres de la fonction de compensation

Le phénomène de convergence est observable pour toutes les phrases. La figure 5.8 montre l'évolution des valeurs du biais calculées par *Biais* pour une phrase (*far-talk* de VODIS) bien reconnue.

Seules les trois premières dimensions sont représentées (c_0 à c_2)[24]. On note que, pour toutes les dimensions, le biais passe par une phase d'oscillations importantes. La durée de cette phase varie selon les dimensions et les phrases.

La phase d'oscillations inhérente au calcul est courte (500ms en moyenne) par rapport à la phrase de test. Elle devrait donc être peu préjudiciable à la bonne reconnaissance de la phrase. Nous aurons en section 5.3.3 une discussion permettant d'évaluer l'influence de cette phase de convergence sur les taux de reconnaissance.

5.3.2 Initialisation des paramètres de la fonction de compensation

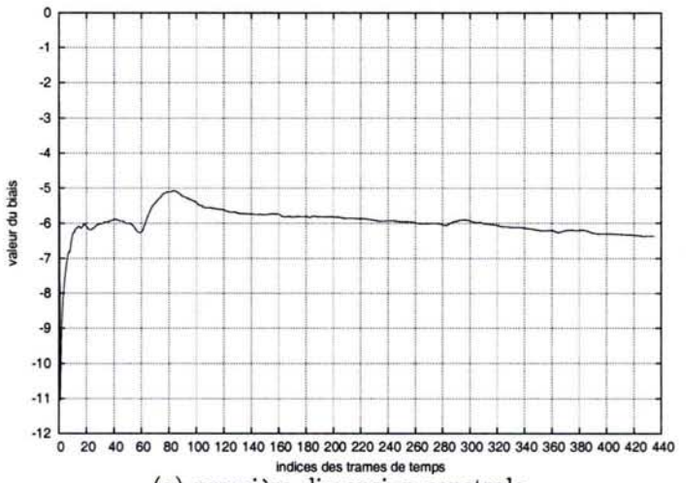
Comme il est dit dans la section 5.1.2, les valeurs initiales des paramètres de la fonction de compensation sont fixées à des valeurs arbitraires au début de chaque séquence d'observations. Après une courte période de fluctuations, le biais converge vers une valeur finale. Afin de réduire cette période de fluctuations plusieurs techniques d'initialisation ont été considérées.

Dans un premier temps, considérons le cas où le système de reconnaissance de la parole est utilisé en continu dans un même environnement. Dans ces conditions, il est légitime de penser que cet environnement ne subira pas de changement brutal entre deux séquences consécutives d'observations. Dans ce cadre particulier, les valeurs initiales des paramètres de la fonction de compensation peuvent être fixées aux valeurs finales obtenues lors de la reconnaissance de la séquence précédente. Ainsi, la convergence est atteinte plus rapidement. La figure 5.9 représente l'évolution du biais sur la seconde dimension de l'espace d'observation (c_1), pour une phrase de la tâche *téléphone* sans utiliser d'initialisation (ligne pleine) et en initialisant le biais avec celui obtenu lors d'une phrase prononcée par le même locuteur, dans un environnement similaire (ligne en pointillés).

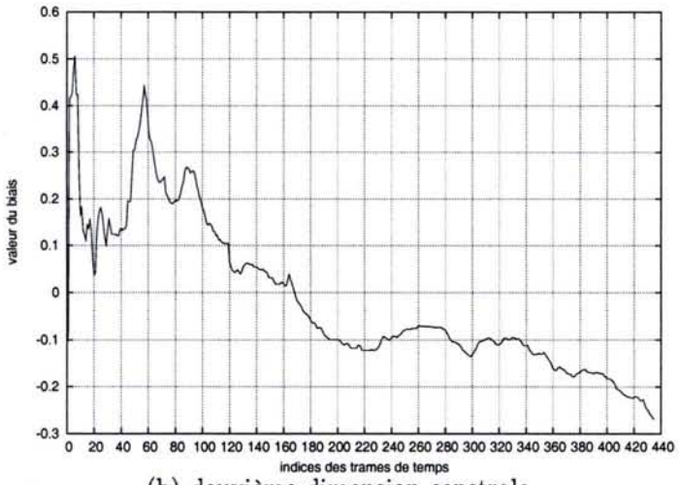
Considérons tout d'abord le cas où la fonction de compensation utilisée dans notre algorithme est un biais (*Biais*). L'application de cette méthode d'initialisation conduit à une amélioration qui n'est pas significative dans le cadre de la tâche *nombres100à15000* de VODIS (chaque phrase de test a une durée de 100 trames en moyenne). En ce qui concerne la tâche *téléphone* (chaque phrase de test a une durée de 600 trames en moyenne), on observe une légère dégradation des performances par rapport à notre *Biais* sans initialisation.

Cette différence de comportement peut s'expliquer de la manière suivante : la convergence des paramètres de la fonction de compensation est *a priori* aussi rapide pour une phrase longue que pour une phrase courte. Cependant la période de stabilité relative de ces paramètres est moins étendue pour une phrase courte. Ainsi, la période sur laquelle le processus de compensation est efficace est réduite pour une phrase brève. En introduisant une initialisation, la période d'oscillation est réduite et la période d'efficacité de la compensation augmentée. Ceci est un atout pour les phrases courtes mais est un apport négligeable pour les phrases disposant déjà d'une grande période de stabilité (phrases longues).

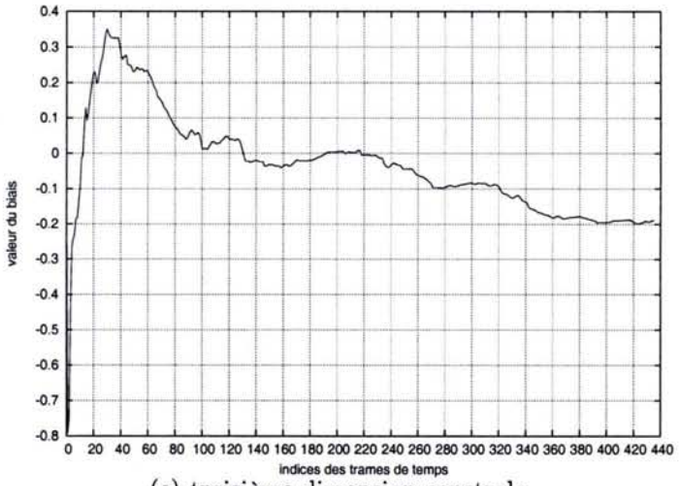
[24] on trouvera au chapitre 1, section 4.3 une description de la paramétrisation utilisée



(a) première dimension cepstrale.



(b) deuxième dimension cepstrale.



(c) troisième dimension cepstrale.

80

FIG. 5.8 – Evolution des valeurs du biais calculées par *Biais* pour une phrase (bruitée) bien reconnue.

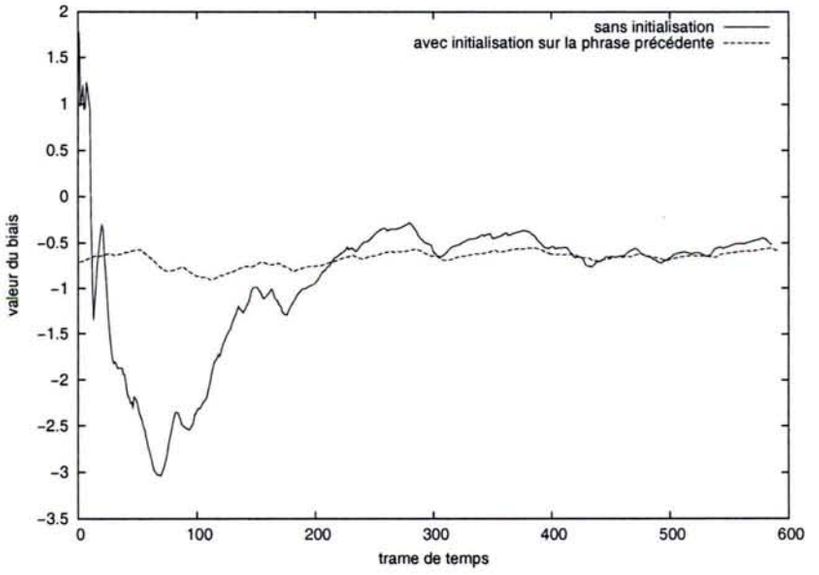


FIG. 5.9 – Influence de l'initialisation sur l'évolution du biais sur la dimension c_1.

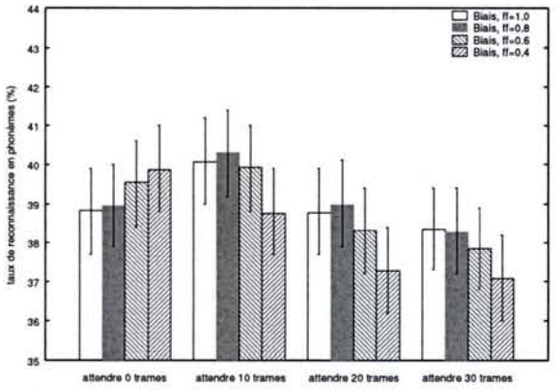
Cette expérience montre que l'initialisation de notre algorithme n'est pas un facteur limitant de son efficacité. En effet, les résultats obtenus lors de l'initialisation de la fonction de compensation à la fonction identité (initialisation du biais à zéro) sont comparables à ceux obtenus lorsque l'initialisation est réalisée à partir de données *a priori*. Par conséquent, dans l'ensemble des expériences exposées par la suite, aucune sorte d'initialisation ne sera utilisée (sauf mention contraire).

5.3.3 Compensation des premières trames d'une séquence

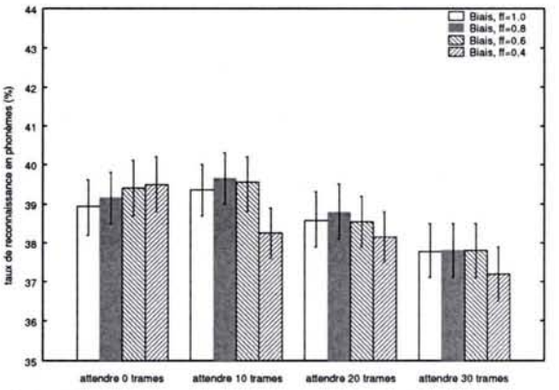
Nous avons vu en section 5.3.1 que les valeurs des paramètres de la fonction de compensation fluctuaient avant de converger. Pendant cette période, l'intervalle de variation peut être très large.

Dans quelle mesure cette phase d'oscillations influence-t-elle la reconnaissance ? Pour donner un élément de réponse, il a été décidé de modifier l'algorithme proposé en section 5.1.2 de sorte que la transformation ne soit effectivement appliquée aux observations acoustiques qu'après une période de temps à définir. A chaque instant t, on estime les paramètres de la fonction mais celle-ci n'est effectivement utilisée pour compenser qu'à $t > T$, où T est une date fixée arbitrairement. La figure 5.10 représente les taux de reconnaissance en phonèmes obtenus par *Biais* sur toute la base de test *far-talk* de VODIS partitionnée selon la durée des phrases. Les résultats sont présentés en fonction du facteur d'oubli (*ff*) utilisé. Notons tout d'abord que, dans leur ensemble, les résultats sont comparables. On remarque ensuite que les maximums obtenus lorsque la période d'attente est de 10 trames ($\approx 250ms$) sont toujours supérieurs aux autres valeurs. De plus, lorsque cette période est de 30 trames, les résultats sont dégradés (significativement inférieurs dans le cas de phrases de courtes durées).

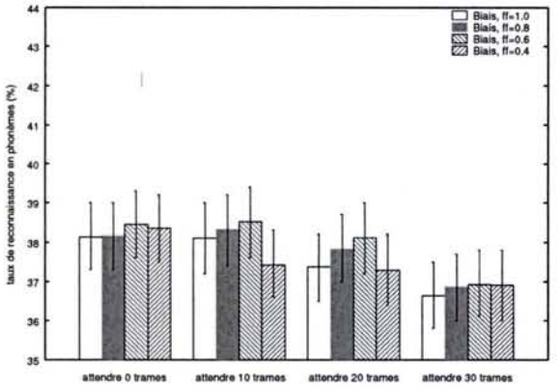
La solution proposée pour éviter que la période de fluctuation n'influence pas le mécanisme



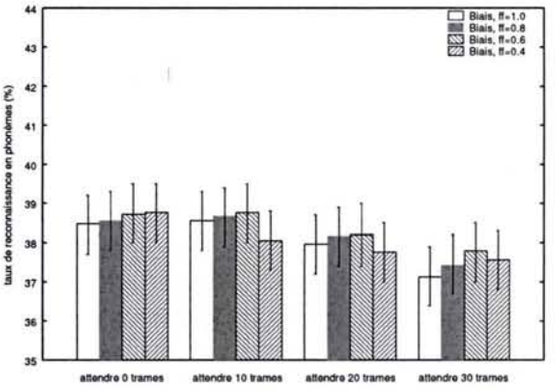
(a) phrases de durée inférieure à 1.5 secondes.



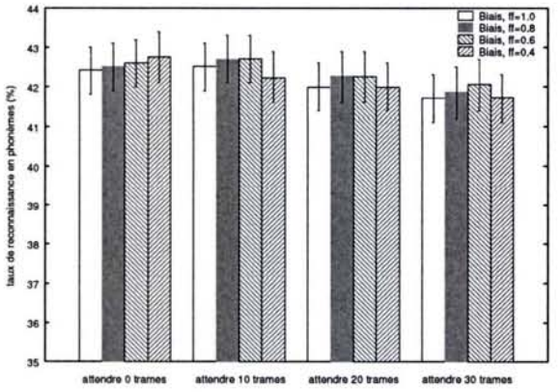
(b) phrases de durée entre 1.5 et 2.0 secondes.



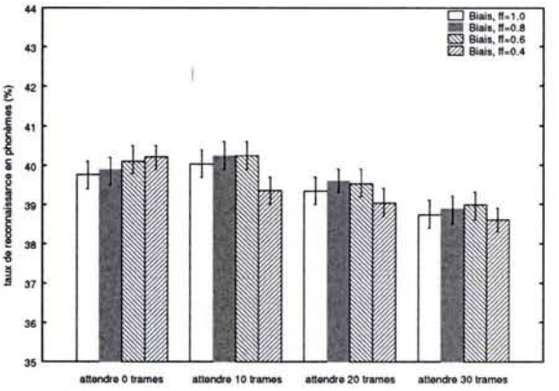
(c) phrases de durée entre 2.0 et 2.5 secondes.



(d) phrases de durée entre 2.5 et 4 secondes.



(e) phrases de durée supérieure à 4 secondes.



(f) Toute les phrases de la base de test

FIG. 5.10 – Taux de reconnaissance en phonèmes de *Biais* sur toutes les phrases de test *far-talk* de VODIS en fonction du facteur d'oubli *ff* et du délai d'attente avant la compensation.

de compensation est donc efficace. En effet, (f) de la figure 5.10 montre que le maximum de reconnaissance sur toute la base de test est obtenu lorsque la fonction de compensation appliquée aux 10 premières trames est l'identité. Cependant, un résultat similaire est obtenu en compensant avec la fonction estimée dès la première trame et en utilisant un facteur d'oubli adapté (ici, ff=0.4). Le facteur d'oubli permet en effet de réduire la durée de la phase de fluctuations. De plus, comme nous le verrons en section 5.4.1, il permet d'effectuer la compensation dans un milieu acoustique variable.

Par conséquent, dans le cadre d'une utilisation réelle, retarder l'application de la fonction de compensation après les premières trames n'apporte qu'un avantage limité. C'est pourquoi, dans la suite de nos expériences, nous avons choisi de n'utiliser que le facteur d'oubli et de compenser dès la première trame.

5.4 Comparaison avec les autres méthodes de compensation

Nous avons comparé les performances de notre méthode aux résultats obtenus par des méthodes de robustesse classiques. Le choix des méthodes utilisées pour l'évaluation n'est pas anodin. En effet, chacune d'elle possède un certain avantage sur les autres selon le cadre applicatif (environnements acoustiques variables, information *a priori* sur la nature du bruit,...). Nous verrons dans cette section que notre algorithme possède des avantages sur toutes ces méthodes et dans la plupart des conditions d'utilisation.

Le but de cette opération est de mettre en évidence les caractéristiques de notre méthode ainsi que sa capacité à compenser en milieux adverses. Cette section s'organise de la façon suivante : dans un premier temps, nous exposerons l'amélioration du taux de reconnaissance en phonèmes par notre approche par rapport à *Référence*. Ceci nous permettra d'étudier l'influence de la taille des séquences à reconnaître sur ces taux de reconnaissance. Dans un deuxième temps, nous comparerons les résultats obtenus par notre méthode et des méthodes classiques utilisées en robustesse.

- Nous comparerons notre méthode à *S-CMN* sur des phrases prononcées dans un environnement acoustique fortement variable. Ceci nous permettra de mettre en évidence que la capacité d'adaptation (de réactivité) de notre algorithme surpasse celle, reconnue, de *S-CMN*.
- Nous étendrons ensuite la comparaison avec *PMC* afin de montrer que notre méthode est aussi efficace, beaucoup moins lourde à mettre en oeuvre et plus réactive aux changements d'environnement acoustique.

Ensuite, nous opposerons notre approche *temps-réel* avec les approches *temps-différé* s'inspirant du *Stochastic Matching*. Nous verrons qu'au-delà d'une simplification des calculs, notre hypothèse apporte au paradigme du *Stochastic Matching* un dynamisme qui induit une amélioration significative des taux de reconnaissance. En conclusion, nous évoquerons les problèmes auxquels pourraient être confrontée notre méthode, notamment ceux liés à l'initialisation de l'algorithme.

5.4.1 Comparaison avec la normalisation de la moyenne cepstrale

Dans une première partie, nous avons comparé les résultats de notre approche avec une méthode classique de compensation synchrone à la trame : la normalisation cepstrale (*S-CMN*).

Dans un environnement variant lentement

Le tableau 5.2 présente les taux de reconnaissance en mots obtenus sur le corpus de test *far-talk* de VODIS, pour la tâche *nombres100à15000*[25]. La méthode *S-CMN* utilisée est intégrée au système de reconnaissance ESPERE[26].

	Référence	*S-CMN*	*Biais*				*S-CMN+Biais*			
ff			1.0	0.8	0.6	0.4	1.0	0.8	0.6	0.4
Acc. (%) int. de conf. :+-1.8	63.5	68.1	72.7	72.8	72.6	72.2	74.5	73.9	74.0	73.0
reduction du taux d'erreur		12.6	25.2	25.5	24.9	23.8	30.1	28.5	28.8	26.0

TAB. 5.2 – Taux de reconnaissance de mots sur les phrases *nombres100a15000 far-talk* de VODIS.

Par de nombreux aspects, cette méthode semble donc avoir les mêmes caractéristiques que notre proposition. Cependant, les tests montrent que notre méthode garde un certain avantage (voir tableaux 5.2 et 5.4). Nous exploitons en effet une information supplémentaire : les valeurs des moyennes des réalisations apprises lors de l'entraînement.

Lorsque l'on compare, dans le tableau 5.2, les résultats obtenus par *S-CMN* et notre algorithme (*Biais*), on peut observer que notre méthode est plus performante. Plus précisément, si *S-CMN* obtient une amélioration du taux de reconnaissance en mots de 7.2% relativement à un système de reconnaissance (ESPERE) n'utilisant pas de méthode de compensation (*Référence* dans le tableau), *Biais* obtient plus de 14%. De plus, on peut montrer que ces deux méthodes ne sont pas incompatibles et peuvent se montrer complémentaires. Si l'on cumule les deux méthodes de reconnaissance *S-CMN* et *Biais*, on obtient une amélioration supplémentaire du taux de reconnaissance (dernières colonnes du tableau 5.2).

Le tableau 5.3 donne les taux de reconnaissance en mots, dans les mêmes conditions, lorsque notre méthode utilise une fonction de compensation affine (*Affine*). On voit ici que l'amélioration

	Référence	*S-CMN*	*Affine*				*S-CMN+Affine*			
ff			1.0	0.8	0.6	0.4	1.0	0.8	0.6	0.4
Acc. (%) int. de conf. :+-1.8	63.5	68.1	72.3	76.5	77.7	78.0	68.6	76.7	76.9	76.2
reduction du taux d'erreur		12.6	24.1	35.6	38.9	39.7	14.0	36.2	36.7	34.8

TAB. 5.3 – Taux de reconnaissance de mots (%) sur les phrases *nombres100à15000 far-talk* de VODIS (intervalle de confiance :+-1.8).

apportée par la fonction affine est conséquente. Cependant, la cumulation de notre méthode *Affine* et la méthode *S-CMN* provoque une dégradation des performances par rapport à *Affine* seule.

[25] voir chapitre 1 pour la description de la base VODIS

[26] voir la section 2.2.1 du chapitre 2 pour une description de *S-CMN*

Dans un environnement variant rapidement

Afin de comparer la réactivité des méthodes *S-CMN* et *Biais*, une base de test artificielle a été créée. La partie *close-talk* de la base de test VODIS a été utilisée comme point de départ. Chacune de ses phrases a ensuite été corrompue par un signal ajouté dans le domaine temporel. Ce signal (bucaneer2.wav de la base NOISEX) a été modulé de 3 manières différentes pour former 3 bases de test :

linear le bruit croît de façon constante du début à la fin de la phrase ((b) sur la figure 5.11).

triangle le bruit croît de façon constante jusqu'à son milieu puis décroît de façon constante ((c) sur la figure 5.11).

invLin le bruit débute avec la phrase puis décroît de façon constante ((d) sur la figure 5.11).

Ces expériences ont été élaborées pour deux raisons :

- simuler des événements pouvant intervenir dans un environnement automobile (le bruit d'un point fixe que l'on dépasse).
- il s'agit de sources de bruits très difficiles, variables et inconnues

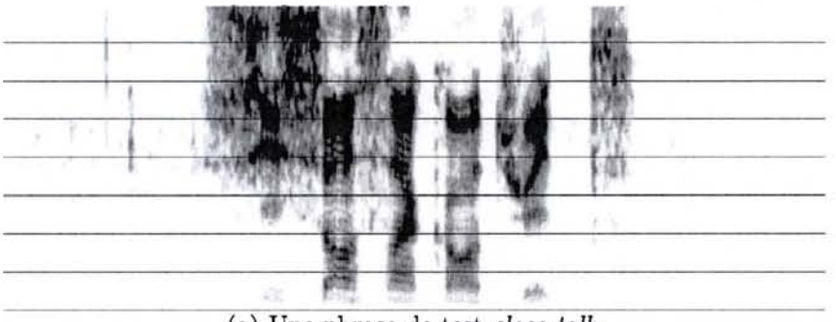
Les spectrogrammes présentés dans la figure 5.11 ont été calculés pour des phrases de chaque base de test constituée artificiellement.

Les résultats obtenus par notre proposition ainsi que par *S-CMN* sont présentés dans le tableau 5.4.

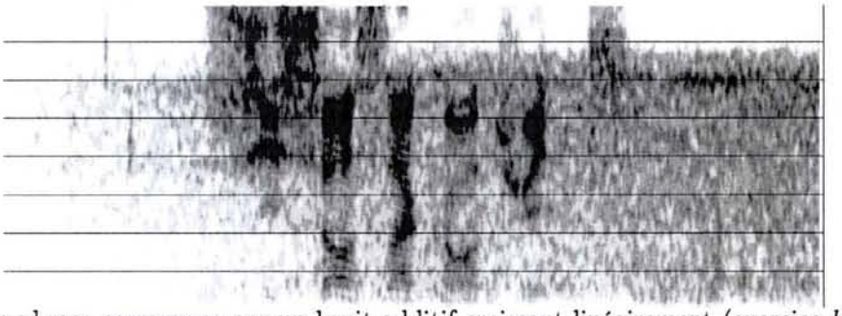
		WER (int. de conf. :+-1.8)	*Réduction du WER*		
		Référence	*S-CMN*	*Biais*	*Affine*
clean	(RSB : 20.3 dB)	10.1	3.0	19.8	9.9
linear	(RSB : 16.8 dB)	32.7	49.8	45.9	49.2
triangle	(RSB : 17.7 dB)	24.4	23.4	27.5	33.6
invLin	(RSB : 16 dB)	49.5	3.8	53.3	58.4

TAB. 5.4 – Réduction du taux d'erreur en mots (%) sur le corpus de test *nombres100à1500* de VODIS artificiellement bruité .

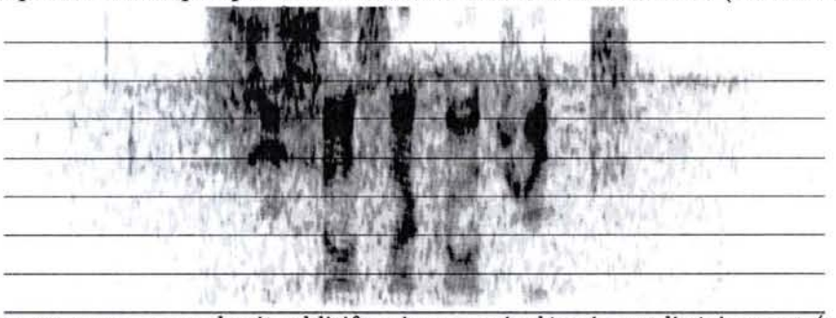
Cette expérience met en évidence deux points forts de notre méthode : d'une part, la méthode de compensation que nous proposons semble plus réactive aux variations de l'environnement acoustique. En effet, pour un environnement fortement variable comme celui de l'exercice *triangle*, notre approche donne des résultats légèrement supérieurs. D'autre part, elle est beaucoup plus robuste à une mauvaise initialisation. Ceci est mis en évidence par la dernière expérience (invLin), où les valeurs des premiers cepstres de la séquence à reconnaître sont très éloignées des valeurs observées pour l'initialisation (généralement, ces valeurs d'initialisation sont obtenues lors de l'entraînement). En effet, les premières trames de parole sont fortement corrompues par le signal ajouté. Par conséquent, la normalisation par la moyenne cepstrale calculée sur ces premières trames est mauvaise, puisqu'elle se base sur de mauvaises valeurs initiales de moyenne. Notre méthode s'est affranchie de ce problème d'initialisation puisque le calcul de la fonction de compensation, dans ce cas, se fonde plus sur les informations fournies par la distance des observations aux modèles (des données qui reflètent donc l'état de l'environnement acoustique) que sur des données d'initialisation (une donnée *a priori*).



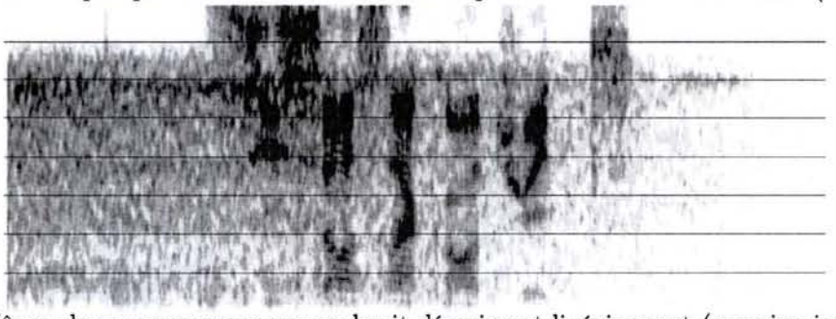
(a) Une phrase de test *close-talk*.



(b) Même phrase corrompue par un bruit additif croissant linéairement (exercice *lineaire*).



(c) Même phrase corrompue par un bruit additif croissant puis décroissant linéairement (exercice *triangle*).



(d) Même phrase corrompue par un bruit décroissant linéairement (exercice *invLin*)

FIG. 5.11 – Exemple de phrases de test du corpus VODIS artificiel.

5.4.2 Comparaison avec Parallel Model Combination

Les performances de notre approche ont été comparées avec celles obtenues lorsqu'on utilise l'*adaptation* comme mécanisme de robustesse. Pour cela, nous avons utilisé la version d'ESPERE utilisant la méthode du *PMC*[27], développée par François Chaffard, de DIALOCA.

Les taux de reconnaissance en mots obtenus sur la partie *far-talk* de la base de test PhoneticalNumbers de VODIS sont exposés dans la table 5.5.

	Référence	*PMC*	*Biais*				*PMC + Biais*			
ff			1.0	0.8	0.6	0.4	1.0	0.8	0.6	0.4
Acc. (%) int. de conf. :+-1.8	63.5	72.8	72.7	72.8	72.6	72.2	72.6	72.4	72.4	72.6
reduction du taux d'erreur		25.5	25.2	25.5	24.9	23.8	24.9	24.4	24.4	24.9

TAB. 5.5 – Réduction du taux d'erreur en mots (%) sur les phrases *nombres100à15000 far-talk* de VODIS.

On peut voir ici que les résultats des deux méthodes sont proches et que leur cumulation n'apporte rien.

Le tableau 5.6 représente les taux de reconnaissance en mots pour la tâche *nombres100à15000 far-talk* de VODIS pour un système de reconnaissance utilisant *PMC*, notre méthode de compensation utilisant une fonction affine (*Affine*) et enfin le cumul des deux. Dans les cas où nous utilisons notre méthode, le facteur d'oubli *ff* varie de 1.0 à 0.4.

	Référence	*PMC*	*Affine*				*PMC + Affine*			
ff			1.0	0.8	0.6	0.4	1.0	0.8	0.6	0.4
Acc. (%) int. de conf. :+-1.8	63.5	72.8	72.3	76.5	77.7	78.0	65.9	75.1	75.8	76.0
reduction du taux d'erreur		25.5	24.1	35.6	38.9	39.7	6.6	31.8	33.7	34.2

TAB. 5.6 – Réduction du taux d'erreur en mots (%) sur les phrases *nombres100à15000 far-talk* de VODIS.

Les résultats obtenus par notre méthode sont significativement supérieurs aux résultats obtenus par *PMC* malgré une charge de calcul moins importante. Dans tous les cas, on n'observe pas d'amélioration lorsque les deux méthodes sont cumulées.

La remarque principale que l'on peut formuler au vu de ces deux expériences est qu'il n'y a pas d'amélioration lorsque l'on utilise conjointement notre approche et la *PMC*. Rappelons ici que *PMC* consiste grossièrement en une translation des moyennes des densités de distribution de probabilité d'observation contenues dans les modèles acoustiques. Quant à elle, notre méthode modifie les moyennes des distributions des observations.

[27] les détails de l'implémentation de *PMC* sont introduits dans la section 2.3.1 du chapitre 2

Lors de la reconnaissance, ces deux phénomènes agissent donc dynamiquement et indépendamment l'un de l'autre sur une même grandeur : la distance d'une observation aux modèles acoustiques. Puisqu'elles sont indépendantes, aucune des deux méthodes ne prend en compte l'action de l'autre.

Pour conclure, nous pouvons dire que notre méthode est incompatible avec une méthode d'adaptation dynamique des modèles acoustiques. Il est impossible de combiner les deux approches sans une perte de performance.

5.4.3 Comparaison avec une approche en *temps-différé*

Dans cette section, nous allons mettre en évidence l'apport d'une approche *temps-réel* par rapport à une implémentation de type en *temps-différé* (*off-line*). L'implémentation *temps-différé* utilisée pour cette comparaison est très simple. Elle met pourtant en évidence certains points forts de notre approche. Pour cela, nous avons créé une série d'expériences qui tend à dissocier les mécanismes de compensation et de reconnaissance.

Compensation utilisant un alignement forcé

Dans une première série d'expériences, nous avons utilisé les séquences d'états optimums obtenus par une première phase de reconnaissance pour obtenir des séquences de transformations. Ces séquences de transformations ont permis de compenser les phrases de test. La figure 5.12 donne le détail de l'opération.

Comme décrit dans cette figure, nous avons défini 3 ensembles de test (Ensembles A, B et C) à partir de la base de test VODIS :

Ensemble A Cet ensemble regroupe les phrases de test ayant été correctement reconnues par ESPERE, sans mécanisme de compensation, lorsqu'on utilise le canal *close-talk*.

Ensemble B Cet ensemble contient les réalisations *far-talk* des phrases de test retenues dans *Ensemble A*. *Ensemble B* regroupe donc les mêmes phrases que *Ensemble A* mais les séquences de *Ensemble A* ont été enregistrées sur le canal *close-talk*, et celles de *Ensemble B*, sur le canal *far-talk*.

Ensemble C Cet ensemble est obtenu par transformation de *Ensemble C* comme décrit plus bas.

Une première passe de reconnaissance est effectuée avec ESPERE sans mécanisme de compensation (*Référence*) sur *Ensemble A*. A la fin de cette première passe nous avons donc obtenu une séquence d'états optimum, pour chaque phrase de test retenue. Ces séquences d'états représentent effectivement les séquences de phonèmes prononcés. Grâce à ces séquences, aux modèles acoustiques et aux phrases de *Ensemble B*, des séquences de transformations ont été calculées pour chaque phrase de test de *Ensemble B*, selon le principe évoqué dans l'équation 5.4. Enfin, chaque phrase de *Ensemble B* a été transformée par la séquence de transformations pour former *Ensemble C*. L'opération permettant d'obtenir *Ensemble C* s'appelle un *alignement forcé*.

Le tableau 5.7 présente les taux de reconnaissance en phonèmes sur *Ensemble B* sans utiliser notre mécanisme de compensation (première colonne), en l'utilisant (simple biais, deuxième colonne) et sur *Ensemble C* sans utiliser de compensation.

On remarque qu'un alignement forcé exact permet d'obtenir de bonnes valeurs de biais. En effet, les résultats obtenus sur *Ensemble C* sont supérieurs à ceux obtenus par notre méthode de compensation *temps-réel* sur *Ensemble B*.

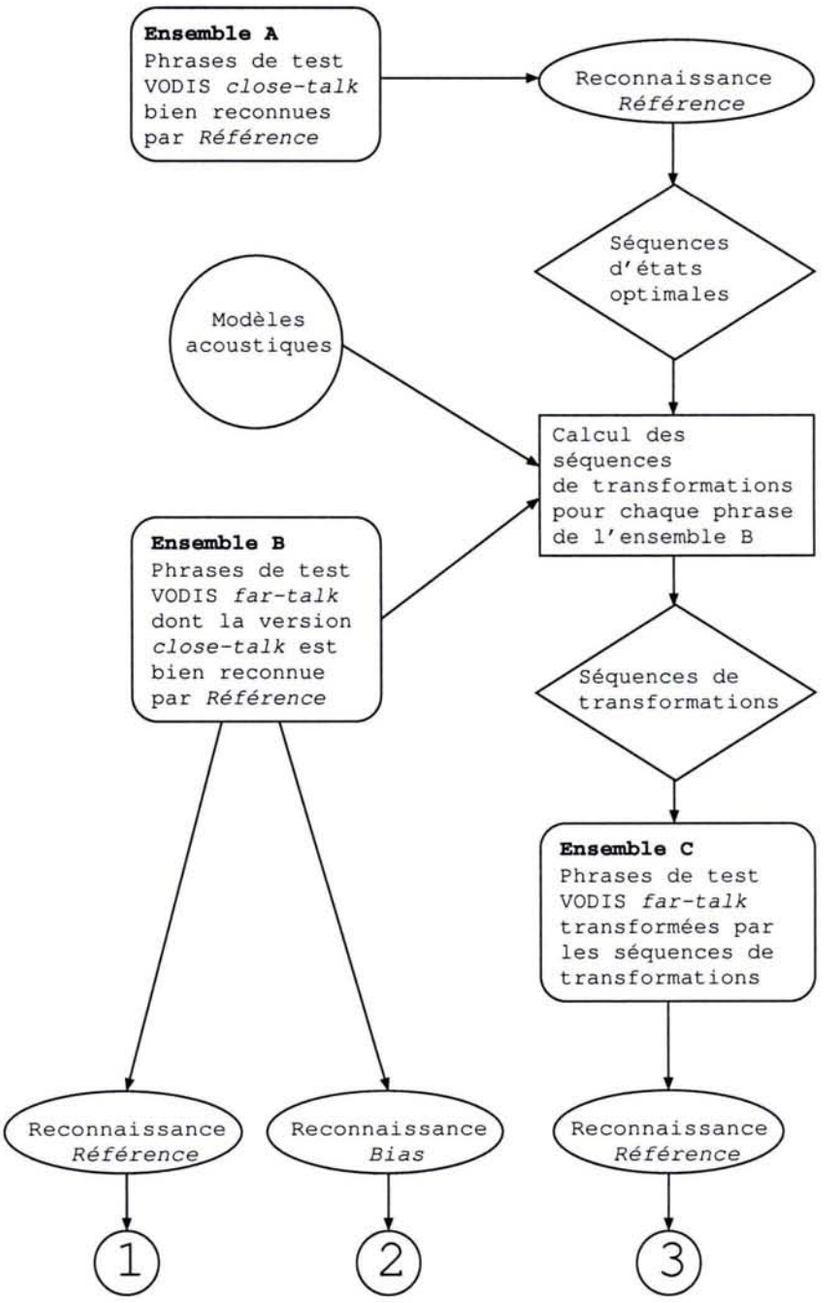


FIG. 5.12 – Cadre expérimental de l'alignement forcé.

	Référence sur *Ensemble B*	*Biais* sur *Ensemble B*	*Référence* sur *Ensemble C*
Acc. (%)	47.4	52.3	56.3

TAB. 5.7 – Taux de reconnaissance en phonèmes (%) sur la partie *far-talk* pour les phrases bien reconnues par *Référence* en *close-talk* (VODIS).

Description de l'approche *temps-différé* choisie

Pour confirmer le résultat précédent, nous utilisons comme ensemble de test, non pas les phrases bien reconnues mais toutes les phrases de test. Ce mode opératoire correspond en fait à une approche *temps-différé* possible du *Stochastic Matching* dans sa plus simple expression. Cette approche *temps-différé* est celle utilisée dans [Sankar and Lee, 1996].

Les séquences d'états optimum obtenus sont issues de phrases bien reconnues et mal reconnues. Qu'advient-il lorsque l'ensemble de test bruité est compensé par les séquences de transformations issues de ces chemins optimum ? Les résultats de cette expérience sont retranscrits dans le tableau 5.8.

	Référence sur *far-talk*	Biais en *temps-réel* sur *far-talk*	*Biais* en *temps-différé* sur *far-talk*
Acc. (%)	32.0	34.5	33.0

TAB. 5.8 – Taux de reconnaissance en phonèmes (%) sur toute la partie *far-talk* de VODIS (intervalle de confiance : +-0.4).

On peut voir que c'est la version *temps-réel* qui obtient les meilleurs résultats. Dans ce cas, les biais calculés *temps-différé* sont estimés à partir de séquences d'états ne correspondant pas à ce qui a été prononcé. Par conséquent les phrases de test modifiées par ces biais ne sont pas compensées mais bruitées par le processus de transformation. En effet, la transformation a pour but de rapprocher la séquence d'observations acoustiques de la séquence des états optimum calculée lors d'une première passe.

De ces deux expériences, nous pouvons tirer les conclusions suivantes :

- Pour une approche *temps-différé*, le processus de reconnaissance et le processus de la compensation sont dissociés. Or, l'approche *Stochastic Matching* nécessite une séquence d'état représentant la séquence acoustique prononcée à compenser. Dans le cas où la séquence d'états fournie est mauvaise, la compensation peut ne pas être efficace [Lee, 1998]. C'est ce phénomène qui a été mis en évidence dans le tableau 5.8.
- Pour une approche *temps-réel*, les processus de reconnaissance et de compensation s'effectuent en parallèle. Il existe donc une égalité permanente entre le chemin des états optimum (au sens de la probabilité *avant*) de la reconnaissance et le chemin des états utilisés par la compensation. C'est cette cohérence qui permet à l'algorithme *temps-réel* d'obtenir des résultats supérieurs à la version *temps-différé*.

5.5 Confirmation des observations

5.5.1 Sur la base VODIS

Les résultats proposés ici en guise de résumé des résultats des section précédentes ont été consignés dans [Barreaud *et al.*, 2003a]. Il s'agit d'une simple comparaison des taux de reconnaissance en mots obtenus par notre approche et par les méthodes classiques. Le tableau 5.9 représente les résultats obtenus sur la base VODIS (tâches *nombres100à15000* et *téléphone*). Dans cette expérience, des modèles acoustiques monophones sont entraînés sur de la parole propre[28]. Les résultats proposés sont les taux de reconnaissance en mots obtenus :

- si aucune méthode de compensation n'est appliquée (*Référence*),
- si les méthodes de compensation classiques appliquées sont :
  - soustraction spectrale (*SS*, section 2.2.1)
  - normalisation cepstrale séquentielle (*S-CMN*, section 2.2.1)
  - adaptation par calcul parallèle (*PMC*, section 2.3.1)
- si l'on utilise notre algorithme pour deux fonctions de compensation
  - avec pour fonction de compensation un simple biais (*Biais*)
  - avec pour fonction de compensation une fonction affine (*Affine*).

On précise pour chaque méthode si elle est synchrone à la trame ou non.

synchrone à la trame		non	oui			
	Référence	*PMC*	*S-CMN*	*SS*	*Biais* (ff=0.8)	*Affine* (ff=0.8)
nombres100à15000	63.5	72.8	67.3	72.1	72.8	**76.5**
téléphone	78.6	81.6	80.8	79.3	**83.5**	**86.3**

TAB. 5.9 – Taux de reconnaissance en mots (%) sur les corpus de test *far-talk* (RSB moyen : 10.8dB).

On peut remarquer que notre méthode surpasse les méthodes classiques pour les deux tâches. Pour la tâche de reconnaissance de nombres (*nombres100à15000*), l'utilisation d'une fonction de compensation de type *Affine* donne une amélioration du taux d'erreur de 13.3% par rapport à *PMC*, de 15.5% par rapport à la soustraction spectrale et de 27.8% par rapport à la version synchrone à la trame de la normalisation cepstrale. En ce qui concerne la tâche *téléphone*, notre méthode employée avec une fonction de compensation réduite à un biais (*Biais*) donne une amélioration du taux d'erreur de 10.5 % par rapport à *PMC*, 20.2% par rapport à *SS* et 14.1% par rapport à *S-CMN*.

5.5.2 Sur la base Aurora3

Afin de confirmer les résultats obtenus par notre méthode sur la base de test VODIS, une expérience similaire a été conduite sur la partie finnoise d'Aurora3[29]. La base de test se divise en trois tâches :

High Mismatch (HM) : les modèles acoustiques sont entraînés sur de la parole propre et le test s'effectue dans l'environnement bruité.

[28] voir le chapitre 4, section 4.2 pour la description des tâches de reconnaissance

[29] on trouvera une description de la base au chapitre 4

Medium Mismatch (MM) : les modèles acoustiques sont entraînés dans un milieu bruité différent du milieu de test.

Well Matched (WM) : les modèles acoustiques sont entraînés dans le même environnement que celui du test.

Par définition, la tâche HM correspond à l'exercice auquel nous nous sommes livrés sur VODIS. En effet, Aurora3 et VODIS sont deux bases enregistrées dans un environnement automobile et les expériences menées sur VODIS l'ont été avec des modèles acoustiques entraînés en milieu calme (grâce au micro *close-talk* sur des données bruitées (enregistrées avec le micro *far-talk*).

Nous avons donc appliqué notre méthode de compensation sur cette partie de la base de test. Nous avons utilisé pour cela comme fonction de compensation un biais simple et un facteur d'oubli de 1.0 (*Biais*). Ces résultats sont représentés dans le tableau 5.10. Les résultats obtenus par notre méthode pour HM montrent une amélioration relative du taux de reconnaissance de 24.6% par rapport à un système de reconnaissance n'intégrant pas de méthode de compensation (*Référence*).

	HM	MM	WM
Acc. Référence (%)	46.2	75.0	88.8
Acc. Biais (%)	57.6	72.1	89.0
interval de confiance	+-1.9	+-2.3	+-0.7

TAB. 5.10 – Taux de reconnaissance en phonèmes (%) sur les trois tâches Aurora3 obtenu par *Biais*.

En ce qui concerne les deux autres tâches (WM et MM), la différence de score n'est pas significative. On observe cependant une légère dégradation du taux de reconnaissance sur la tâche MM avec notre méthode. Ce recul peut s'expliquer ainsi : les modèles entraînés en milieu bruité sont peu précis. En effet, les distributions de probabilités d'émissions sont très larges et les limites des mots sont mal reconnues. De plus, dans ce cas précis, les modèles sont peu représentatifs du milieu de test. Or, dans le cadre du *Stochastic Matching*, nous estimons les valeurs du biais en utilisant la distance des observations aux modèles acoustiques les plus probables au sens des probabilités *avant*. Pour MM, les observations sont aussi éloignées des modèles qu'elles le sont dans HM, les distances sont donc du même ordre. Cependant, ici, les modèles acoustiques sont moins discriminants que dans HM (du fait de l'élargissement des densités). Par conséquent l'élection par les probabilités *avant* des modèles les plus probables est moins précise (les probabilités *avant* associées à chaque modèle sont semblables). Cette combinaison explique la baisse de performance obtenue sur HM. Dans le cas de WM, l'entraînement et le test sont réalisés dans les mêmes conditions. Les distances des observations aux modèles sont plus faibles et des probabilités *avant* plus discriminantes : le biais est bien estimé.

Une expérience similaire a été conduite avec pour fonction de compensation une fonction affine. La figure 5.13 représente les scores maximums obtenus (par rapport au facteur d'oubli *ff*) lorsqu'un biais (*Biais*) et une fonction affine (*Affine*) ont été utilisés comme fonction de compensation. On peut voir que, dans ce cas, *Affine* apporte une amélioration significative par rapport à *Référence*, quelle que soit la tâche considérée. L'amélioration apportée par *Affine* sur *Biais* est significative dans le cas des tâches MM et WM mais on note une légère dégradation (non significative) pour la tâche HM. Cette dernière observation est aussi en accord avec ce qui a été observé sur la base VODIS.

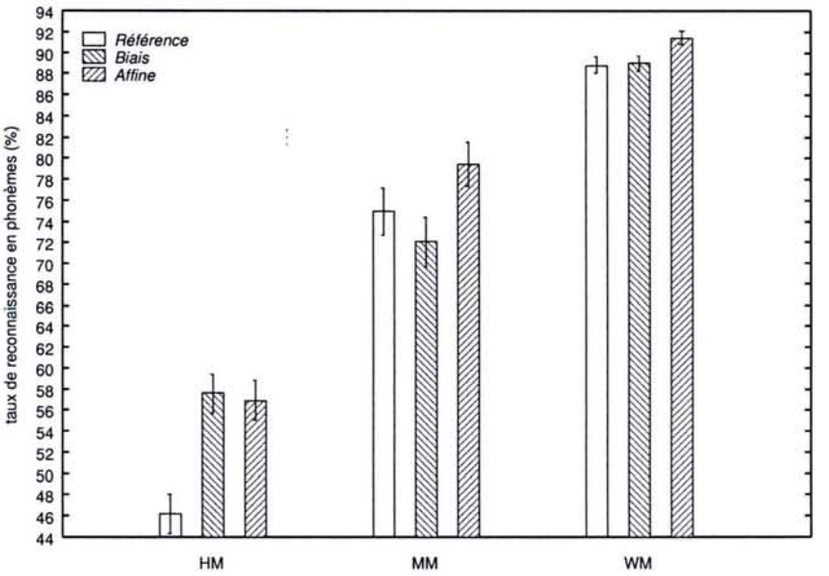


FIG. 5.13 – Comparaison des taux de reconnaissance sur Aurora3 entre *Référence*, *Biais* et *Affine*.

5.6 Conclusion

Dans ce chapitre, nous avons décrit un algorithme de compensation en *temps-réel* basé sur l'approche *Stochastic Matching*. Ce processus de compensation possède des qualités qui n'apparaissent pas (ou partiellement) dans les plus courantes des méthodes de robustesse. Parmi ces qualités, on trouve que :

- la robustesse est accrue grâce à l'interaction en temps réel entre la compensation et le reconnaissance.
- la compensation se passe d'information *a priori* et permet donc d'opérer dans divers environnements acoustiques.
- l'algorithme permet d'effectuer une compensation dans un environnement acoustique variant lentement.

Ces propriétés ont été mises en évidence par une série d'expériences en conditions réelles, sur deux bases de test différentes.

Dans les chapitres suivants, nous nous emploierons à exploiter les propriétés de cette approche de la robustesse. Dans un premier temps, nous verrons comment il est possible de faire évoluer notre algorithme afin d'obtenir une structure hiérarchique de fonctions de compensation spécifiques à des sous-ensembles de l'espace acoustique. Puis nous verrons comment faire en sorte que le processus de compensation décrit dans ce chapitre puisse être utilisé dans un environnement acoustique variant abruptement et inopinément.

6

Structure hiérarchique de transformations de compensation

Sommaire

Afin d'améliorer les résultats obtenus par la méthode présentée au chapitre précédent, nous avons proposé une fonction de compensation utilisant un arbre binaire de transformations [Barreaud *et al.*, 2003b]. Cette approche est motivée par plusieurs constatations.

Il est reconnu que des observations acoustiques similaires sont affectées de la même façon si elles sont émises dans le même environnement acoustique. Par contre, toutes les réalisations acoustiques ne vont pas être influencées de la même manière par un même environnement. Par exemple, dans un même environnement acoustique, deux voyelles seront affectées d'une certaine façon alors qu'une consonne plosive le sera d'une autre. Par conséquent, un ensemble de fonctions de compensation spécifiques à des sous-ensembles de l'espace d'observation doit fournir de meilleurs résultats qu'une seule et unique fonction de compensation.

Cependant, une fonction de compensation semblable à celle que nous avons proposé mais spécifique à un sous-ensemble acoustique, rencontre un problème majeur : si la phrase à compenser comporte très peu d'observations dans un de ces sous-ensembles, la transformation associée à ce sous-ensemble sera mal estimée. Pour résoudre ce problème, nous organisons hiérarchiquement l'ensemble des fonctions spécifiques aux sous-ensembles. Ainsi, lorsqu'une observation doit être compensée, on pourra utiliser la transformation associée au plus petit sous-ensemble acoustique la contenant (un nœud de l'arbre). Si cette dernière n'est pas bien estimée, on utilisera la transformation associée à l'ensemble acoustique qui le contient, c'est à dire le noeud père.

Ce chapitre s'organise de la façon suivante : d'abord, nous présenterons en introduction où réside l'intérêt d'utiliser un arbre de fonction de compensation. Puis nous étudierons la construction de cet arbre. Ensuite, nous présenterons une série d'expériences permettant de valider notre approche. Enfin, nous proposerons les résultats obtenus par cette méthode en conditions de test.

6.1 Introduction

6.1.1 Motivation

Il est reconnu que deux réalisations acoustiques semblables sont affectées de manière similaire si elles sont prononcées dans le même environnement. Par exemple, du point de vue de la perception, deux phonèmes proches comme le **s** de *su* ([**s y**]) et le **z** de *zut* ([**z y t**]) seront mal perçus si ils sont émis dans un bruit blanc (qui s'apparente à un sifflement). En effet, il existe dans ce cas un effet de masquage. Le spectre du bruit blanc recouvre partiellement celui des fricatives. D'un autre côté, les phonèmes **O** de *port* ([**p O R**]) et le **o** de *peau* ([**p o**]) seront moins affectés par ce bruit blanc (si celui-ci reste à un niveau faible par rapport au signal de parole).

La corruption d'une unité acoustique est fonction des indices acoustiques pertinents qu'elle présente (les formes de son spectre) et du spectre des bruits de fond et de canal. On peut voir dans des ouvrages comme [Calliope, 1989] ou [Landercy and Renard, 1982] que les phonèmes peuvent se regrouper en classes articulatoires mais aussi en classes acoustico-perseptives. Ainsi, il est possible de dire qu'un phonème est plus semblable (ou proche), du point de vue spectral, qu'un autre et ainsi définir une *localisation phonétique*. Donnons ici deux exemples pour illustrer ce propos :

- Les phonèmes **p** de *port* ([**p O R**]) et **b** de *bord* ([**b O R**]) sont tous deux des *occlusives labiales non-nasalisées* ou *orales*. La caractéristique qui les distingue est que la première est non-voisée alors que la deuxième l'est. Le spectre de **b** présente donc des indices de voisement qui n'apparaissent pas dans le cas de **p**. Si la région du spectre comportant ces formes est recouverte par la présence d'un bruit, les caractéristiques permettant de discerner ces deux types de réalisation deviennent moins évidentes.
- Les phonèmes **i** de *pire* ([**p i R**]) et **y** de *pur* ([**p y R**]) se distinguent l'un de l'autre par la position de leur deuxième et surtout de leur troisième formant (le premier formant étant aux alentours de 250Hz pour les deux types de réalisation). Si ces sons sont produits dans un bruit de fond couvrant les fréquences supérieures à 1500Hz, les bandes de fréquences contenant les deuxièmes formants seront recouvertes. Du point de vue perceptif, il sera impossible de distinguer une réalisation de l'autre.

De ces exemples, on peut conclure qu'un mécanisme de robustesse doit tenir compte des différentes sensibilités des réalisations acoustiques par rapport à la nature du bruit. C'est pourquoi de nombreuses techniques de robustesse effectuent un traitement distinct selon la localisation

phonétique des observations traitées. C'est le cas par exemple pour des techniques utilisant des dictionnaires (*code-books*) comme le *CDCN* de [Stern *et al.*, 1992], ou encore des techniques utilisant des fonctions de compensation non-linéaires.

Par conséquent, la proposition que nous avons faite dans le chapitre 5 est de toute évidence sous-optimale. Nous avons en effet proposé un algorithme de compensation appliquant une fonction de transformation aux observations acoustiques sans distinction de leur localisation phonétique.

Dans ce chapitre, nous allons substituer un ensemble de transformations spécifiques à des sous-ensembles de l'espace d'observation à l'unique transformation que nous employions dans notre algorithme. Pour transformer une observation acoustique y_t à l'instant t, notre algorithme ne disposera donc pas d'une seule transformation mais d'un ensemble de transformations. Comme précédemment, la forme de ces transformations est fixe (une fonction affine ou un simple biais). De même, les paramètres de ces fonctions sont calculés de manière récursive, d'une façon similaire à celle décrite dans le chapitre 5. Cependant, ces transformations sont spécifiques à la localisation de la réalisation acoustique dans l'espace des observations.

6.1.2 Collection de transformations spécifiques à un état

Dans la section précédente, nous avons proposé d'utiliser une collection de transformations de compensation spécifiques à un espace acoustique plutôt qu'une unique fonction. Cette approche soulève deux questions :

- Quelle partition de l'espace d'observation doit-on utiliser ?
- Comment rendre les transformations spécifiques aux sous-espaces obtenus ?

Une réponse à ces deux questions peut être tirée des deux constations suivantes :

1. Les probabilités d'émission des états des modèles acoustiques fournissent une partition de l'espace d'observation [Beatie and Young, 1991].
2. L'hypothèse-clé de notre approche, déjà avancée dans le chapitre 5, pose qu'à chaque instant lors de l'alignement de Viterbi, l'état associé à la plus grande probabilité *avant* donne une bonne modélisation du vecteur de cepstre observé y.

La première constatation se justifie de la manière suivante : les paramètres des modèles acoustiques utilisés lors de la reconnaissance sont entraînés (par l'algorithme de Baulm-Welch, par exemple) afin qu'ils modélisent les unités acoustiques sur lesquels ils sont estimés. Ainsi, les densités de probabilités d'émission des états de ces modèles acoustiques représentent la répartition de vecteurs acoustiques dans l'espace des observations. On peut donc dire que la probabilité d'émission d'un vecteur acoustique x calculée à partir de la densité d'observation tirée d'un état d'un modèle acoustique MA représente une distance de cette observation x à la région acoustique ayant servie à entraîner MA. Par conséquent, il est possible de classifier un vecteur d'observation x donné dans une zone acoustique en calculant la plus courte distance entre x et l'ensemble des modèles acoustiques. Remarquons qu'il découle de ce raisonnement que les zones de classification obtenues ne sont pas forcément disjointes.

Par conséquent, il existe un moyen simple d'associer à chaque observation acoustique une transformation qui soit spécifique d'un sous-espace d'observation. Pour cela, il faut associer une fonction de compensation à chaque état des modèles acoustiques. En effet, d'après la première constatation, on obtient un ensemble de fonctions de compensation chacune spécifique à un sous-espace d'observation (ces sous-espaces pouvant se recouvrir).

De plus, notre algorithme nous donne la possibilité d'associer à chaque observation acoustique une des fonctions de compensation de l'ensemble défini plus haut, puisqu'à chaque instant un état est déclaré comme le plus représentatif de cette observation.

Voici l'algorithme que nous proposons (nous utilisons un biais comme fonction de compensation) :

1. initialisation : $t := 0$ et association à chaque état d'un ensemble de paramètres décrivant une fonction de compensation (un simple biais dans le cas présent) que l'on initialise.
2. à l'indice de temps t,
   (a) pour chaque état $n \in 1, \ldots, N$ calcul de la transformée de l'observation y_t par la fonction de compensation spécifique à l'état n : $y_t + b^n_{t-1}$
   (b) pour chaque état $n \in 1, \ldots, N$ calcul de
   $$\alpha_{t|B_{t-1}}(n) := p(y_t + b^n_{t-1}, s_t = n|B_{t-1}),$$
   la probabilité *avant* associée à chaque couple (état ; observation transformée) pour l'alignement de Viterbi.
   (c) Choix de l'état $n'_t \in 1, \ldots, N$, le plus probable au sens de la probabilité *avant* à l'instant t.
   $$n'_t = \arg \max_{n \in 1,\ldots,N} \alpha_{t|B_{t-1}}(n)$$
3. à l'indice de temps t, réestimation des paramètres des fonctions de compensation en fonction de la distance de y_t à n'_t comme décrit dans l'équation 6.1
4. $t := t + 1$

En ce qui concerne l'estimation des paramètres des fonctions de compensation, elle s'effectue d'une façon similaire à celle décrite dans le chapitre précédent, de la façon suivante : dans le cas où un simple biais b_t est utilisé, pour chaque état $n \in 1, \ldots, N$,

$$b^n_{t+1} = \begin{cases} b^n_t - \dfrac{\frac{y_{t+1} + b^n_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}}}{\sum_{\tau=1}^{t+1} \delta(n,\tau) \frac{1}{\sigma^2_{(n,k)_\tau}}} & \text{si } n \text{ est le plus probable à l'instant } t+1 \\ b^n_t \text{ sinon} \end{cases} \tag{6.1}$$

avec

$$\delta(n,\tau) = \begin{cases} 1 \text{ si } n \text{ est le plus probable à l'instant } \tau \\ 0 \text{ sinon} \end{cases}$$

On voit donc, à partir de l'équation 6.1 qu'une fonction spécifique n'est ré-estimée que lorsque l'état auquel elle est associée est *élu* (considéré comme le plus probable au sens de la probabilité *avant*). Cette dernière remarque soulève un problème : les fonctions associées à des états peu utilisés seront moins bien estimées que celles associées à des états souvent "élus". En effet, il a été vu dans le chapitre précédent que les paramètres des fonctions de compensation oscillent avant de converger vers des valeurs relativement stables. Par conséquent, à un instant t, il est possible que les paramètres de certaines fonctions (associées à des états peu visités) n'aient pas convergé. Ces fonctions ne reflètent donc pas correctement l'influence de l'environnement acoustique sur les observations dans le sous-espace auquel elles sont associées. En d'autres termes, ces fonctions souffrent d'une pénurie d'observations (*data scarcity problem*).

6.1.3 Regroupement des états en classes

La solution envisagée pour contourner ce problème est assez classique. Au lieu d'associer une fonction de compensation à chaque état comme décrit précédemment, il suffit de les associer à des groupes d'états semblables. Ainsi, la fonction associée à un groupe d'états est ré-estimée à chaque fois qu'un état du groupe est élu. Donc plus le groupe contient d'états, plus la fonction de compensation associée sera bien estimée. L'ensemble des groupes d'états forme une partition de l'espace des observations acoustiques. A l'extrême, si l'on regroupe tous les états, on obtient l'algorithme que nous avons décrit dans le chapitre précédent (n'utilisant qu'une fonction de compensation unique).

Cette approche soulève à son tour un problème important. Il concerne l'élaboration des groupes d'états. En effet, ces groupes doivent :

- couvrir des régions acoustiques proches,
- être suffisamment petits pour que la transformation associée reste spécifique à ces régions acoustiques seules,
- être suffisamment grands pour que la transformation associée soit bien estimée.

La première condition sera remplie si l'on utilise une métrique appropriée qui sera discutée dans la section 6.2.1. Afin de remplir les deux dernières conditions, nous avons envisagé une approche classique. Cette solution consiste à créer une collection de partitions de l'espace acoustique. Pour chaque membre de cette collection, la taille des groupes d'états est différente. On dispose donc d'un ensemble de partitions de l'espace des observations acoustiques, chaque partition opérant à un degré de précision différent. Une illustration en deux dimensions de cette collection est représentée en figure 6.1.

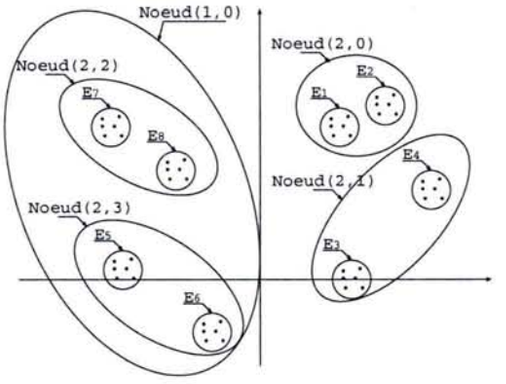


FIG. 6.1 – Etats regroupés dans une structure hiérarchique selon leur couverture phonétique.

La figure 6.1 se lit de la façon suivante : les états $E_1, \ldots, E_8$ recouvrent chacun une région acoustique. Ces états sont ensuite regroupés en classes de similitude (E_1 et E_2 regroupés dans le $Noeud(2,0)$). Ces classes sont enfin inclues dans des classes d'ordre supérieur ($Noeud(2,2)$ et $Noeud(2,3)$ dans le $Noeud(1,0)$).

D'une part, pour les partitions grossières, les groupes contiennent beaucoup d'états et par

conséquent, les fonctions de compensation qu'on y associe sont bien estimées. D'autre part, pour les partitions précises, les groupes contiennent peu d'états et les fonctions de compensations sont bien spécifiques à un sous-espace précis de l'espace des observations acoustiques.

6.1.4 Collection hiérarchique

Les approches structurelles ont souvent été utilisées pour améliorer les techniques d'adaptation (au locuteur, le plus souvent). Dans [Shinoda and Lee, 2001] par exemple, on utilise une méthode MAP structurelle pour améliorer l'estimation de modèles acoustiques dans le cas où les données d'adaptation sont réduites. Le but de cette approche est de lier entre eux les paramètres des modèles acoustiques dans une structure d'arbre. Selon la disponibilité des données d'adaptation, le nombre et la profondeur des nœuds à modifier varient : plus ces données sont nombreuses et plus l'adaptation progressera en profondeur.

Afin de lier les deux avantages d'une partition grossière (bonne estimation) et d'une partition précise (spécificité), on organise cette collection de façon hiérarchique en utilisant la transformation associée à un degré de granularité spécifique selon un critère que nous allons préciser. Ainsi, on dispose d'une hiérarchie de fonctions de compensation et il est possible d'associer à chaque état un ensemble de fonctions de compensation à divers degrés de spécificité.

Nous allons présenter le critère guidant le choix du degré de granularité. Lors de notre algorithme, lorsqu'un état est élu au sens de la probabilité *avant*, il faut lui associer la fonction de compensation qui soit à la fois :

- la plus spécifique à la région acoustique
- bien estimée

Pour cela, on cherche à utiliser la fonction de compensation associée au plus petit groupe d'états contenant l'état élu. Si celle ci n'est pas bien estimée (moins de N observations ont été utilisées pour l'estimer), on utilise la fonction de compensation associée au groupe de taille supérieure contenant l'état élu (groupe parent). Ce raisonnement peut être réitéré si la fonction de compensation obtenue n'est pas bien estimée. Dans ce cas, on utilise la fonction associée au groupe de taille supérieur et ainsi de suite.

En résumé, voici l'algorithme final que nous proposons :

1. initialisation : $t := 0$; et association à chaque état d'un ensemble de paramètres décrivant une fonction de compensation (un simple biais ou les deux paramètres d'une fonction affine) que l'on initialise.
2. à l'indice de temps t,
   (a) pour chaque état $n \in 1, \ldots, N$
      i. choix de la fonction de compensation f^n associée à l'état n à utiliser (la plus précise des bien-estimées).
      ii. calcul de la transformée de l'observation y_t par cette fonction $f^n(y_t)$
   (b) pour chaque état $n \in 1, \ldots, N$ calcul de

$$\alpha_{t|\Theta_{t-1}}(n) := p(f^n(y_t), s_t = n|\Theta_{t-1}),$$

   la probabilité *avant* associée à chaque couple (état ; observation transformée) pour l'alignement de Viterbi.

(c) choix de l'état $n'_t \in 1, \ldots, N$, le plus probable au sens de la probabilité *avant*.

$$n'_t = \arg \max_{n \in 1,\ldots,N} \alpha_{t|\Theta_{t-1}}(n)$$

3. à l'indice de temps t, réestimation des paramètres des fonctions de compensation associées à tout les groupes contenant l'état n'_t comme décrit dans l'équation 6.1.
4. $t := t + 1$;

Dans la section suivante, nous décrirons en détails la conception de l'arbre de transformations. Nous analyserons ensuite la structure obtenue.

6.2 Construction de l'arbre

La construction de cette structure se fait avant la phase de reconnaissance, après l'apprentissage des modèles acoustiques. Comme nous l'avons vu précédemment, la construction repose sur les états des modèles acoustiques. Dans la solution que nous avons retenue, l'arbre (binaire) est édifié selon la technique *bottom-up*. Dans un premier temps, chaque état issu de l'ensemble des modèles acoustiques est associé à une feuille de l'arbre. Chaque feuille est regroupée dans un nœud d'ordre supérieur (groupes de deux états) avec la feuille la plus proche. Les nœuds obtenus sont regroupés ensuite deux par deux pour former un nœud d'ordre supérieur. En répétant l'opération jusqu'à ce que l'on obtienne un unique nœud contenant tout les états, on obtient un arbre binaire. La figure 6.2 représente un exemple d'arbre construit à partir d'un ensemble de 4 états : E_1, E_2, E_3, E_4.

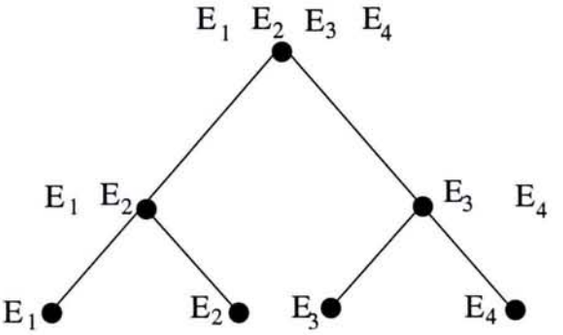


FIG. 6.2 – Exemple d'arbre à 4 états.

Il faut noter à ce point que nous avons opté pour un arbre binaire balancé. Ce choix a été guidé par notre volonté d'avoir un nombre équivalent d'états par nœuds, à chaque niveau de l'arbre. Ainsi, pour chaque nœud d'un même niveau, le nombre d'observations contribuant à l'estimation de la transformation associée est comparable. Ceci garantit que les transformations associées à ces nœuds convergent au même rythme.

6.2.1 Métrique utilisée

Afin de créer cet arbre, nous avons besoin d'établir une distance entre deux nœuds de même ordre dans l'arbre. Dans la solution retenue, la distance entre deux nœuds $N_{(i,k)}$ et $N_{(l,k)}$ (au

même niveau k dans l'arbre binaire) est la moyenne des distances entre les états de $N_{(i,k)}$ et ceux de $N_{(j,k)}$. La distance entre deux états est définie par :

$$D(i,l) = \sum_{k_i=1}^{K_i} \sum_{k_l=1}^{K_l} w_{(i,k_i)} w_{(l,k_l)} KL((i,k_i),(l,k_l)) \tag{6.2}$$

où

- $D(i,l)$ est la distance entre l'état i et l'état j,
- $KL((i,k_i),(l,k_l))$ est la distance de Kullback-Leibler entre la k_i-ème composante gaussienne de l'état i et la k_l-ème composante gaussienne de l'état l.

$$KL((i,k_i),(l,k_l)) = \frac{(\sigma^2_{(i,k_i)} - \sigma^2_{(l,k_l)})^2 + (\mu_{(i,k_i)} - \mu_{(l,k_l)})^2(\sigma^2_{(i,k_i)} + \sigma^2_{(l,k_l)})}{4\sigma^2_{(i,k_i)}\sigma^2_{(l,k_l)}}$$

- $w_{(i,k_i)}$ est le poids de la k_i-ème composante gaussienne dans la distribution de probabilité d'émission associé à l'état i.
- $\sigma^2_{(i,k_i)}$ et $\sigma^2_{(l,k_l)}$ sont les variances de la k_i-ème composante gaussienne de l'état i et la k_l-ème composante gaussienne de l'état l
- $\mu_{(i,k_i)}$ et $\mu_{(l,k_l)}$ sont les moyennes de ces composantes.

6.2.2 Description de l'arbre obtenu

La figure 6.3 reproduit la répartition des états de modèles acoustiques (MA) dans les nœuds d'un arbre de profondeur 6 construit comme décrit précédemment. Les modèles acoustiques utilisés dans cet exemples sont ceux utilisés par le SRAP lors des expériences sur la bases VODIS. Leur description peut être vue dans le tableau 4.1 à la section 4.4.1. Dans ce cas précis, les modèles acoustiques sont des modèles *monophones* à 3 états utilisés pour la reconnaissance sur la base VODIS. Le tableau 4.1 du chapitre 4 de la première partie contient une description des modèles acoustiques utilisés et référencés dans l'arbre de la figure 6.3 .

Les feuilles contiennent chacune au plus 8 états. Les états sont représentés sous la forme :

'nom du MA'$_{\text{indice dans le MA}}$

Deux nœuds intermédiaires ont été représentés (à la profondeur 2). Ils regroupent tout les états des *MA* (**y, u, n, R, S, l, Z, 9~**) pour l'un et (**silence, !, E/, 2, O/, a**) pour l'autre.

On peut noter que la forme générale contient des informations d'ordre phonétique bien que l'arbre n'ait été construit que sur des considérations de distances entre états. Par exemple, l'ensemble des états des *MA* des consonnes liquides (**R** et **l**) sont regroupés dans le même nœud de niveau 3. De même, tout les états appartenant aux voyelles **u** d'une part et **a** d'autre part sont regroupés dans des nœuds de profondeur 4.

Compte tenu de son mode de construction, il n'est pas possible de garantir que la structure hiérarchique obtenue corresponde à une quelconque classification phonétique. Toutefois, au vu de la répartition des états obtenue, on considèrera par la suite que chaque nœud de l'arbre couvre bien une région phonétique particulière de l'espace des observations, comme évoqué plus haut. De plus, la construction de l'arbre implique que les états des modèles acoustiques soient regroupés dans ses nœuds selon leur similitude. Ceci justifie que l'on associe aux nœuds des transformations dépendantes des états.

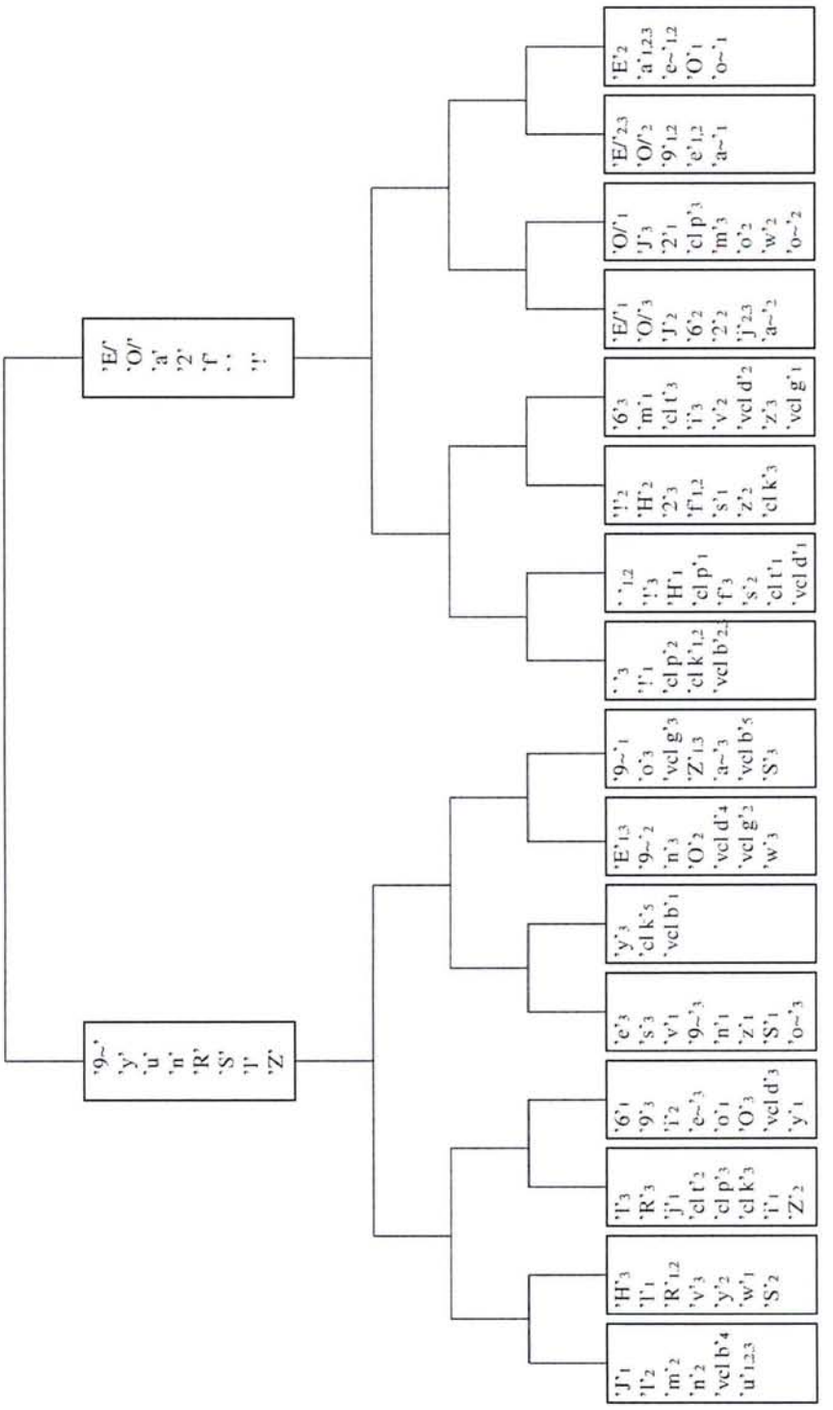


FIG. 6.3 – Arbre des états obtenu à partir des modèles phonétiques pour VODIS.

6.3 Résultats

6.3.1 Premiers Résultats

Des expériences ont été menées sur la partie *far-talk* de la base de test de VODIS.

Le tableau 6.1 donne les premiers résultats concernant l'approche structurelle que nous proposons. Il présente les taux de reconnaissance en mots (les grammaires utilisées sont en annexe) pour les tâches *nombres100à15000* et *téléphone* selon la profondeur de l'arbre utilisée. A noter que l'arbre de profondeur 1 ne comporte qu'un seul nœud : le nœud racine. Pour toutes les expériences proposées, la fonction spécifique à un noeud de l'arbre est considérée comme utilisable dans le processus de compensation si plus de 10 observations ont contribué à son estimation ($N = 10$ dans l'algorithme donné en section 6.1.4). Cette valeur a été fixée d'après les résultats obtenus par la version non-structurelle de notre algorithme présenté au chapitre 5, section 5.3.3.

Profondeur de l'arbre	1	2	3	4
nombre de noeuds	1	3	7	15
nombres100à15000	72.8	73.7	72.3	71.0
téléphone	83.4	85.2	84.3	83.5

TAB. 6.1 – Taux de reconnaissance en mots (%) obtenus par *Biais* hiérarchique pour *nombres100à15000* et *téléphone far-talk* de VODIS.

Cette première expérience montre que l'approche structurelle augmente le taux de reconnaissance par rapport à la version originale de notre algorithme (les scores de cette version sont ceux de notre arbre lorsque la profondeur est de 1). Ceci est particulièrement évident pour la tâche *téléphone* où l'on obtient, pour un arbre de profondeur 2, une amélioration relative du taux de reconnaissance de 2.2%. L'amélioration est donc significative dans ce cas, alors que le taux de reconnaissance est déjà élevé.

On peut observer dans ce tableau que les meilleures performances sont obtenues pour un arbre de faible profondeur constitué d'un noeud père et de deux feuilles. Les taux de reconnaissances décroissent à mesure que la profondeur augmente.

Plusieurs hypothèses non-exclusives ont été envisagées pour expliquer ce comportement :

- pour les nœuds les plus bas contenant peu d'états, les paramètres des fonctions de compensation sont mal estimés. En effet, le nombre d'observations contribuant à leur estimation est faible
- l'initialisation des paramètres des fonctions de compensation associées à chaque nœud n'est pas assez judicieuse. En effet, tout ces paramètres sont initialisés de sorte que la fonction de compensation à $t = 0$ soit $f(y_t) = y_t$ (fonction identité).
- la fonction est discontinue. En effet, la fonction de compensation est la succession dans le temps des transformations de compensation associées aux états les plus probables successifs. Donc, à chaque transition d'état, la fonction de compensation subit une rupture en passant d'une transformation à une autre[30].

Dans la prochaine section, nous effectuerons une expérience qui permettra de valider l'approche hiérarchique que nous présentons. Puis dans les sections suivantes nous mettrons en évidence le caractère discontinu de la fonction de compensation évoquée précédement. Enfin nous

[30] voir la section 6.3.3 pour une discussion sur les effets de cette discontinuité

présenterons la solution que nous avons retenue pour atténuer les effets de discontinuité de la fonction de compensation.

6.3.2 Validation de l'approche hiérarchique

Comme il a été envisagé dans le chapitre 5, une bonne initialisation des paramètres des fonctions de compensation doit pouvoir remédier à une convergence lente. Il a été montré dans ce même chapitre que, dans le cas de la version non-hiérarchique de notre algorithme, une telle initialisation n'apportait pas une amélioration significative. Cependant, dans le cas d'une version hiérarchique, les nœuds situés profondément dans l'arbre devraient tirer bénéfice d'une initialisation et palier à un manque d'observations.

Pour vérifier cette proposition, nous avons élaboré une expérience simple. Si elle ne peut pas être directement reprise dans une application concrète, cette expérience permet de mettre en évidence que l'organisation hiérarchique de ces transformations doit apporter un réel bénéfice à notre algorithme.

Voici le mode opératoire :

1. une première passe de reconnaissance sur les phrases de test *close-talk* donne un chemin optimal correct pour toutes les phrases de test.
2. pour chaque séquence d'états, on calcule l'ensemble des paramètres des transformations de l'arbre associé.
3. Une deuxième passe de reconnaissance est effectuée sur chaque phrase de test *far-talk*. A chaque phrase, les valeurs des transformations de l'arbre sont fixées aux valeurs calculées à l'étape 2.

L'évaluation de cette expérience porte sur le taux de reconnaissance obtenu par notre algorithme hiérarchique lorsque l'arbre des transformations est fixé avec des valeurs calculées de façon optimale dans un environnement très semblable (même locuteur, même séquence de phonèmes, même environnement acoustique). Les résultats de cette expérience donneront la limite supérieure des performances que l'on peut attendre de notre approche hiérarchique.

Le tableau 6.2 reproduit les résultats obtenus.

profondeur de l'arbre	1	2	3	4	5	6	7	8
Amélioration de l'*Acc.* (%)	0	+1.4	+17.1	+23.3	+28.1	+32.9	+33.1	+36.5

TAB. 6.2 – Amélioration du taux de reconnaissance relativement au taux obtenu pour la profondeur 1, pour une reconnaissance en phonèmes par *Biais* structurelle, les biais étant fixés à des valeurs obtenues par alignement forcé (toute la base de test *far-talk* de VODIS).

Le tableau 6.2 rend compte que l'utilisation d'une hiérarchie de transformations profonde améliore le taux de reconnaissance en comparaison avec l'utilisation d'une seule transformation (arbre de profondeur 1). Cette amélioration est d'autant plus significative que le nombre de transformations utilisées est important.

Ces observations justifient donc l'utilité de l'approche hiérarchique et la validité de la structure utilisée.

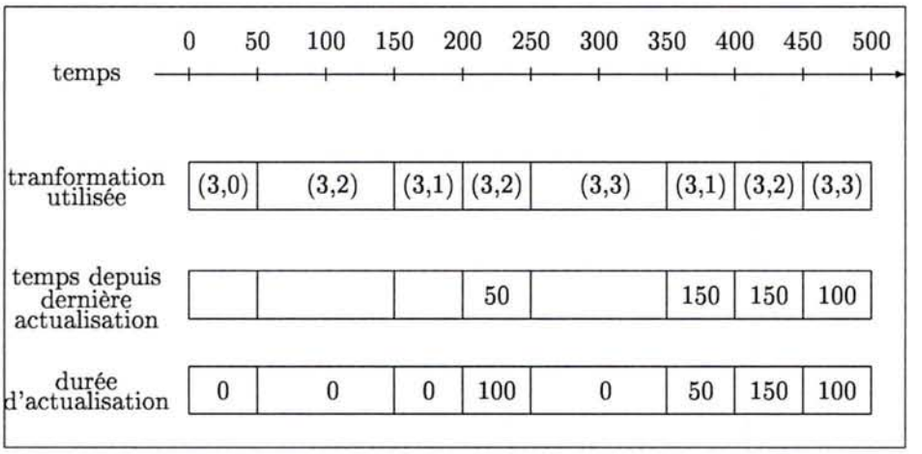


FIG. 6.4 – Succession des transformations constituant la fonction de compensation.

6.3.3 Discontinuité de la fonction de compensation

Cette section permet de mettre en évidence le caractère discontinu de la fonction de compensation. Cette discontinuité est inhérente à la forme de l'algorithme présenté en section 6.1.4. En effet, la fonction de compensation est une suite de transformations. Les transitions d'une transformation à une autre interviennent à chaque fois que l'état le plus probable au sens de la *probabilité avant* change. Les transformations n'ayant pas forcément les mêmes paramètres, on observe donc des sauts dans les valeurs de paramètres de la fonction de compensation.

La figure 6.4 donne un exemple de succession des transformations utilisées comme fonction de compensation entre les instants $t = 0$ et $t = 500$.

Sur cet intervalle, on utilise successivement les transformations associées aux nœuds (3,0), (3,2), (3,1), (3,2), (3,3), (3,1), (3,2) puis (3,3). La figure met en évidence que la transformation de compensation d'une feuille n'est réestimée qu'aux instants où un vecteur issu de la région acoustique associée a été observé. Par exemple, à l'instant 200, la transformation associée au nœud (3,2) n'a pas été réestimée depuis 50 unités de temps et la période sur laquelle elle a été estimée a duré 100 unités de temps.

Plusieurs questions peuvent alors être posées :

- L'estimation de la transformation est elle encore valide après ce laps de temps ? En d'autres termes : l'environnement acoustique a-t-il évolué pendant cette période au point que cette estimation soit devenu caduque ?
- Que se passe-t-il à la transition entre deux transformations, par exemple à $t = 400$?

La figure 6.5 représente les valeurs du biais utilisées comme fonction de compensation, sur une portion de phrase de test. Cette figure met en évidence le fait que la fonction de compensation est une succession de transformations et est, par construction, discontinue. Les lignes verticales indiquent les moments auxquels il y a un changement de transformation. Par exemple, on peut voir que la transformation liée au nœud (8,4) est utilisée entre les instants $t = 3$ et $t = 8$ puis à partir de l'instant $t = 57$. Les transitions entre deux fonctions peuvent être brusques, comme à $t = 44$ ou plus continues, comme à $t = 8$. On observe des continuités lorsqu'on passe d'un nœud

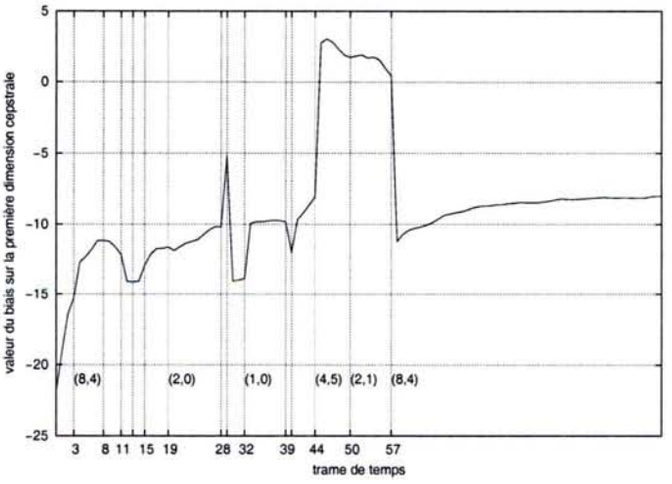


FIG. 6.5 – Valeur du biais (première dimension cepstrale) utilisée dans la version hiérarchique de *Biais* pour une portion de phrase de test *far-talk* de VODIS.

à un autre contenant des états proches, comme à la transition entre (4,5) et (2,1), à $t = 50$. Inversement, lorsque deux nœuds contiennent des états très distincts (comme (8,4) et (2,1)), on observe une transition d'autant plus brusque que les transformations qui leur sont associées sont bien estimées (comme par exemple en fin de phrase, ici à $t = 57$).

6.3.4 Processus de lissage de la fonction de compensation (*RootSmooth*)

La fonction de compensation est une succession de transformations spécifiques à une région acoustique. Afin de réduire les discontinuités dans cette fonction, nous avons mis en place une procédure de lissage simple. Il découle de la construction de l'arbre que la transformation associée à sa racine soit estimée à chaque instant puisque la racine contient tous les états. La racine contient donc une transformation qui n'est pas spécifique à une région acoustique particulière mais qui est estimée à chaque instant.

La solution que nous avons adoptée pour réduire la discontinuité de la fonction de compensation est d'y introduire une composante continue. Dans le cas où la transformation utilisée est un biais (*Biais*), on utilisera à l'instant t, la moyenne arithmétique $\hat{b}_{(i,j,t)}$ entre $b_{(i,j,t)}$, le biais associé au nœud (i,j) et $b_{(0,0,t)}$, le biais associé au nœud racine (0,0), soit

$$\hat{b}_{(i,j,t)} = \frac{b_{(i,j,t)} + b_{(0,0,t)}}{2}$$

plutôt que le biais $b_{(i,j,t)}$ seul. Ainsi, à chaque instant, le biais utilisé dans la fonction de compensation contient une composante continue et une composante discontinue. Dans la suite du document, nous ferons référence à cette technique sous le nom de *RootSmooth*.

Voici l'algorithme proposé, en utilisant un biais comme fonction de compensation :

1. initialisation : $t := 0$; association à chaque état d'un ensemble de paramètres décrivant une fonction de compensation (un simple biais dans ce cas) que l'on initialise.
2. à l'indice de temps t,

(a) pour chaque état $n \in 1, \ldots, N$

i. choix du nœud (i_n, j_n) contenant l'état n et possédant la plus précise et la mieux estimée des transformations de l'arbre.

ii. calcul de la transformé $\overline{y}_t$ de l'observation y_t par cette fonction

$$\overline{y}_t = y_t + \hat{b}_{(i_n, j_n, t)}$$

(b) pour chaque état $n \in 1, \ldots, N$ calcul de

$$\alpha_{t|B_{t-1}}(n) := p(\overline{y}_t, s_t = n|B_{t-1}),$$

la probabilité *avant* associée à chaque couple (état ; observation transformée) pour l'alignement de Viterbi.

(c) Choix de l'état $n'_t \in 1, \ldots, N$, le plus probable au sens de la probabilité *avant*.

$$n'_t = \arg \max_{n \in 1, \ldots, N} \alpha_{t|B_{t-1}}(n)$$

3. à l'indice de temps t, réestimation des paramètres des fonctions de compensation associées à tous les groupes contenant l'état n'_t comme décrit dans l'équation 6.1.
4. $t := t + 1$;

La figure 6.6 compare les taux de reconnaissance en phonèmes avec (*RootSmooth*) et sans processus de lissage (*Sans RootSmooth*).

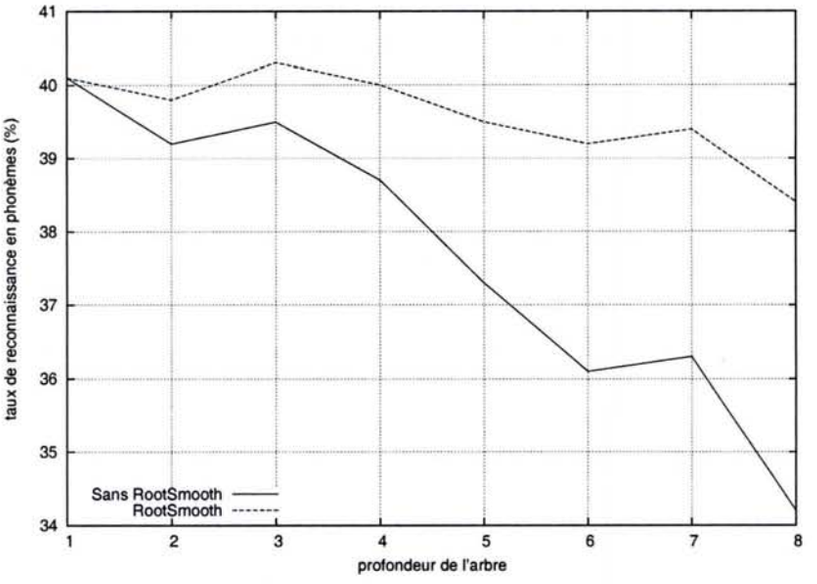


FIG. 6.6 – Taux de reconnaissance en phonèmes (%) obtenus par *Biais* hiérarchique en utilisant un processus de lissage *RootSmooth* pour toute la base de test *far-talk* de VODIS.

On peut observer qu'en l'absence de lissage, les performances du système se dégradent à mesure que la profondeur croît. Cependant, lorsque la technique de lissage *RootSmooth* est utilisée,

cette perte de performance est réduite voir annulée (pour la profondeur 3). Pour la profondeur 3, le taux de reconnaissance en phonèmes est supérieur au maximum obtenu par la version n'intégrant pas de lissage.

Cette expérience, ainsi que les résultats obtenus dans les sections suivantes nous ont conforté dans l'idée que ce lissage permettait une amélioration des performances de notre approche.

6.4 Résumé et expériences sur des tâches spécifiques

6.4.1 Initialisation

Le but de cette expérience est de mettre en évidence l'importance de l'initialisation des valeurs de biais dans le cadre du *Biais* hiérarchique. Pour cela, il a fallu segmenter la base de test VODIS selon le *RSB* des phrases de test de la façon décrite dans le tableau 6.3. En effet, pour la première expérience, les paramètres des transformations de compensation devaient être initialisés avec des valeurs obtenues après la reconnaissance d'une phrase prononcée dans un environnement similaire. Pour cette expérience, nous avons choisi une mesure de similarité basique : le niveau de bruit. Ainsi, les ensembles de test ont été partitionnés en classes de Rapport Signal à Bruit (RSB). La reconnaissance s'est effectuée sur chaque ensemble, en phases successives. Durant chaque phase, les paramètres des transformations de compensation étaient initialisés avec les paramètres obtenus à la fin de la reconnaissance de la phrase précédente. Ainsi, au début de chaque phrase, les paramètres des fonctions de compensation étaient initialisés avec des paramètres obtenus par compensation dans un milieu similaire.

	Cluster_1	Cluster_2	Cluster_3	Cluster_4	Cluster_5	Cluster_6
Nombre de phrases	1118	828	1096	1057	916	1102
RSB	< 3	$3 < 5$	$5 < 7$	$7 < 9$	$9 < 12$	$12 <$

TAB. 6.3 – Segmentation de la base de test VODIS *far talk* en classes de RSB.

D'autre part les chemins optimums obtenus sur les versions *close-talk* des phrases de test ont servi à calculer des arbres de biais d'une manière similaire à celle décrite dans la section 6.3.2,. Ces arbres de biais ont été utilisés dans des test d'initialisations décrit ci-dessous.

La figure 6.7 montre les taux de reconnaissance en phonèmes sur toute la base de test *far-talk* de VODIS, lorsque plusieurs processus d'initialisation sont utilisés avec le *Biais* hiérarchique :

- Les biais sont initialisés à 0, la fonction de compensation est donc initialisée à l'identité (*Sans Init.*).
- Les biais sont initialisés aux valeurs obtenues à la fin de la phrase précédente, les phrases étant regroupées par RSB croissant (*Init. par Cluster*).
- Les biais sont initialisés aux valeurs moyennes sur les ensembles RSB des biais obtenus par alignement avec les séquences optimales des phrases *close-talk* (*Init. par Moyenne*).
- Les biais sont initialisés aux biais obtenus par alignement avec les séquences optimales des phrases *close-talk* (*Init. par Phrase*).

Pour l'expérience *Init. par Phrase*, le taux de reconnaissance est globalement croissant (en fonction de la profondeur). Ce résultat est à rapprocher de celui donné en section 6.3.2 et confirme que la structure hiérarchique des transformations est valable.

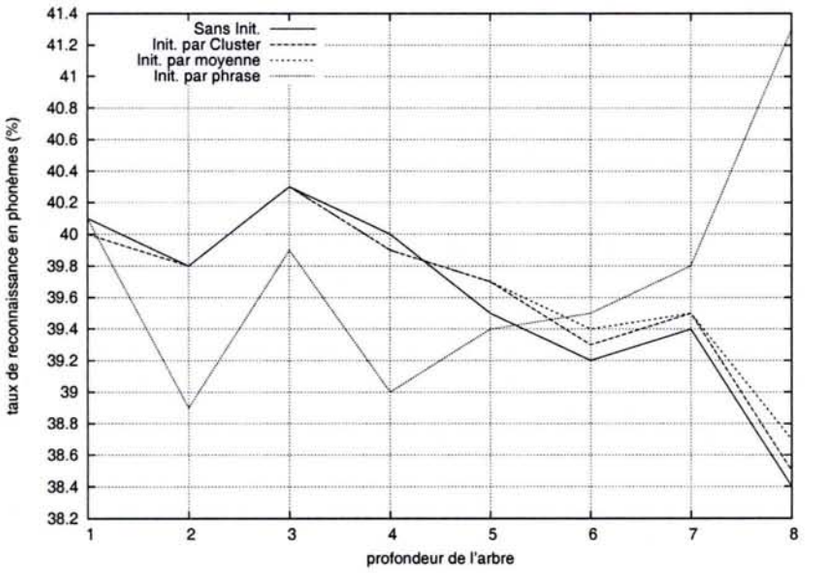


FIG. 6.7 – Taux de reconnaissance en phonèmes (%) obtenus par *Biais* hiérarchique sur toute la base de test *far-talk* de VODIS, en utilisant un processus de lissage *RootSmooth* et différentes initialisation.

Cependant, le fait que pour les autres expériences les taux de reconnaissance soient globalement décroissants montre que cette structure hiérarchique ne peut être correctement exploitée qu'à condition que la phase d'initialisation soit correctement conduite. En effet, pour les expériences *Init. par Cluster* et *Init. par Moyenne*, le maximum du taux de reconnaissance est obtenu pour une profondeur de 3 et décroit à mesure que la profondeur augmente. Pour ces expériences, l'initialisation des paramètres des fonctions apporte peu par rapport à *Sans Init.*. La contribution de ces initialisations est visible pour les profondeurs supérieures à 4.

Cette série d'expériences permet de conclure que l'approche hiérarchique telle que nous l'avons conçue est fondée mais qu'elle semble nécessiter une phase d'initialisation précise pour être réellement efficace. Cette phase d'initialisation dépend du type d'application dans lequel ce système serait implanté.

6.4.2 Application à la reconnaissance

Dans cette section, nous appliquons les solutions obtenues précédemment à la reconnaissance pour les tâches *nombres100à15000* et *téléphone* de VODIS.

La figure 6.8 représente les taux de reconnaissance en mots obtenus pour la tâche *nombres100à15000 far-talk* pour la version de *Biais* hiérarchique selon que l'on utilise un lissage ou pas. La figure 6.9 représente les mêmes valeurs pour la tâche *téléphone*.

Dans cette expérience, les transformations (*Biais*) sont initialisées à la fonction identité au début de chaque phrase. Cette expérience confirme les conclusions tirées en section 6.3.4 quant à l'influence du lissage sur les taux de reconnaissance obtenus par les profondeurs supérieurs.

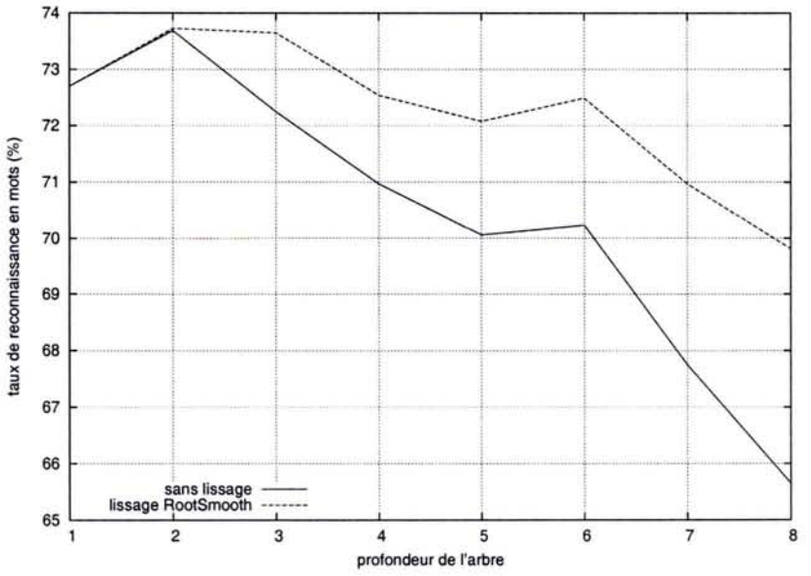


FIG. 6.8 – Taux de reconnaissance en mots (%) obtenus par *Biais* hiérarchique avec et sans lissage *RootSmooth* pour *nombres100à15000 far-talk* de VODIS.

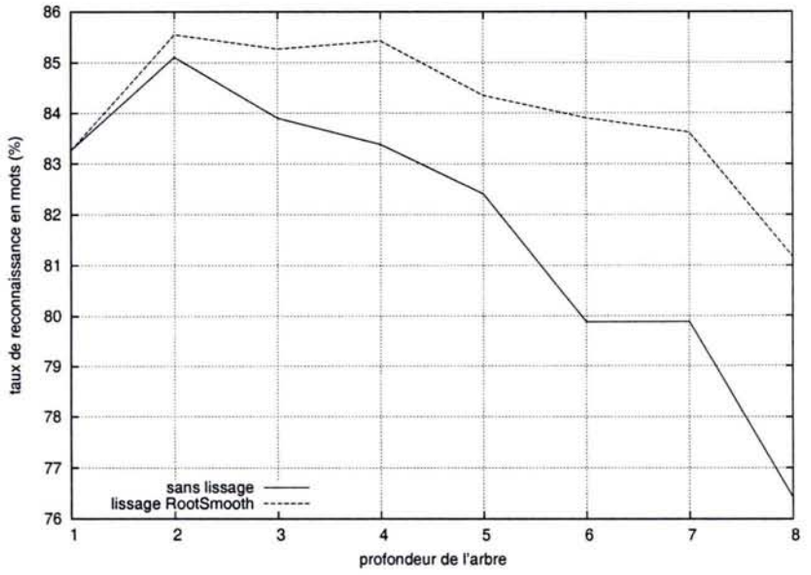


FIG. 6.9 – Taux de reconnaissance en mots (%) obtenus par *Biais* hiérarchique avec et sans lissage *RootSmooth* pour *téléphone far-talk* de VODIS.

L'expérience suivante permet d'explorer les possibilités d'initialisation des transformations, au début de chaque phrase. Ici encore nous avons segmenté la base de test *nombres100à15000 far-talk* en 4 classes de RSB :

$$\text{RSB} < 3 \text{ dB},\ 3 \text{ dB} \leq \text{RSB} < 9 \text{ dB},\ 9 \text{ dB} \leq \text{RSB} < 14 \text{ dB et RSB} \geq 14 \text{ dB}$$

Cette segmentation nous a permis de mettre en place un système d'initialisation similaire à *Init. par Cluster* décrit à la section précédente. La figure 6.10 représente les taux de reconnaissance en mots pour cette tâche

- sans initialisation (*Sans Init*),
- avec initialisation sur la valeur des paramètres obtenus à la fin de la reconnaissance de la phrase précédente dans la classe de RSB (*Init. par Cluster*) et
- avec initialisation sur la valeur des paramètres obtenus à partir de l'alignement sur la version *close-talk* de la phrase à reconnaître (*Init. par Phrase*).

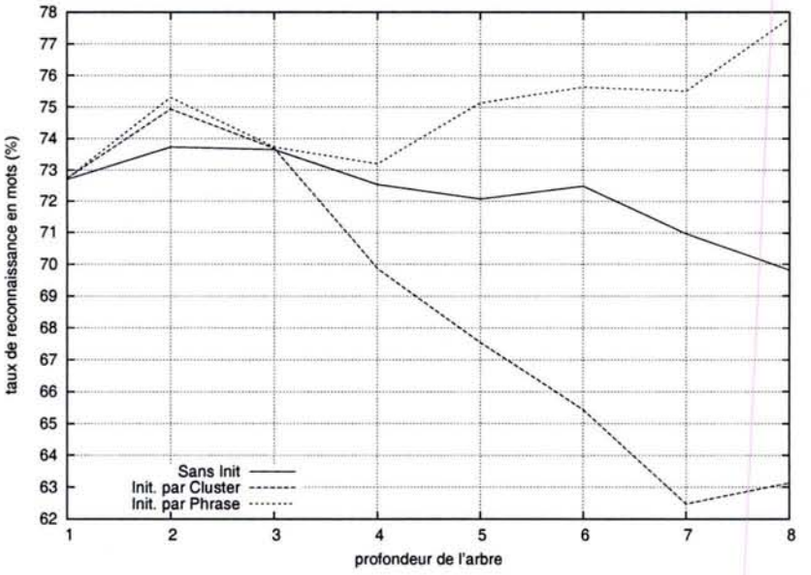


FIG. 6.10 – Taux de reconnaissance en mots obtenus par *Biais* hiérarchique avec lissage *RootSmooth* et différentes initialisation pour *nombres100à15000 far-talk* de VODIS.

Pour l'exercice *nombres100à15000*, on observe que l'initialisation *Init. par Cluster* apporte un gain dans la reconnaissance comparable à celui apporté par une initialisation de type *Init. par Phrase* pour les profondeurs 1, 2 et 3. La combinaison du processus de lissage *RootSmooth* et de l'initialisation *Init. par Cluster* produit une amélioration relative du taux de reconnaissance en mots de 3% par rapport à la version non-hiérarchique de *Biais* (pour une profondeur de 2, c'est à dire : un arbre d'une racine et 2 feuilles). Cependant, cette initialisation n'est pas efficace pour les profondeurs d'ordre supérieur et dégrade les performances par rapport à *Sans Init.*

Enfin, l'expérience *Init. par Phrase* confirme que l'organisation hiérarchique des transformations dépendantes des états est fondée mais que la phase d'initialisation des paramètres de ces fonctions est indispensable pour que l'apport de cette approche soit significatif.

6.5 Conclusion

Dans ce chapitre, nous avons présenté un algorithme de compensation en temps réel utilisant une structure hiérarchique de transformations de compensation. Cet algorithme a pour base celui développé dans le chapitre 5 et est utilisé pour la reconnaissance de mots dans un environnement variant lentement. Son objectif est de tenir compte des distorsions non linéaires subies par le signal de parole dans un environnement bruité.

La structure hiérarchique contenant les transformation utilisées pour la compensation est construit à partir des états des modèles acoustiques utilisés lors de la reconnaissance. Il existe donc, pour chaque état des modèles acoustiques, une correspondance avec une transformation. Lors de la reconnaissance et à chaque instant, la transformation spécifique à l'état le plus probable est utilisée pour compenser l'observation bruitée.

Cette approche affiche de très bonnes performances pour la reconnaissance de nombres et de numéros de téléphones prononcés dans l'habitacle d'une voiture. Cependant elle est limitée par un problème de pénurie de données qui empêche d'utiliser une structure trop importante. Cette difficulté peut être contournée en posant des hypothèses permettant d'initialiser les transformations présentes dans la structure.

7

Reconnaissance dans un milieu non-stationnaire

Sommaire

Une application utilisant un SRAP en temps réel peut être amenée à fonctionner dans un environnement acoustique non-stationnaire. Dans ce cas, le signal sonore à reconnaître peut être pollué par des bruits inopinés et soudains. Dans ce chapitre, nous exposons les solutions que nous avons envisagées pour permettre à l'algorithme de compensation proposé au chapitre 5 d'être robuste en présence de ces évènements [Barreaud *et al.*, 2003b].

Nous allons montrer qu'un changement dans l'environnement acoustique induit une altération dans la distribution d'une variable dérivée du processus de compensation. Nous verrons alors qu'il est possible de détecter ces altérations de manière simple ; la détection déclenchant une réinitialisation du processus de compensation améliore la reconnaissance en mots de plus de 30% pour la reconnaissance de nombres de la base VODIS, bruitée par un source additive intervenant à mi-phrase.

Trois méthodes de détection, empruntées à divers domaines de recherche, ont été étudiées sur les bases Aurora3 et VODIS bruitées artificiellement.

7.1 Motivation de l'approche

Au cours d'une utilisation réelle, un SRAP peut être confronté à des apparitions soudaines d'un bruit dans l'environnement acoustique. Considérons par exemple le cas d'un opérateur utilisant un SRAP dans une usine ou simplement un bureau ou cohabitent plusieurs collègues. Le signal de parole de l'opérateur peut être corrompu par des bruits inopinés, comme par exemple la mise en marche d'une machine, une sonnerie de téléphone. Ce signal additif peut aussi être court, comme la chute d'un outil, le claquement d'une porte...

Dans ce cas, aucune information n'est disponible sur le moment d'apparition, le niveau ou la nature de ce bruit. Par conséquent, un algorithme de compensation devrait réagir à cet événement et réactualiser sa politique de compensation très rapidement. Dans cette optique, deux problèmes sont à résoudre. Le premier concerne la détection du changement d'environnement et le deuxième la stratégie à adopter pour s'adapter à la nouvelle source de bruit.

La détection de changement dans un signal est un sujet fréquemment abordé dans de nombreux domaines. Mais dans la plupart des cas, lorsqu'une détection temps réel est envisagée, un modèle de la perturbation qui apparaît est nécessaire, ce qui implique une hypothèse *a priori* dont cherche à s'affranchir notre solution. Dans [Barreaud *et al.*, 2003b] a été présentée une amélioration de notre algorithme de base prenant en charge les variations brusques de l'environnement. A chaque trame, en parallèle avec l'algorithme de Viterbi, la distance entre une observation et l'état le plus probable est calculée. La distribution de cette distance (appelé δ par la suite) est gaussienne dans un environnement stable. Une brusque variation de ce dernier peut donc être repérée en surveillant toute rupture dans le comportement de cette variable. Une fois le changement d'environnement détecté, le processus de compensation est réinitialisé afin de ne plus prendre en compte les caractéristiques passées.

Cette section s'organise de la façon suivante. Dans un premier temps, nous présenterons la base de test utilisée pour simuler la reconnaissance dans un milieu acoustique changeant brusquement. Dans un deuxième temps, nous mettrons en évidence que l'algorithme de compensation présenté au chapitre 5 obtient de très bon résultats dans un milieu variant inopinément s'il est réinitialisé aux moments où l'environnement acoustique change. Puis, nous présenterons les caractéristiques de la variable δ, et montrerons pourquoi sa distribution est un indice de la variation de l'environnement acoustique. Enfin, nous exposerons comment l'information relative au changement d'environnement peut s'intégrer dans le processus de compensation que nous proposons.

7.1.1 Base de test artificiellement bruitée

Dans le chapitre 5, nous avons proposé un algorithme de compensation *temps-réel* permettant d'augmenter le taux de reconnaissance de SRAP opérant dans un milieu bruyant variant lentement (typiquement : l'habitacle d'une voiture). Ce cadre applicatif est repris par la plupart des méthodes de compensation et d'adaptation. Cependant, un SRAP utilisé dans une application destinée au grand public n'opérera pas toujours dans ce milieu particulier.

Afin de tester notre approche dans ce type d'environnements particulièrement difficile, nous avons bruité artificiellement les deux bases de test VODIS et Aurora3 [31].

Les parties propres (c'est-à-dire *close-talk*) des bases de test ont été bruitées artificiellement par addition d'un bruit d'avion (bucanneer2.wav de NOISEX) à différents rapports signal à bruit. Cette opération nous a permis de créer 2 nouvelles épreuves de test :

- l'épreuve *échelon* : le bruit est ajouté à partir du milieu de chaque phrase de test. L'objectif de cette épreuve est de mettre en évidence la capacité d'un algorithme de compensation à détecter le changement dans l'environnement acoustique. Cette base simule par exemple un ensemble de phrases débutées dans un environnement calme et au milieu desquelles une machine bruyante est lancée.
- l'épreuve *aléatoire* : le bruit est ajouté pendant une durée de 300ms, aléatoirement à chaque phrase de test, deux apparitions du bruit ne pouvant être séparées de moins de 300ms. Cette épreuve vient compléter la première et permettra d'évaluer la capacité d'un processus de compensation à s'adapter rapidement à un nouvel environnement. Cette base simule par exemple un ensemble de phrases où interviennent des claquements de porte.

La figure 7.1 montre trois spectrogrammes. Le premier (a) est celui d'une phrase issue de la partie *close-talk* de la base de test de VODIS. Le deuxième (b) est celui de la même phrase corrompue par *échelon* (RSB moyen sur la partie bruitée : 8dB). Le troisième (c) est celui de la même phrase corrompue par *aléatoire* (RSB moyen sur les parties bruitées : 8dB).

On peut remarquer que les bases de test obtenues n'ont été corrompues que par un bruit additif dans le domaine temporel. Nous n'avons pas jugé bon de simuler une variation rapide d'un bruit de convolution car nous avons considéré que ce type de bruit était moins sujet à des variations rapides que le bruit additif.

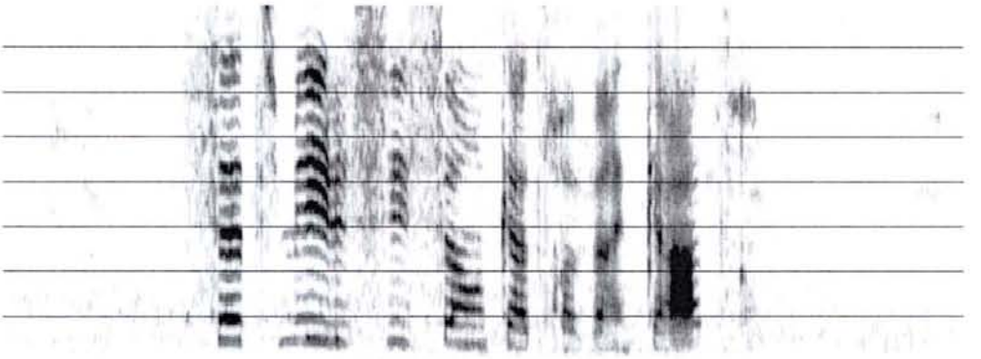
Les tableaux 7.1 et 7.2 représentent les taux de reconnaissance en mots obtenus par le SRAP ESPERE (sans aucun processus de robustesse) sur les bases de test développées.

RSB (dB)	14.8	8.8	5.5	2.8	-0.8	-5.2	-9.3
échelon	78.0	61.2	54.1	52.0	48.7	48.1	46.4
aléatoire	85.4	72.7	62.6	59.9	54.4	45.6	38.1

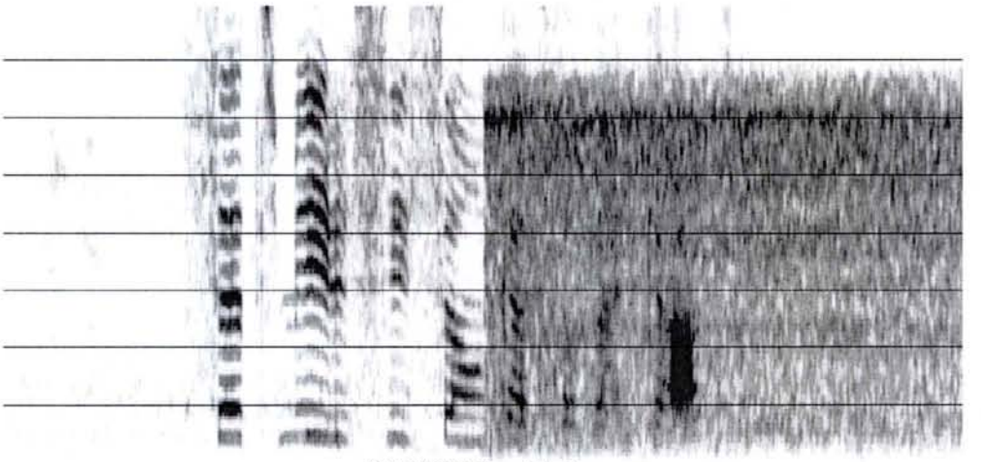
TAB. 7.1 – Taux de reconnaissance en mots pour *Référence* sur VODIS *échelon* et *aléatoire* (RSB calculé sur parties bruitées).

La section suivante montre comment un SRAP intégrant un processus de compensation peut présenter une dégradation de ses performances dans un milieu variant inopinément, même s'il est efficace dans un milieu variant lentement. Nous verrons aussi comment il est possible, dans le cas où notre algorithme de compensation est utilisé, d'annuler cette dégradation par un mécanisme de réinitialisation très simple.

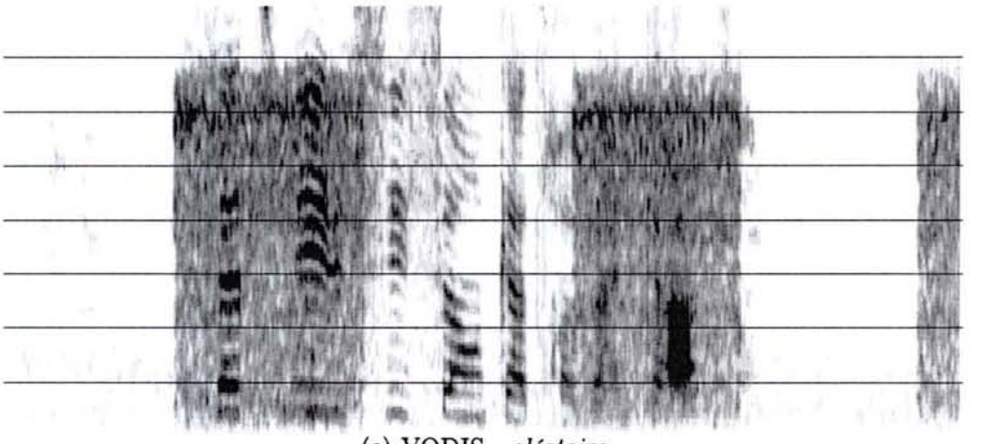
[31] Les descriptifs de ces bases de test sont au chapitre 4, section 4.2



(a) VODIS - *close-talk*



(b) VODIS - *échelon*



(c) VODIS - *aléatoire*

FIG. 7.1 – Exemples de spectrogrammes de phrases de test VODIS artificiellement corrompues par *échelon* et *aléatoire*.

RSB (dB)	14.2	10.7	8.2	6.3	4.7	0.3	-3.8
échelon	96.6	82.9	76.2	71.7	69.0	56.7	46.0
aléatoire	89.2	75.3	67.7	64.9	60.7	55.8	48.7

TAB. 7.2 – Taux de reconnaissance en mots pour *Référence* sur Aurora *échelon* et *aléatoire* (RSB calculé sur parties bruitées).

7.1.2 Justification de la réinitialisation du processus de compensation

Le tableau 7.3 donne un premier résultat intéressant, qui oriente notre progression vers un algorithme de compensation utilisable dans un milieu variant inopinément. Pour cette expérience, nous avons effectué une reconnaissance sur la base de test *nombres100a15000* de VODIS, corrompue par un bruit additif aléatoire (épreuve : *aléatoire*) avec un très faible RSB. Nous avons comparé les taux de reconnaissance obtenus par

- un système n'intégrant pas de compensation (*Référence*),
- un système intégrant la version non-hiérarchique de notre algorithme (ff=1.0) (*Biais*),
- un système intégrant une dernière version de *Biais* qui effectue une remise à zéro (*Raz*) des paramètres de la fonction de compensation aux instants de variation de l'environnement acoustique (ces instants étant fournit au SRAP par l'intermédiaire d'un fichier constitué lors du bruitage de la base de test).

	Référence	*Biais*	*Biais et Raz aux dates données*
Acc. (%)	38.9%	40.1%	42.8%

TAB. 7.3 – Taux de reconnaissance en mots sur la base de test *nombres100à15000* de VODIS, bruitées par un son additif aléatoire (épreuve : *aléatoire*) sans compensation (*Référence*), avec (*Biais*) et avec compensation et remise à zéro des paramètres de compensation aux moments de changements d'environnement acoustique (*Biais et Raz aux dates données*).

On peut voir que le fait de réinitialiser le processus de compensation à chaque changement d'environnement acoustique améliore le taux de reconnaissance par rapport à la première version de *Biais*. Ceci s'explique par le fait que notre processus de compensation repose sur un calcul itératif des paramètres de la fonction de compensation. C'est-à-dire qu'à chaque instant, le calcul des paramètres se fait à partir d'un historique relatif à l'environnement acoustique. Donc, pour l'expérience *Biais* du tableau 7.3, les paramètres de la fonction de compensation aux instants où s'effectue un changement d'environnement acoustique sont calculés à partir d'un historique se rapportant à un environnement qui n'existe plus. On peut donc dire que, lors du passage d'un environnement acoustique à un autre (inconnu) cet historique devient nuisible pour le calcul des paramètres de la fonction de compensation. Par conséquent, réinitialiser ces paramètres à ces instants de passages doit permettre au processus de compensation d'élaborer une transformation qui corresponde mieux au nouvel environnement.

Dans la section suivante, nous allons montrer comment il est possible de repérer les changements dans l'environnement acoustique afin de piloter la réinitialisation du processus de compensation.

7.1.3 Variable rendant compte de l'environnement acoustique : δ

Dans le chapitre 5, nous avons présenté comment il était possible de calculer de façon itérative les paramètres d'une fonction de compensation simple. Dans le cas où cette fonction de compensation est un biais, le calcul du seul paramètre b_t s'opère de la façon suivante (voir équation 5.4, section 5.1.1 du chapitre 5) :

$$b_{t+1} = b_t - \frac{\delta_{t+1}}{\sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}}} \tag{7.1}$$

avec

$$\delta_{t+1} = \frac{y_{t+1} + b_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}} \tag{7.2}$$

Le biais à l'instant $t+1$ s'exprime donc en fonction de sa valeur à l'instant précédent et de la variable δ_{t+1} qui représente une distance entre l'observation et la densité de probabilité d'émission de l'état le plus probable au sens de la probabilité *avant*. La figure 7.1.3-(a) représente la distribution de la variable δ, pour la seconde dimension cepstrale, sur l'ensemble des phrases de test de VODIS, dans les conditions d'enregistrement *close-talk* et *far-talk*[32]. Cette figure met en évidence que la distribution de δ obtenue dans l'environnement bruité est décalée par rapport à celle obtenue dans l'environnement calme. Suite à cette observation, nous avons conclu qu'un changement soudain de l'environnement en cours de reconnaissance entraîne un changement abrupte dans la distribution de δ. Par conséquent, un algorithme de surveillance détectant une rupture dans les valeurs de δ peut déclencher une ré-initialisation du dénominateur de l'équation 7.1 qui représente l'historique de l'environnement, devenu caduque.

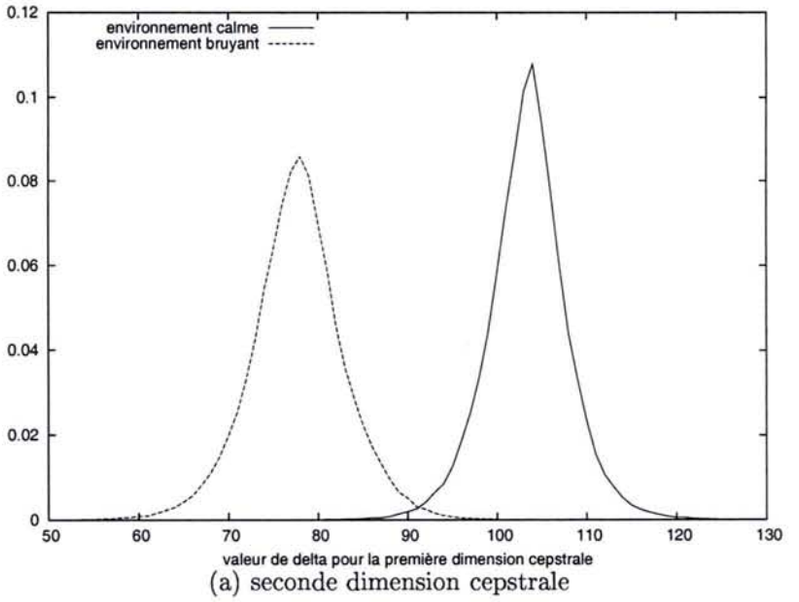
La figure 7.1.3-(b) représente la distribution de la variable δ, pour la troisième dimension cepstrale, sur l'ensemble des phrases de test de VODIS. La distribution concernant les phrases bruitées (enregistrées par un micro distant dans l'habitacle d'une voiture) apparaît en pointillés tandis que la distribution se rapportant à la parole propre (mêmes conditions mais avec un microphone proche de la bouche) est représentée en trait plein. On observe que le décalage entre les deux distributions n'est pas d'amplitude comparable sur toutes les dimensions. En effet, l'influence de l'environnement acoustique n'est pas uniforme sur toutes les dimensions cepstrales.

7.1.4 Surveillance de la variable δ

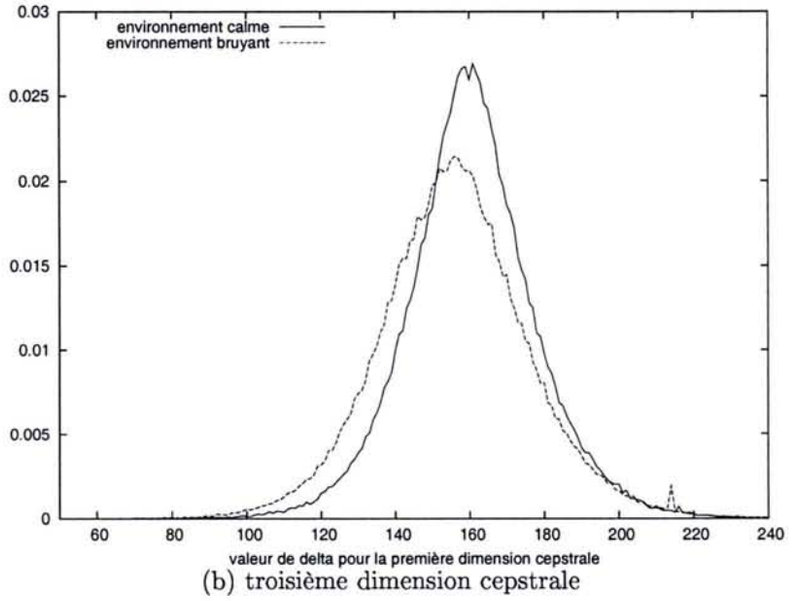
Au vu des sections 7.1.2 et 7.1.3, nous pouvons élaborer une modification de l'algorithme présenté au chapitre 5 permettant à celui-ci d'opérer dans un environnement variant inopinément.

En effet, dans la section 7.1.3, nous avons mis en évidence une différence de distribution de la valeur δ selon l'environnement ambiant. Cette observation nous incite à penser qu'un changement rapide intervenant dans la distribution de la variable δ est signe d'une variation brusque des conditions acoustiques. Or, selon la section 7.1.2, un changement d'environnement devrait s'accompagner d'une réinitialisation du processus de compensation pour que celui-ci soit efficace.

[32] voir chapitre 4, section 4.2 pour une description de cette base de test



(a) seconde dimension cepstrale



(b) troisième dimension cepstrale

FIG. 7.2 – Distribution de δ pour la deuxième et troisième dimension cepstrale pour la parole propre (*close-talk*) et bruitée (*far-talk*).

Par conséquent, nous proposons d'intégrer un mécanisme de surveillance temps-réel pour noter les variations abruptes dans la distribution de δ. Ce processus, en présence d'une variation abrupte déclenchera une réinitialisation du processus de compensation.

Cette première approche soulève deux questions :

- Comment détecter automatiquement les changements dans l'environnement acoustique, cette opération devant se faire en temps réel ?
- Quelle attitude adopter lors de la détection d'un changement :
  - la réinitialisation de la fonction de compensation doit elle se faire à la fonction identité ou à une fonction issue d'une base de donnée ?
  - la réinitialisation doit-elle se faire sur toutes les dimensions en même temps ou peut elle être réalisée indépendamment ?

Dans ce chapitre, nous utilisons et comparons plusieurs processus de surveillance permettant de repérer les changements dans l'environnement acoustique :

- L'algorithme de Shewhart.
- Le critère BIC (*Bayesian Information Criterion*).
- L'approche SVF (*Spectral Variation Function*) qui n'exploite plus la forme gaussienne de la distribution de δ mais compare à chaque instant les contextes droit et gauche de δ.

Après avoir exposé les approches classiques de la détection de changement en RAP et les caractéristiques des algorithmes de surveillance que nous nous proposons d'utiliser, nous comparerons leurs performances en termes de taux de reconnaissance et de précision de détection. Pour cela, nous utiliserons les données bruitées artificiellement proposées plus haut, simulant un environnement acoustique variant brusquement.

7.2 La détection de changements en RAP

Notre recherche s'est concentrée sur l'utilisation d'algorithmes de surveillance afin de repérer des fractures dans le comportement de δ. Dans une certaine mesure, ce repérage peut être considéré comme un problème de segmentation d'un signal en plages uniformes (ici, le signal est la séquence des variables δ).

Or, la segmentation est une problématique qui se retrouve dans de nombreux champs de la recherche sur le traitement de la parole :

- classification de signaux sonores
  - segmentation parole/non-parole (indexation d'archives)
  - segmentation en locuteurs pour (indexation d'archives)
  - détection de classes sonores dans une scène acoustique (signature sonar)
- Reconnaissance automatique (approche en temps réel le plus souvent)
  - détection de la parole (pour le début de la reconnaissance, pour le calcul d'un spectre de bruit moyen pour la soustraction spectrale)
  - détection de changement dans le locuteur/l'environnement (pour l'adaptation des modèles acoustiques)

Selon la problématique étudiée, un panel de solutions peut être envisagé. Nous nous sommes donc intéressés aux différentes méthodes utilisées pour la segmentation en RAP, afin de nous en inspirer pour développer un algorithme de surveillance adapté à notre cadre applicatif.

7.2.1 Segmentation basée sur des informations *a priori*

La détection de silences permet de détecter éventuellement les frontières entre les mots. Elle se base sur le calcul de l'énergie du signal à étudier. En effet, le niveau d'énergie (en l'absence de bruit) est plus faible dans les portions de silence que dans les périodes de parole [Delacourt, 2000]. Pour détecter les périodes de silence, on utilisera donc :

- la puissance moyenne du signal acoustique calculée sur une fenêtre glissante : si elle descend en dessous d'un seuil fixé, la partie du signal sur laquelle elle a été calculée est considérée être du silence. Le problème de cette technique réside particulièrement dans le choix de la taille de la fenêtre et celui du seuil.
- l'histogramme d'énergie à court terme permet de déterminer l'homogénéité énergétique d'une plage sonore. Les moyennes des plages homogènes se séparent alors en deux groupes : parole et silence. La valeur de séparation donne une bonne estimation d'un seuil qui peut être utilisé dans une seconde passe de segmentation.
- la variabilité de l'énergie : le silence sera caractérisé par une faible variabilité. Cette technique nécessite également le choix d'un seuil et d'une largeur de fenêtre d'étude qui dépend de la nature de la parole et de la qualité du silence.

Le silence (avec ou sans bruit de fond) se caractérise aussi par son taux de passage par zéro : il est plus élevé que celui observé dans la parole. Détecter les zones où le taux de passage par zéro est grand permet donc de repérer des zones de silences. Cette méthode, là encore, est tributaire de la sélection d'un seuil qui dépend du signal traité.

L'ensemble de ces méthodes repose donc sur la comparaison de valeurs calculées sur le signal par rapport à des valeurs déterminées par des informations *a priori* sur la nature du signal. De ces informations découlent les valeurs à attribuer au seuil de décision, la durée minimale des segments de silence [Meignier, 2002].

7.2.2 Segmentation basée sur la détection de changements

La segmentation par détection de changements essaye de se départir d'hypothèses *a priori* pour déterminer les frontières (la transition entre deux discours issus de deux locuteurs différents, ...) en ne se basant que sur les informations contenues dans le signal. Cette méthode repose sur l'utilisation d'une distance pour mesurer, à chaque instant t, la différence entre la portion de signal se situant avant t et celle se situant après. On utilisera par exemple :

- la distance de Kullbach-Leibler entre les distributions du signal, calculée sur deux fenêtres de 2 secondes situées de part et d'autre du point d'étude. Cette distance ("entropie croisée relative") atteint un pic lorsque la frontière entre deux segments de qualité acoustique différente est franchie.
- le critère d'information bayésien (BIC, *Bayesian Information Criterion* ou MDL, *Minimum Description Length*). C'est un critère qui mesure la vraisemblance d'un modèle par rapport à une séquence d'observations, pénalisée par la complexité du modèle. A chaque instant, deux hypothèses sont en compétition : la séquence de parole peut être modélisée par un seul modèle gaussien ou par deux modèles consécutifs. Le rapport des BIC des deux hypothèses passe alors par 0 lorsqu'une frontière entre deux segments de qualité acoustique différente est franchie.
- des structures neuro-mimétiques adaptatives permettant de modéliser les sources sonores d'une scène auditive simple et de repérer les transitions d'un modèle à un autre. Cette

approche très originale est développée dans [Linares, 1999] dans le cadre de la surveillance de scène auditive sous-marine.

Ces méthodes semblent au premier abord pouvoir s'affranchir de la connaissance *a priori* d'une valeur de seuil ou de la nature du signal. Cependant elles utilisent toujours des fenêtres glissantes dont la largeur reste à définir [Meignier, 2002]. Dans les travaux portant sur la segmentation en locuteurs comme [Delacourt, 2000], toutes ces méthodes ont une implémentation en temps différé et utilisent plusieurs passes dans lesquelles les largeurs de ces fenêtres sont ajustées.

7.2.3 Conclusion

Considérons le cadre dans lequel doit opérer le processus de surveillance de la variable δ. Cet algorithme doit être combiné avec le processus de compensation proposé, il doit donc être temps-réel et dépendre le moins possible de données *a priori*. Par conséquent, on ne cherchera pas à adapter les méthodes issues de la segmentation parole/non-parole, trop dépendantes des caractéristiques du bruit.

Les approches proposées dans la section 7.2.2 semblent plus adaptées à notre cadre applicatif car elles ne font pas d'hypothèse *a priori* sur la nature des signaux. Cependant, l'approche en temps différé n'est pas compatible avec les hypothèses de travail que nous nous sommes fixées pour l'utilisation de notre algorithme de compensation.

Par conséquent, nous nous sommes intéressés aux algorithmes temps réel de surveillance de processus aléatoires. Ces algorithmes sont très utilisés dans la surveillance d'automatismes en industrie par exemple [Basseville and Nikiforov, 1993]. Le plus simple de ces algorithmes, celui faisant intervenir les tableaux de contrôles de Shewhart, est exposé dans la section suivante.

7.3 L'algorithme de Shewhart

7.3.1 Cadre théorique de l'algorithme de Shewhart

Comme décrit dans [Basseville and Nikiforov, 1993] l'algorithme de surveillance de *Shewhart* permet de repérer le moment où une variable cesse de suivre sa distribution.

Considérons une séquence d'observations indépendantes $Z = \{z_1, \ldots, z_K\}$. Soit $p_\theta(z)$ la densité de probabilité de z, dépendante du paramètre θ. Supposons qu'à l'instant de commutation t_c, la valeur de θ passe de θ_0 à $\theta_1 \neq \theta_0$. C'est-à-dire qu'à l'instant t_c, la distribution de la variable z passe brusquement de $p_{\theta_0}(z)$ à $p_{\theta_1}(z)$

L'objectif de l'algorithme de surveillance de Shewhart est de détecter ce changement dans la valeur du paramètre. Cet algorithme est le plus élémentaire des algorithmes de détection de changement en temps réel. C'est ce qu'on appelle un algorithme de contrôle de qualité : il observe une séquence de vecteurs et déclenche une alarme au moment où les observations ne correspondent plus à un schéma pré-établi.

A chaque instant t, on émet deux hypothèses sur la séquence d'observations en cours. La première hypothèse pose que le paramètre θ contrôlant la distribution de z est θ_1. La deuxième suggère que ce paramètre est égal à θ_0 :

$$\begin{aligned} H_0 &: \theta = \theta_0 \\ H_1 &: \theta = \theta_1 \end{aligned} \tag{7.3}$$

Comme la plupart des algorithmes de détection de changement, on utilise ici le rapport de vraisemblance

$$s_\tau = ln(\frac{p_{\theta_1}(z_\tau)}{p_{\theta_0}(z_\tau)})$$

et plus particulièrement sa moyenne sur une fenêtre glissante de N éléments :

$$S_t = \sum_{\tau=t-(N-1)}^{t} s_\tau$$

pour déclencher une alarme (s_τ est appelée *sufficient statistic*). L'alarme est lancée quand S_t (la *fonction de décision*) est supérieure à un seuil, généralement fixé expérimentalement.

Un cas particulier du contrôle de Shewhart est le changement de moyenne entre deux processus gaussiens de même variance. C'est le cas que nous utiliserons puisqu'il s'applique totalement à la surveillance de la variable δ présentée à la section 7.1.3. Voici les dérivations de l'algorithme dans ce cas particulier : Considérons les densités de probabilité de type gaussien de moyenne m_0 et m_1 et même variance σ :

$$p_{\theta_0}(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-m_0)^2}{2\sigma^2}}$$

et

$$p_{\theta_1}(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-m_1)^2}{2\sigma^2}}$$

Ici,

$$\theta_0 = m_0 \text{ et } \theta_1 = m_1$$

La *sufficient statistic* est :

$$\begin{aligned} s_\tau &= log\left(\frac{\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z_\tau-m_1)^2}{2\sigma^2}}}{\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z_\tau-m_0)^2}{2\sigma^2}}}\right) \\ &= \frac{(z_\tau - m_0)^2}{2\sigma^2} - \frac{(z_\tau - m_1)^2}{2\sigma^2} \\ &= \frac{1}{2\sigma^2}\left(2z_\tau(m_1 - m_0) - (m_1^2 - m_0^2)\right) \\ &= \frac{m_1 - m_0}{\sigma^2}\left(z_\tau - \frac{m_0 + m_1}{2}\right) \\ &= \beta\left(\frac{z_\tau - m_0}{\sigma} - \frac{\beta}{2}\right) \end{aligned} \tag{7.4}$$

Où β est la magnitude pondérée du changement :

$$\beta = \frac{m_1 - m_0}{\sigma}$$

Donc la fonction de décision est :

$$S_t = \beta \sum_{\tau=t-(N-1)}^{t} \left(\frac{z_\tau - m_0}{\sigma} - \frac{\beta}{2}\right)$$

et la loi de décision est :

$$d = \begin{cases} 0 \quad \text{(pas d'alarme) si} \quad S_t < h; \\ \\ 1 \quad \text{(alarme) si} \quad S_t \geq h \end{cases}$$

où h est un seuil à déterminer expérimentalement.

Ainsi, si on considère $m_1 > m_0$, l'alarme est déclenchée dès que :

$$\bar{z}_t \geq m_0 + \kappa \frac{\sigma}{\sqrt{N}} z$$

Avec

$$\bar{z}_t = \frac{1}{N} \sum_{\tau=t-(N-1)}^{t} z_\tau$$

où κ est un facteur pondérant la déviation standard.

Pour une loi de déclenchement symétrique (c'est-à-dire où on ne fait aucune hypothèse sur $m_1 > m_0$), on utilisera :

$$|\bar{z}_t - m_0| \geq \kappa \frac{\sigma}{\sqrt{N}}$$

Le critère de déclenchement dépend donc de la variance de la distribution de la variable et d'un facteur de pondération κ à déterminer.

En d'autres termes, le changement de comportement est détecté lorsque la différence entre la moyenne calculée sur une fenêtre de taille N et m_0 devient trop importante par rapport à la variance σ de la distribution.

7.3.2 Application à la surveillance de la variable δ

Nous proposons d'utiliser l'algorithme de surveillance de Shewhart pour contrôler la variable δ. En effet, nous avons vu à la section 7.1.3 que la distribution de δ pouvait être modélisée par une Gaussienne sur un environnement variant lentement. Nous avons vu, de plus, que la moyenne de cette distribution variait lorsque l'environnement acoustique changeait. Surveiller le changement de comportement de la variable δ revient donc à surveiller un changement de moyenne dans une distribution gaussienne. Or il s'agit exactement du cadre d'application de l'algorithme de surveillance de Shewhart.

Considérons que l'environnement acoustique change brusquement à l'instant t de env_0 à env_1. La distribution de δ avant t (resp. après) peut être modélisée par une Gaussienne $mode_0$ (resp. $mode_1$) de moyenne et variance (m_0, σ) (resp. (m_1, σ)). Notre but est de détecter, de façon réactive, le basculement de env_0 vers env_1.

L'utilisation de l'algorithme de surveillance de Shewhart suppose la connaissance de la moyenne m_0 de la distribution (gaussienne) de δ avant le changement d'environnement. Dans notre approche, ce paramètre est estimé pendant les premières trames de chaque phrase et affiné tant qu'aucun changement d'environnement n'est repéré.

Bien que la variable δ soit un vecteur de dimension D (D étant la dimension d'un cepstre), nous considérerons pour les calculs qu'il s'agit d'un scalaire. Les calculs dérivés s'étendent facilement à une version multi-dimensionnelle. Une discussion sur la façon dont sont liées les alarmes sur les différentes dimensions est proposée en section 7.3.5.

Le test repose classiquement sur la comparaison, à chaque trame t, de la vraisemblance de deux hypothèses :

H_0 l'environnement est env_0 jusqu'en t.

H_1 l'environnement env_1 a succédé à env_0 dans les N dernières trames précédant t.

Ici, une alarme sera lancée à l'instant t dans le cas où :

$$|\bar{\delta}_t - m_0| \geq \kappa \frac{\sigma}{\sqrt{N}} \tag{7.5}$$

où :

- $\bar{\delta}_t$ est la moyenne sur une fenêtre de N cepstres de δ.
- m_0 et σ sont la moyenne et la variance de δ depuis la dernière alarme.
- κ est le seuil.

En l'absence d'informations *a priori* sur la nature du bruit (associé à l'environnement acoustique où à l'identité du locuteur), on ne peut pas calculer la valeur optimale de κ. Dans la suite de la discussion, κ est donc fixée empiriquement.

En ce qui concerne l'interaction du processus de surveillance avec l'algorithme de compensation, l'alarme déclenche la réinitialisation du processus de compensation. Le processus de compensation peut être réinitialisé de plusieurs façons (nous en discuterons à la section 7.3.6). Dans la suite des expérimentations et sauf mention du contraire, la réinitialisation consiste à remettre à zéro le dénominateur de l'expression itérative du biais (équation 7.1). Ceci a pour effet de supprimer l'historique de l'environnement (voir section 7.1.2). La nouvelle valeur du biais est alors initialisée à la dernière valeur obtenue avant l'alarme afin de conserver une certaine continuité.

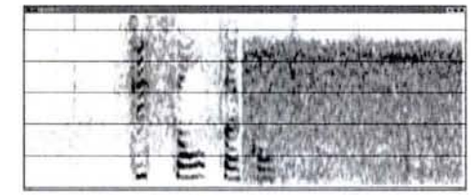
Afin d'illustrer le comportement de l'algorithme de surveillance de Shewhart et son interaction avec notre algorithme de compensation (*Biais*)), nous avons effectué la reconnaissance d'une phrase de test corrompue par la méthode *échelon*. La figure 7.3 montre l'évolution de la valeur de *Biais* sur la première dimension cepstrale, lors de cette reconnaissance.

Sur la figure 7.3, on peut voir :

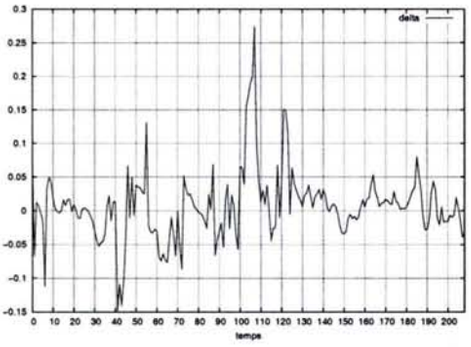
- en (a) : le spectrogramme de la phrase à reconnaître : sa deuxième moitié est bruitée. Le mot prononcé est "six cent cinquante huit". Les deux premiers mots sont prononcés sans bruit de fond et le bruit additif est ajouté au niveau de la sifflante de "cinquante".
- en (b) : l'évolution de la variable δ. On remarque une coupure à $t = 110$ où une alarme (et une réinitialisation du dénominateur de l'expression du biais) a été lancée pour cette dimension.
- la raison de cette alarme est montrée en (c) où l'on peut voir l'évolution de la moyenne de longue durée de δ et l'évolution de la moyenne sur une fenêtre de 10 cepstres. En $t = 110$, l'écart entre ces deux moyennes devient supérieur à la déviation standard pondérée par le seuil et l'alarme (ainsi la réinitialisation du biais) est déclenchée sur cette dimension.
- en (d) : la valeur du biais sur cette dimension. On observe la coupure en $t = 110$ et la convergence vers une valeur différente de celle observée avant $t = 110$. A la réinitialisation, le dénominateur de l'expression du biais est remis à zéro et la valeur du biais est initialisée sur la dernière valeur prise avant l'alarme. C'est pourquoi on observe une continuité dans la courbe du biais à l'instant de l'alarme.

La figure 7.4 montre l'évolution de la valeur de *Biais* sur la deuxième dimension cepstrale, lors de la reconnaissance de la même phrase.

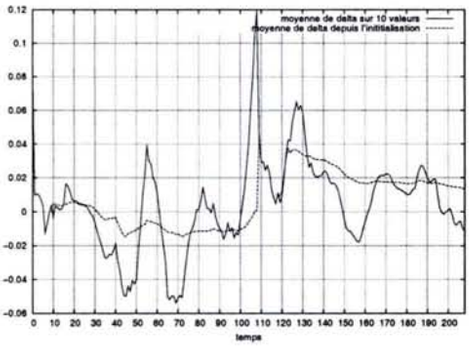
Sur la figure 7.4, on peut voir que sur la deuxième dimension cepstrale (c_1), il y a 3 réinitialisations du processus de compensation (à $t = 87$, $t = 120$ et $t = 175$). Après chaque alarme, le



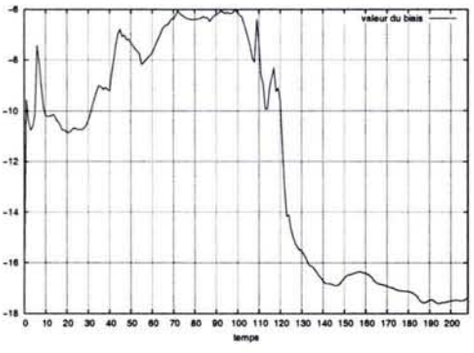
(a) spectrogramme d'une phrase de test bruitée par *échelon*



(b) valeur de δ de la première dimension cepstrale

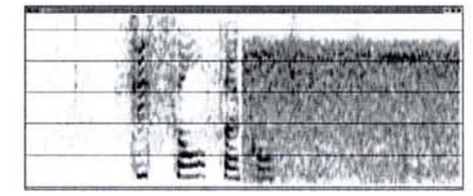


(c) moyenne de δ sur 10 cepstres et depuis la dernière alarme

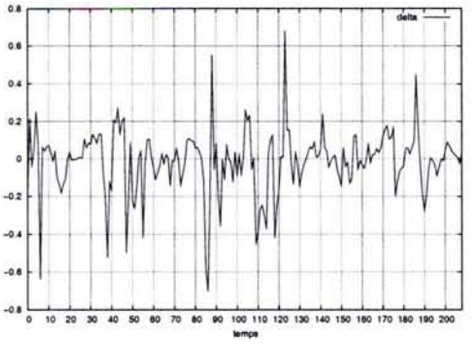


(d) valeur de *Biais* pour la première dimension cepstrale

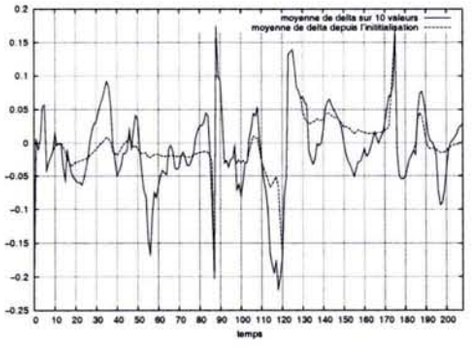
FIG. 7.3 – Evolution du *Biais* sur la première dimension cepstrale (c_0) pour une phrase de test *échelon* avec une réinitialisation commandée par l'algorithme de Shewhart.



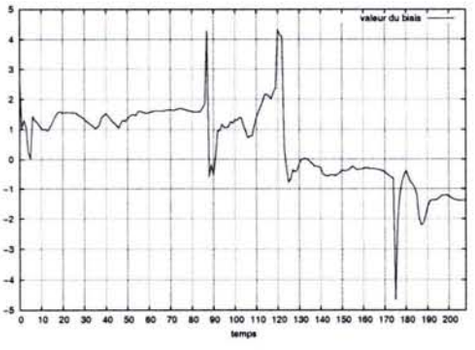
(a) spectrogramme d'une phrase de test bruitée par *échelon*



(b) valeur de δ de la deuxième dimension cepstrale



(c) moyenne de δ sur 10 cepstres et depuis la dernière alarme



(d) valeur de *Biais* de la deuxième dimension cepstrale

FIG. 7.4 – Evolution du *Biais* sur la deuxième dimension cepstrale (c_1) pour une phrase de test *échelon* avec une réinitialisation commandée par l'algorithme de Shewhart.

biais de cette dimension est réinitialisé et converge à nouveau (après une phase d'adaptation). La première alarme semble être déclenchée à la transition entre la voyelle "i" et la deuxième sifflante, issu de "cent". La deuxième alarme est déclenchée suite à l'ajout du bruit. La dernière correspond à la fin de la parole. On peut donc conclure que le processus de surveillance ne permet pas seulement de repérer les transitions dans l'environnement acoustique mais peut aussi relever les enchaînements des phonèmes. Ceci peut présenter un problème car il est possible que réinitialiser trop souvent l'algorithme de compensation empêche le biais de converger et rende le mécanisme de compensation inefficace[33]. Ce problème pourrait être partiellement levé en élargissant la fenêtre d'analyse de sorte qu'elle puisse englober plus d'une unité acoustique. En effet, un phonème dure de 10 à 30 trames : une fenêtre d'analyse de plus de 30 trames permet de lisser les discontinuités observées aux transitions entre phonèmes.

Les figures 7.3 et 7.4 montrent que les indices de transitions ne sont pas aussi évidents dans toutes les dimensions cepstrales. En effet, une discontinuité (transition entre deux phonèmes ou changement brusque de l'environnement acoustique) peut apparaître sur certaines dimensions sans affecter les autres (voir la figure 7.1.3 de la section 7.1.3 de ce chapitre). C'est pourquoi dans l'implémentation utilisée pour cette expérience, chaque dimension cepstrale est surveillée par un processus de Shewhart. Ce processus, lorsqu'il lance une alarme, déclenche la réinitialisation du processus de compensation pour cette dimension cepstrale. Nous verrons par la suite que ce n'est pas la seule solution envisageable.

7.3.3 Problématiques

On peut déduire de la loi de décision donnée par l'équation 7.5 et les choix d'implémentation de l'algorithme que :

- il existe un temps de latence d'au moins N trames entre l'apparition du changement et son éventuelle détection. Cette valeur est réglée empiriquement.
- le seuil de déclenchement κ est réglé empiriquement et dépend du niveau de bruit.
- dans l'implémentation proposée, la réinitialisation consiste à supprimer l'historique de l'environnement dans le calcul du biais (*remise à zéro de l'historique*). Il est possible d'imaginer d'autres processus de réinitialisation.
- dans l'implémentation exposée, chaque dimension est décorrélée : une alarme signalant un changement peut être lancée pour une dimension cepstrale sans pour autant être déclenchée sur les autres dimensions. Cependant, une autre solution est envisageable, où la réinitialisation peut se faire sur toutes les dimensions cepstrales en même temps, selon un critère déterminé avec l'ensemble des dimensions.

L'utilisation du processus de Shewhart tel que nous l'avons décrit est donc conditionnée par 4 choix :

- la taille de la fenêtre d'analyse,
- la valeur du seuil,
- le lien qui unit les dimensions dans la détection de changement,
- le processus de réinitialisation déclenché par l'alarme.

Dans la suite de cette section, nous étudierons l'influence de ces facteurs sur le taux de reconnaissance. Pour l'ensemble des expériences, nous utiliserons la tâche *échelon* car c'est la plus simple et elle permet de mettre en évidence les mécanismes qui aident aux choix des paramètres décrit plus haut.

[33] voir la section 7.5.2 de ce chapitre à ce sujet

7.3.4 Taille de la fenêtre d'analyse

La figure 7.5 représente les taux de reconnaissance obtenus par *Biais* avec une réinitialisation commandée par l'algorithme de Shewhart sur la base de test Aurora3 corrompue par *échelon*, le RSB sur la partie bruitée étant de $8.2dB$. Pour cette expérience, la largeur N de la fenêtre

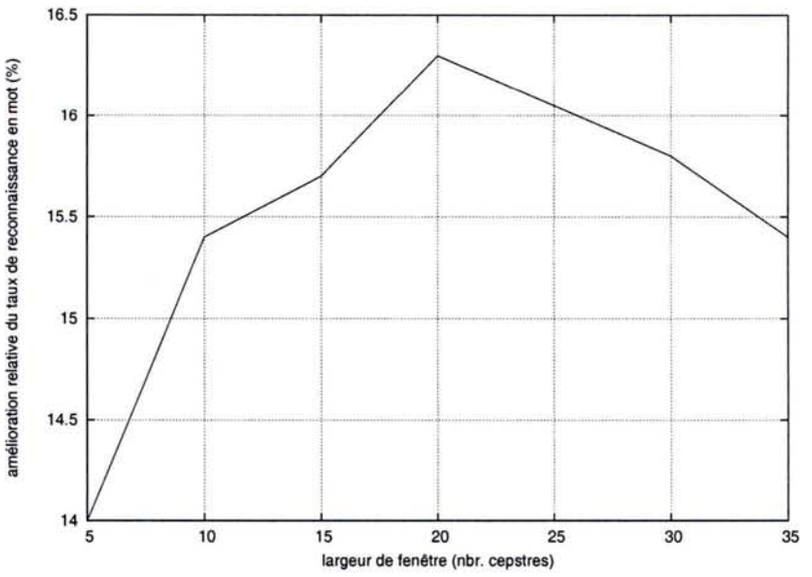


FIG. 7.5 – Taux de reconnaissance en mot sur Aurora3 corrompu artificiellement par un bruit soudain à mi-phrase (RSB=8.2dB sur partie bruitée) (*échelon*) pour *Biais* et réinitialisation par l'algorithme de Shewhart.

glissante sur laquelle est calculée la moyenne de δ varie de 50ms ($N = 5$) à 350ms ($N = 35$), sachant que la durée moyenne d'une phrase de test est de 7 secondes (la plus courte faisant 2 secondes) et que le bruit est ajouté à mi-phrase. La valeur du seuil pour toutes les dimensions est fixée à $\kappa = 3.0$ (cette valeur est discutée plus loin).

On observe que le maximum du taux de reconnaissance, pour cette tâche spécifique, est atteint pour une taille de fenêtre de 200ms ($N = 20$). Pour des valeurs de fenêtre trop petites, le taux de reconnaissance diminue significativement, la moyenne de δ n'étant pas précisément estimée sur ces courts intervalles. De plus, il faut remarquer que la taille de fenêtre optimum dépend du type de variations auxquels est soumis l'environnement acoustique. Par exemple, une fenêtre de taille N pourrait être inadaptée pour un environnement changeant plus d'une fois durant N trames.

L'utilisation de cette approche est donc soumise à un choix (*a priori*) de la taille de fenêtre qui doit tenir compte des considérations suivantes :

- une fenêtre trop étroite peut fournir une mauvaise estimation de la moyenne à court terme de δ et provoquer une fausse alarme.
- une fenêtre trop large peut englober une variation brève et abrupte dans l'environnement. Ce lissage peut empêcher le déclenchement de l'alarme.

Dans la suite de notre travail, notre choix s'est porté sur une fenêtre de 200ms, sauf mention contraire.

7.3.5 Détermination de la valeur de seuil κ

Comme vu dans l'équation 7.5 décrivant le critère d'alarme, le processus de surveillance de Shewhart requiert une valeur de seuil κ. Dans l'implémentation choisie, cette valeur est fixe et définie expérimentalement. Il découle de l'expression de l'équation 7.5 que la valeur de κ qui permettra de détecter un changement dans la moyenne de la distribution de δ dépend de l'amplitude de ce changement. Or la valeur de cette amplitude est influencée par plusieurs facteurs dont le plus important est la valeur du rapport signal à bruit.

Cette constatation est confirmée par l'expérience suivante. Plusieurs bases de test de type *échelon* ont été créées à partir de Aurora3, en faisant varier le RSB. Pour chacune de ces bases, une reconnaissance a été faite avec *Biais* intégrant le processus de réinitialisation inspiré de l'algorithme de Shewhart en faisant varier le seuil κ. La figure 7.6 représente l'augmentation des taux de reconnaissance en mots par rapport à *Référence* en fonction du RSB sur la partie bruitée et du seuil.

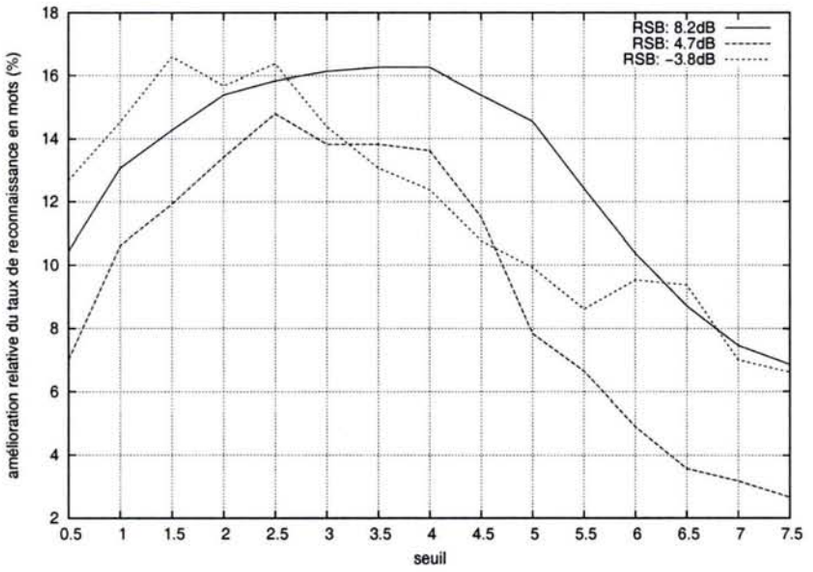


FIG. 7.6 – Amélioration du taux de reconnaissance en mots sur Aurora3 *échelon* par rapport à *Référence* pour *Biais* utilisant l'algorithme de Shewhart pour sa réinitialisation, indépendamment sur chaque dimension (RSB relevé sur la partie bruitée).

On observe que, pour chaque RSB, le taux de reconnaissance varie en fonction du seuil et passe par un maximum pour une valeur de seuil optimale proche de 3. La valeur de seuil optimale varie en fonction du RSB et décroît légèrement à mesure que le RSB diminue.

Indépendance des dimensions

Dans une deuxième série d'expériences, nous avons modifié le processus de déclenchement des alarmes. Dans cette nouvelle version, on ne dispose pas d'un test de surveillance par dimension, permettant ainsi d'effectuer une réinitialisation sur chaque dimension séparément. Au contraire,

le processus de surveillance s'applique à toutes les dimensions à la fois et si une alarme est lancée la réinitialisation des paramètres de la fonction de compensation se fait sur toutes les dimensions à la fois.

La figure 7.7 représente l'augmentation des taux de reconnaissance en mots par rapport à *Référence* en fonction du RSB sur la partie bruitée et du seuil. Comme observé précédemment,

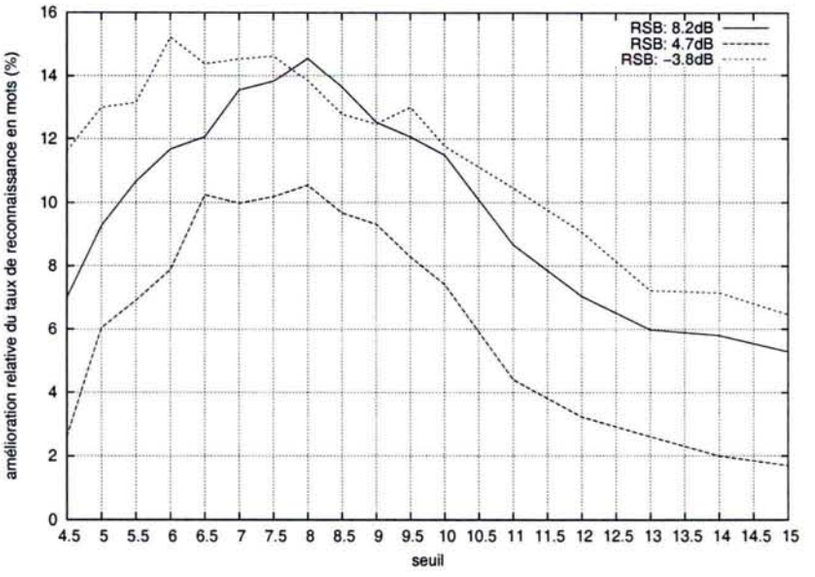


FIG. 7.7 – Amélioration du taux de reconnaissance en mots sur Aurora3 *échelon* par rapport à *Référence* pour *Biais* utilisant l'algorithme de Shewhart pour sa réinitialisation, en se basant sur toutes les dimensions cepstrales (Shewhart-Lie) (RSB relevé sur la partie bruitée).

les maximums sont atteints pour des valeurs de seuil qui diminuent à mesure que le RSB (sur la partie bruitée) diminue. Cependant on peut voir ici que ces maximums sont bien plus faibles que dans le cas précédent : l'amélioration relative par rapport à *Référence* est moins importante lorsque l'algorithme de surveillance et de réinitialisation est appliqué à toutes les dimensions que dans le cas où chaque dimension est traité indépendamment.

Dans la suite on distinguera deux versions de l'algorithme de surveillance de Shewhart appliqué au déclenchement de la réinitialisation du processus de compensation :

- *Shewhart-nonLié* : il existe un processus de surveillance par dimension, et une réinitialisation du *Biais* peut intervenir sur une dimension indépendamment des autres.
- *Shewhart-Lié* : le processus surveille toutes les dimensions en même temps et la réinitialisation se fait sur toutes les dimensions.

7.3.6 Alternative à la remise à zéro de l'historique lors de la réinitialisation.

Dans les expériences proposées précédemment, nous avons toujours supposé qu'une alarme déclenchée produisait une réinitialisation du dénominateur de l'expression du biais, supprimant ainsi tout historique du calcul du biais. Dans l'expérience suivante, nous avons supposé que les paramètres de la transformation pouvaient s'initialiser avec des valeurs déjà obtenues au cours

de la reconnaissance. La détermination du biais après une alarme ne se fait donc pas à partir d'un historique vierge.

Considérons une phrase de test prononcée dans un environnement acoustique *env* qui est la succession dans le temps de env_0, env_1 et env_0. Si une alarme est déclenchée à chaque passage d'un environnement à un autre, les paramètres de la fonction de compensation lors du deuxième changement peuvent être initialisés à ceux obtenus avant le premier.

On dispose d'un tableau où sont enregistrés, pour chaque date d'alarme $t = a$:
- les paramètres de la transformation θ_a avant l'alarme. Dans le cas de *Biais*, θ_a :
  - le biais b_a, dernière valeur du biais obtenu avant l'alarme a.
  - l'historique $hist_a$ calculé entre la date de l'alarme a et celle qui la précédait.
- la moyenne m_a de la variable δ, calculée entre la date de l'alarme a et celle qui la précédait.
- la variance σ_a associée à m_a

Pour une alarme à l'instant t, il est possible d'initialiser les paramètres de la transformation à θ_t si la condition suivante :

$$|\bar{\delta}_d - m_\alpha| \leq \kappa_d \frac{\sigma_\alpha}{\sqrt{N}} \tag{7.6}$$
$$\alpha = \arg\min_a \frac{|\bar{\delta}_d - m_a|}{\kappa_d \frac{\sigma_a}{\sqrt{N}}}$$

est remplie, où $\bar{\delta}_d$ est la moyenne de la projection de δ sur d pour les N derniers échantillons. Dans ce cas, après l'alarme au temps t, la réinitialisation du processus de compensation *Biais* est telle que :

$$b_{t+1} = b_a + \frac{y_{t+1} + b_a - \mu}{hist_a + \frac{1}{\sigma^2}}$$

où μ et σ sont les moyennes et variances de l'état le plus probable au sens de la probabilité *avant*, comme décrit au chapitre 5.

Dans le cas où aucune valeur consignée dans le tableau ne permet d'obtenir une réponse positive au test, la transformation est initialisée par annulation de l'historique.

Ce mécanisme permet donc de réinitialiser les paramètres de la fonction de compensation (valeur de biais et valeur de l'historique) aux paramètres obtenus dans des conditions similaires. La mesure de similarité entre les conditions est inspiré du test de l'équation 7.5 et est donné dans l'équation 7.6. Cette solution a été choisie afin que la mesure garde une correspondance avec le test permettant la détection de ruptures dans la séquence des δ. Cependant, d'autres solutions peuvent être envisagées pour élire la valeur à laquelle le paramètre de la fonction de compensation doit être initialisée après une alarme. Par exemple il est possible d'utiliser une distance de Kullback-Leibler.

Le tableau 7.4 donne les taux de reconnaissance en mots pour les phrases de test Aurora3 bruitées par *aléatoire* quand :
- aucune compensation n'est utilisée (*Référence*)
- *Biais* est utilisé avec le système de surveillance Shewhart, les transformations de compensations étant réinitialisées par remise à zéro de l'historique et initialisation du biais à zéro après chaque alarme, pour chaque dimension séparément (*Biais-Shewhart avec RAZ Tout*)
- *Biais* est utilisé avec le système de surveillance Shewhart, les transformations de compensations étant réinitialisées par remise à zéro de l'historique après chaque alarme, pour chaque dimension séparément (*Biais-Shewhart avec RAZ Historique*)

- *Biais* est utilisé avec le système de surveillance Shewhart, les transformations de compensations étant réinitialisées grâce au test de l'équation 7.6 (*Biais-Shewhart avec réinitialisation*).

Référence	*Biais-Shewhart avec RAZ Tout*	*Biais-Shewhart avec RAZ Historique*	*Biais-Shewhart avec réinitialisation*
67.67%	68.34%	70.28 %	71.13%

TAB. 7.4 – Taux de reconaissance en mots pour les phrases de test Aurora3 *aléatoire* (RSB sur parties bruitées : 8.2dB).

Pour ces expériences, la taille de la fenêtre d'analyse est de $N = 20$ et le seuil fixé à 2.0. Cette expérience a été menée sur une base bruitée par la tâche *aléatoire* car c'est cette tâche qui est la plus difficile. Une réinitialisation efficace est donc indispensable. Enfin cette tâche met en évidence l'intérêt de récupérer les informations obtenues dans les phases antérieures du processus de compensation.

On peut voir dans le tableau 7.4 que ce mécanisme de réinitialisation après alarme donne de bons résultats dans cet exercice. Ce mécanisme de réinitialisation peut être encore amélioré en disposant d'un ensemble de tuples $(\theta_a, m_a, \sigma_a)$ obtenus lors de la reconnaissance sur des phrases précédentes. Cet ensemble constituerait alors une collection (ou *codebook*) de valeurs de réinitialisation. Cette solution n'a pas encore été implémentée.

7.3.7 Conclusion sur l'algorithme de Shewhart

Nous avons donc vu que l'algorithme de Shewhart, bien qu'étant le plus simple des algorithmes de surveillance, permet de repérer efficacement des ruptures dans le comportement de la variable δ. Ces ruptures peuvent être interprétées comme des signes de changements dans l'environnement acoustique. L'implémentation de cet algorithme peut prendre plusieurs formes que nous avons explorées :

- différentes tailles de fenêtre d'analyse (dépendantes de la nature du bruit)
- différentes valeurs de seuils (dépendantes de la dimension, de la nature et du niveau de bruit)
- la possibilité de joindre ou dissocier les dimensions dans l'analyse des changements
- la possibilité de réinitialiser des paramètres à des valeurs non-triviales

Sous toutes ces formes, l'utilisation de cet algorithme suppose l'établissement de valeurs de seuil, et d'une largeur de fenêtre d'analyse qui dépendent fortement de la nature de l'environnement acoustique.

Dans la section suivante, nous cherchons à nous affranchir de ces contraintes en utilisant deux autres mécanismes de surveillance.

7.4 Autres mécanismes de détection

7.4.1 Critère d'information bayésien

Le critère d'information bayésien (BIC, *Bayesian Information Ciriterion*) est également connu sous le nom de MDL (*Maximum Description Length*) ou encore critère de Rissanen. C'est

un critère de vraisemblance pénalisé par la complexité du modèle, c'est à dire son nombre de paramètres ([Basseville and Nikiforov, 1993]).

Le critère BIC permet de sélectionner un modèle parmi plusieurs pour modéliser la séquence d'observations. Il est utilisé dans des domaines comme la segmentation en locuteurs [Delacourt, 2000], [Almera, 2004], [Bonastre *et al.*, 2000].

Dans [Chen and Gopalakrishnan, 1998] le critère BIC est utilisé pour une segmentation plus générale en classes acoustiques (parole, musique, silence ...). C'est l'un des premiers travaux utilisant BIC. On y considère que la séquence de vecteurs paramétriques peut être modélisée localement par une gaussienne multidimensionnelle. Le critère BIC est alors utilisé pour détecter le passage d'un modèle à un autre, effectuant ainsi une segmentation du signal de parole.

L'utilisation de BIC que nous proposons dans cette section s'inspire beaucoup de l'approche de [Chen and Gopalakrishnan, 1998]. Cependant la séquence de variables surveillée dans nos travaux n'est pas la suite des vecteurs paramétriques mais la séquence des variables δ. S'il peut être contestable de modéliser la parole prononcée dans un environnement spécifique par une gaussienne (ou un mélange de Gaussiennes), on a démontré en section 7.1.3 de ce chapitre que δ pouvait l'être parfaitement. Par conséquent, BIC devrait se révéler être un bon algorithme de surveillance pour la variable δ [Barreaud *et al.*, 2004].

Cadre théorique du BIC

Soit $X = \{x_1, \ldots, x_N\}$ une séquence de N vecteurs. Soit M un modèle du processus X dépendant de m paramètres. Le critère BIC associé au modèle M et à la séquence X a pour expression :

$$BIC(X, M) = log\mathcal{L}(X, M) - \lambda \frac{m}{2} \log N$$

où

- le premier terme $\mathcal{L}(X, M)$ (la vraisemblance du modèle M pour la séquence d'observations X) reflète l'ajustement du modèle aux données
- le deuxième terme correspond à la complexité du modèle, pondérée par un facteur λ.

Le critère BIC permet de sélectionner une modélisation parmi plusieurs pour les mêmes données. La modélisation qui maximise ce critère est la modélisation qui correspond le plus aux données en terme de vraisemblance et dont la complexité est la plus faible.

Par exemple :

- Soit $\{x_1, \ldots, x_N\}$ une séquence de N vecteurs
- Soit le test d'hypothèses suivant :

  H_0 : la séquence est générée par un seul processus Gaussien multi-dimensionnel :

  $$X_0 = \{x_1, \ldots, x_N\} \sim \mathcal{N}(\mu_0, \Sigma_0)$$

  H_1 : la séquence est générée par deux processus Gaussiens multi-dimensionnels successifs :

  $$X_1 = \{x_1, \ldots, x_t\} \sim \mathcal{N}(\mu_1, \Sigma_1) \quad \text{et} \quad X_2 = \{x_{t+1}, \ldots, x_N\} \sim \mathcal{N}(\mu_2, \Sigma_2)$$

  où
  - μ représente la moyenne
  - Σ représente la matrice de covariance des processus gaussiens
  - t est la date à laquelle on passe du premier processus (μ_1, Σ_1) au deuxième (μ_2, Σ_2).

Alors, en posant $R(t)$, le rapport de vraisemblance entre les hypothèses H_0 et H_1 (GLR, *Generalized Likelihood Ratio*, [Bonastre *et al.*, 2000]) :

$$R(t) = \frac{\mathcal{L}(X_0, \mu_0, \Sigma_0)}{\mathcal{L}(X_1, \mu_1, \Sigma_1)\mathcal{L}(X_2, \mu_2, \Sigma_2)}$$

Le logarithme de ce rapport de vraisemblance donne :

$$\begin{aligned} logR(t) &= log\mathcal{L}(X_0, \mu_0, \Sigma_0) - log\mathcal{L}(X_1, \mu_1, \Sigma_1) - log\mathcal{L}(X_2, \mu_2, \Sigma_2) \\ &= -\frac{N}{2}log|\Sigma| + \frac{t}{2}log|\Sigma_1| + \frac{N-t}{2}log|\Sigma_2| \\ &\quad -\frac{1}{2}\sum_{\tau=1}^{N}(x_\tau - \mu_0)^T\Sigma_0^{-1}(x_\tau - \mu_0) \\ &\quad +\frac{1}{2}\sum_{\tau=1}^{t}(x_\tau - \mu_1)^T\Sigma_1^{-1}(x_\tau - \mu_1) + \frac{1}{2}\sum_{\tau=t+1}^{N}(x_\tau - \mu_2)^T\Sigma_2^{-1}(x_\tau - \mu_2) \end{aligned} \tag{7.7}$$

A un instant donné t, le critère de décision permettant de favoriser une hypothèse par rapport à l'autre est la différence $\Delta BIC(t)$ entre les valeurs de BIC associées à H_0 et H_1. Cette différence s'écrit :

$$\Delta BIC(t) = logR(t) - \lambda P$$

où la complexité P est donnée par :

$$P = \frac{1}{2}(D + \frac{D(D+1)}{2})log(N)$$

dans le cas où les matrices de covariance sont symétriques et la dimension de l'espace paramétrique est D (D paramètres pour le vecteur de moyenne, et $D(D+1)/2$ pour les paramètres de la matrice). Dans le cas où la matrice est considérée diagonale, comme c'est le cas dans le cadre d'application de notre algorithme de compensation,

$$P = Dlog(N)$$

Le paramètre λ est un paramètre théoriquement fixé à 1.

Si $\Delta BIC(t)$ est négatif, alors le modèle à deux gaussiennes est privilégié (hypothèse H_1) et on peut déclarer qu'il y a effectivement eut une transition entre deux processus gaussiens, en t, dans la séquence X. Dans ce cas, on peut dire que la séquence des observations X peut se décrire de la façon suivante :

- la première partie de la séquence (X_1) a une densité de probabilité estimée à

$$\mathcal{N}(\mu_{X_1}, \Sigma_{X_1})$$

- la deuxième partie (X_2) a une densité de probabilité estimée à

$$\mathcal{N}(\mu_{X_2}, \Sigma_{X_2})$$

Le paramètre de pondération λ est théoriquement fixé à 1. Cependant, comme vu dans [Delacourt, 2000], ce facteur, introduit dans [Chen and Gopalakrishnan, 1998] est idéal pour régler (expérimentalement) la sensibilité du processus de surveillance et ainsi influencer le taux de fausse alarme (c'est-à-dire de détection sans changement effectif). Nous conservons ce facteur de pondération dans l'application de BIC à notre algorithme de compensation. Comme l'algorithme de Shewhart, cette version de BIC utilise donc un seuillage mais nous montrerons par la suite qu'il n'est pas nécessaire de le régler avec précision pour obtenir de bons résultats.

Application à la surveillance de la variable δ

Dans [Barreaud *et al.*, 2004] nous avons utilisé le critère BIC pour surveiller les changements dans la distribution de la variable δ. Le principal attrait de cette technique est qu'en théorie, elle ne fait pas intervenir de valeur de seuil. Par conséquent, le processus de détection devrait s'affranchir d'une étape d'estimation du seuil dépendant du RSB, comme c'est le cas lorsqu'on utilise l'algorithme de Shewhart[34].

Comme représenté dans la figure 7.4.1, à chaque instant t on évalue le rapport des valeurs des critères pour deux hypothèses pouvant modéliser la séquence des δ jusqu'à l'instant t. Ces hypothèses sont :

H_0 : la séquence est générée par un seul processus Gaussien multi-dimensionnel.

H_1 : la séquence est générée par deux processus Gaussiens multi-dimensionnels successifs.

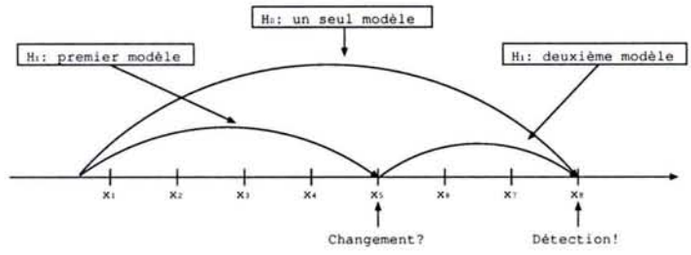


FIG. 7.8 – Critère d'alarme reposant sur la comparaison des valeurs de BIC pour deux hypothèses.

Nous avons choisi d'utiliser des gaussiennes multi-dimensionnelles pour modéliser les séquences de δ et ainsi détecter les changements de régime. Ce choix impose le déclenchement de l'alarme pour toutes les dimensions cepstrales, et non pas sur chaque dimension indépendamment. Ainsi, il sera possible de mesurer avec efficacité le taux de fausses alarmes et d'omissions de cette approche, comme montré en section 7.5.2.

Les paramètres des modèles utilisés dans les hypothèses sont réestimés à chaque trame, sur une fenêtre glissante. On définit :

- a, l'instant de la dernière alarme
- t, l'instant du test
- N, la taille de la fenêtre d'analyse
- (μ_0, σ_0), les moyennes et variances du modèle de l'hypothèse H_0.
- (μ_1, σ_1), les moyennes et variances du premier modèle de l'hypothèse H_1.
- (μ_2, σ_2), les moyennes et variances du deuxième modèle de l'hypothèse H_2.

L'intervalle sur lequel est testé la présence d'une rupture est $[a, t]$. Si elle est détectée, la rupture sera située en $t - N$. En effet, l'algorithme est :

1. initialiser l'intervalle de test : $[a, t]$
2. si $t - a \geq 2N$, alors
   (a) estimer (μ_0, σ_0) sur l'intervalle $[a : t]$
   (b) estimer (μ_1, σ_1) sur l'intervalle $[a : t - N]$
   (c) estimer (μ_2, σ_2) sur l'intervalle $[t - N : t]$

[34] voir section 7.3

(d) si $\Delta BIC(t) < 0$ alors il y a détection et $a = t - N$

3. t=t+1, retourner en (2)

On peut remarquer qu'il existe un laps de temps de taille N incompressible entre la date à laquelle intervient un changement et sa détection. On peut noter aussi qu'après une alarme, il existe une période de durée N où il est impossible de détecter un changement. Ces dernières constatations laissent présager que cet algorithme de surveillance apporterait peu dans le cas où l'environnement varie rapidement. Comme l'algorithme de Shewhart, l'algorithme de surveillance utilisant le critère BIC nécessite l'utilisation d'une fenêtre glissante afin de calculer la moyenne et la variance d'une suite (courte) de valeurs de δ. Ici encore, le choix doit être fait entre :

- une fenêtre longue permettant une bonne évaluation de ces paramètres
- une fenêtre courte permettant de minimiser le retard entre le changement de comportement de la variable δ et le déclenchement de l'alarme (et donc la réinitialisation des paramètres de la fonction de compensation).

Afin de vérifier que ce critère est peu sensible aux variations du RSB, nous avons fait varier le facteur de pondération λ pour différents RSB de la tâche *échelon* sur Aurora3 (la largeur de la fenêtre d'analyse étant de $N = 20$). La figure 7.9 représente l'amélioration de la reconnaissance en mots par rapport à *Référence*. Les résultats sont données en fonction du RSB sur la partie bruitée et en fonction du niveau de seuil utilisé.

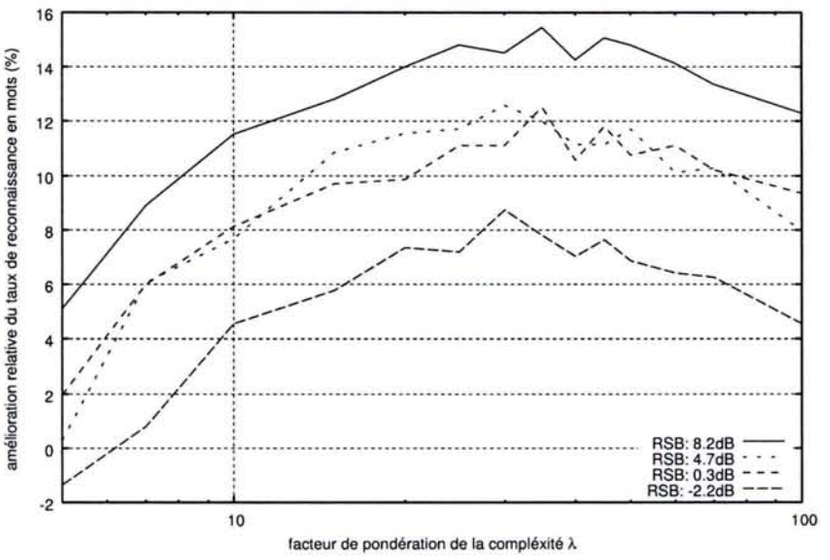


FIG. 7.9 – Amélioration du taux de reconnaissance en mots sur Aurora3 *échelon* par rapport à *Référence* pour *Biais* utilisant le BIC comme mécanisme de réinitialisation (RSB relevé sur la partie bruitée).

On peut voir que les maximums d'amélioration sont obtenus pour des valeurs de λ proches ($\lambda = 30$), quelque soit le RSB sur la partie bruitée. Cette simple expérience permet de mettre en évidence que le critère BIC permet un repérage de changement de régime de la variable δ. De plus, on peut voir que la valeur du seuil λ dépend peu de la variation dans le RSB.

Le critère BIC utilisé dans l'algorithme de surveillance de la variable δ permet donc d'améliorer le taux de reconnaissance d'un SRAP utilisé dans un environnement présentant une variation

abrupte inopinée. Comme l'algorithme de Shewhart, ce processus de surveillance est tributaire de la détermination d'un seuil et surtout du calcul des paramètres de modèles gaussiens sur une courte fenêtre.

La section suivante expose une tentative pour s'affranchir du calcul d'une moyenne et d'une variance sur un petit ensemble. L'objectif est la réduction de la taille de la fenêtre d'analyse et donc la minimisation du retard entre un changement et sa détection.

7.4.2 Fonction de variation spectrale

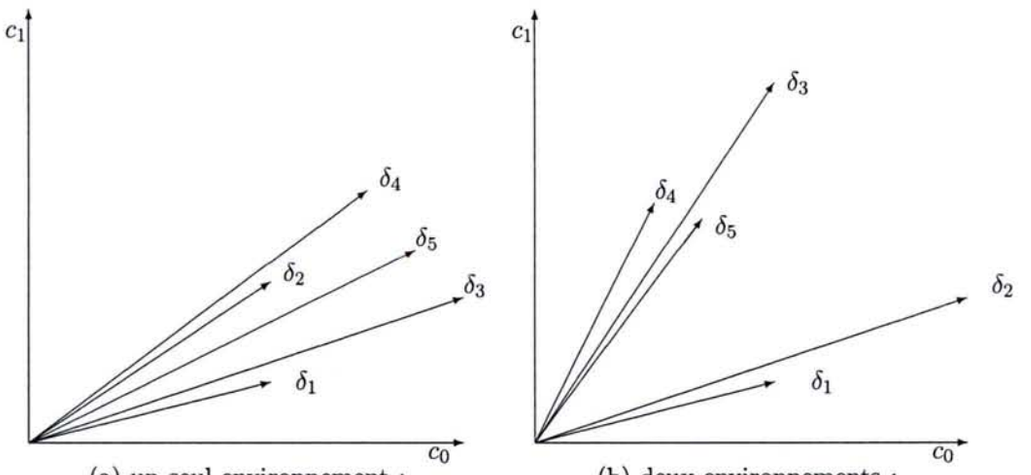
La fonction de variation spectrale (SVF, *Spectral Variation Function*) a été proposée dans [Brugnara *et al.*, 1992] pour exprimer le taux de variation acoustique du signal à un instant donné. Cette approche développée dans [Brugnara *et al.*, 1992] et reprise dans [Colotte, 2002] dans le domaine de la segmentation de la voix compare deux ensembles de vecteurs et donne une mesure de similarité sans faire intervenir de modélisation. Cette méthode se retrouve dans différents champs de recherche en parole, comme la segmentation en locuteurs, la recherche des silences...

La figure 7.10 permet d'illustrer l'idée sous-jacente à l'algorithme de surveillance que nous proposons. Elle montre (en simulation) les projections de δ sur les deux premières dimensions cepstrales. Deux cas sont envisagés :

- en (a), l'environnement acoustique est quasi-stationnaire pour toute la durée de la séquence des δ. Donc, comme nous l'avons vu à la section 7.1.3 de ce chapitre, les projections de δ sur chaque dimension cepstrale adoptent une distribution gaussienne. Cela se traduit, sur la figure, par une concentration des projections de δ dans une région (inconnue) du plan (c_0, c_1).
- en (b), l'environnement acoustique subit un changement brutal et bascule de env_0 à env_1. Cela se traduit par une rupture dans le comportement de la variable δ. Dans ce cas, comme proposé dans la section 7.1.3 de ce chapitre, la séquence des δ va adopter successivement deux distributions (de type gaussien). Sur la figure, les vecteurs δ_1 et δ_2, générés sous env_0 ne suivent pas la même distribution que δ_3, δ_4 et δ_5, générés sous env_1. Leurs projections sur (c_0, c_1) sont donc localisées dans deux régions du plan *a priori* distinctes.

La figure 7.10 met donc en avant le fait que les vecteurs δ générés sur deux environnements distincts occuperont deux régions (distinctes) de l'espace. Par conséquent, prouver qu'une rupture est intervenu dans le comportement de δ peut se résumer à prouver que la séquence peut se diviser en deux, les vecteurs de chaque séquence se séparant en deux régions distinctes de l'espace.

Les méthodes pour parvenir à la détermination de ces régions sont nombreuses : on peut par exemple penser à une segmentation VQ. Nous avons choisit d'utiliser une approche similaire à SVF (nous conserverons cette dénomination par clarté). La méthode proposée repose sur l'observation, à chaque instant t de la région de l'espace occupée par les δ. Si cette région se déplace, alors, l'algorithme de surveillance conclura à un changement dans l'environnement. Ce "déplacement" de la région est mesuré simplement par l'angle moyen observé entre vecteurs δ successif. En effet, sur la figure 7.10-(b), par exemple, les angles formés par les vecteurs issus de l'environnement env_0 et ceux de env_1 sont supérieurs à ceux que l'on peut observer entre les vecteurs issus de env_0 ou de env_1. Donc, afin de mesurer la dissemblance entre les vecteurs δ_1 et δ_2 d'une part et δ_3, δ_4 et δ_5 d'autre part, la moyenne des angles est calculée. Un angle moyen supérieur à une valeur de seuil permet de rendre compte que ces vecteurs peuvent se diviser en deux séquences qui ne suivent pas le même comportement.



(a) un seul environnement : les projections de δ sont localisées dans une même région.

(b) deux environnements : les projections de δ se séparent en deux régions

FIG. 7.10 – Disposition des projections de δ sur les deux premières dimensions cepstrales avec changement (b) et sans changement (a) dans la distribution de δ.

Application à la surveillance de la variable δ

On remarque que cette méthode si elle utilise la nature gaussienne de la distribution de δ, ne cherche pas à en calculer les paramètres. De plus, même si le calcul de dissemblance fait intervenir la mesure d'un angle moyen sur une fenêtre glissante, on ne calcule ni l'écart-type ni la moyenne des δ sur cette fenêtre. On s'affranchit donc des difficultés rencontrées par les algorithmes de Shewhart et BIC qui nécessitaient une estimation (peu précise) de ces paramètres.

Dans l'implémentation choisie, on considère à l'instant t deux suites de vecteurs consécutifs :

$$\Delta_g = \{\delta_{t-2N+1}, \ldots, \delta_{t-N}\}$$

et

$$\Delta_d = \{\delta_{t-N+1}, \ldots, \delta_t\}$$

Le but est ici de décider s'il existe en $t - N$ un changement de distribution dans la séquence totale des

$$\delta = \Delta = \{\delta_{t-2N+1}, \ldots, \delta_t\}$$

Ces suites constituent un contexte droit (Δ_d) et gauche (Δ_g) de l'hypothétique instant de changement dans l'environnement.

On effectue une comparaison entre les vecteurs issus du premier contexte et ceux issus du deuxième. Cette comparaison se fait par l'intermédiaire du produit scalaire normalisé. Pour mesurer la dissemblance entre chacun des vecteurs d'un contexte avec ceux du contexte opposé, il suffit de mesurer leur cosinus :

$$cos(\delta_i, \delta_j) = \frac{< \delta_i, \delta_j >}{|\delta_i||\delta_j|}$$

avec $t - 2N + 1 \leq i \leq t - N$ et $t - N + 1 \leq j \leq t$.

Ainsi, une valeur proche de 1 devrait témoigner que les deux vecteurs sont très semblables. Inversement, une valeur proche de -1 indiquerait que les deux vecteurs ont des caractéristiques éloignées.

La moyenne des cosinus d'un vecteur δ_i de Δ_d avec tout les autres vecteurs du contexte opposé est $M(\delta_i)$:

$$M(\delta_i) = \frac{\sum_{j=t-N+1}^{t} cos(\delta_i, \delta_j)}{N}$$

La moyenne de toutes les $M(\delta_i)$ avec δ_i de Δ_d donne :

$$\frac{\sum_{i=t-2N+1}^{t-N} \sum_{j=t-N+1}^{t} cos(\delta_i, \delta_j)}{N^2}$$

dont les valeurs sont comprises entre -1 et 1.

On définit alors la fonction SVF à l'instant t comme étant :

$$SVF(t) = \frac{1}{2}(1 - \frac{\sum_{i=t-2N+1}^{t-N} \sum_{j=t-N+1}^{t} cos(\delta_i, \delta_j)}{N^2})$$

Ainsi, la fonction SVF donne des valeurs proches de 0 lorsque les vecteurs δ sont issus d'un même environnement stable. Les valeurs sont proche de 1 lorsqu'il existe une transition. Cette approche fait donc encore intervenir un seuil : à chaque instant t, la valeur de $SVF(t)$ va être comparer à un seuil λ. La fonction de décision, à l'instant t est alors :

$$decision = \begin{cases} 0 \quad \text{(pas d'alarme) si} \quad SVF(t) < \lambda; \\ \\ 1 \quad \text{(alarme) si} \quad SVF(t) \geq \lambda \end{cases}$$

Comme pour les autres mécanismes de surveillance exposés (Shewhart et BIC), une alarme déclenche une réinitialisation du processus de compensation *Biais*. Nous avons fait le choix de faire une réinitialisation par remise à zéro de l'historique.

Dans ce nouvel algorithme de surveillance, toutes les dimensions des δ sont utilisées pour élaborer une fonction de décision qui implique une réinitialisation sur toutes les dimensions. Par conséquent, on ne pose aucune condition ***a priori*** quant à la nature de la différence qui existe entre les contextes Δ_d et Δ_g. Cette différence peut se situer par exemple sur les dimensions hautes et/ou les dimensions basses.

Considérons deux environnements consécutifs env_0 et env_1 se succédant au cours de la reconnaissance d'une même phrase. Ces environnements sont de nature différente mais cette différence n'est quantifiable que par l'intermédiaire de leur RSB respectif. L'expérience suivante permet de mettre en évidence la relation qui existe entre la différence de RSB et le seuil λ à employer pour détecter ce changement. Nous avons choisi d'effectuer des tests de reconnaissance sur la base Aurora bruitée par *échelon* car l'épreuve *échelon* propose une rupture simple dans chaque phrase de test. Cette épreuve est donc idéale pour mettre en évidence le rôle du paramètre λ dans le processus de surveillance.

La figure 7.11 représente l'amélioration du taux de reconnaissance obtenue par cette approche, par rapport à *Référence*, sur cette base de test. Pour chaque RSB, la courbe d'amélioration du taux de reconnaissance passe par un maximum (particulièrement visible pour $RSB = -3.8dB$).

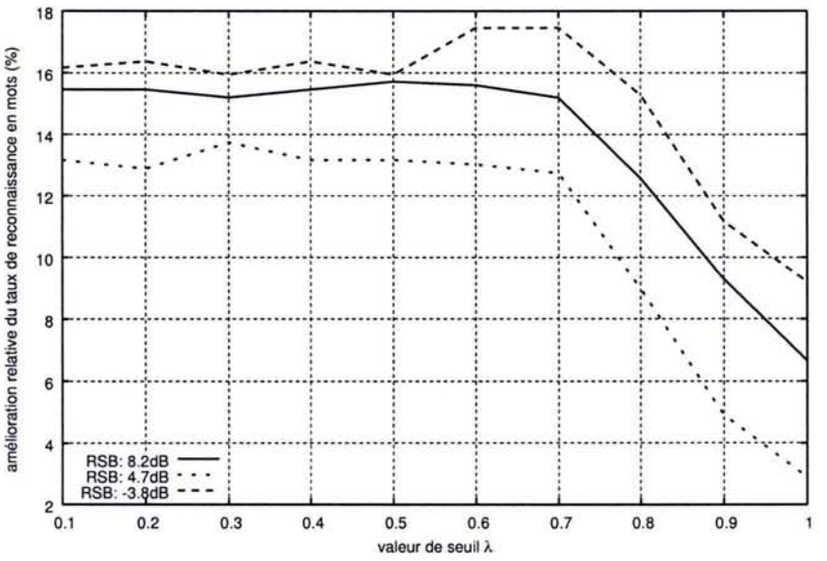


FIG. 7.11 – Amélioration du taux de reconnaissance en mots sur Aurora3 *échelon* par rapport à *Référence* pour *Biais* utilisant l'approche SVF comme mécanisme de réinitialisation (RSB relevé sur la partie bruitée).

On peut remarquer sur cette figure que le seuil optimal λ est le même ($\lambda = 0.6$), quelque soit la différence de RSB entre env_0 et env_1. Cette constatation nous permet de dire que l'approche SVF semble robuste, puisqu'une seule valeur de λ permet d'obtenir une bonne amélioration du taux de reconnaissance quelque soit la différence de niveau de bruit de part et d'autre de la fracture.

La section suivante propose une comparaison des performances des trois processus de surveillance que nous avons proposés.

7.5 Comparaison des approches

Le but que nous nous sommes fixé en intégrant un processus de surveillance dans notre algorithme de compensation était d'augmenter le taux de reconnaissance de notre SRAP dans un environnement acoustique variant brusquement. Dans cette section, nous nous attacherons donc à mesurer et comparer les taux de reconnaissance obtenus par des SRAP intégrant les trois processus de surveillance proposés (Shewhart, BIC et SVF). Dans une seconde partie, nous comparerons ces méthodes en terme de taux de bonnes réponses et de fausses alarmes, c'est-à-dire que nous évaluerons l'habilité de chaque approche à repérer les transitions dans un environnement variant inopinément.

7.5.1 Taux de reconnaissance

L'épreuve *échelon*

La figure 7.12 donne une vue de l'amélioration du taux de reconnaissance apportée par les trois méthodes proposées. La figure 7.13 donne les réductions des taux d'erreur en mots (WER) correspondantes. Nous avons choisi comme tâche de reconnaissance l'épreuve *échelon* sur les bases VODIS et Aurora[35]. Cet exercice simple permet de mettre en évidence les capacités de chaque algorithme de surveillance à détecter la variation soudaine dans l'environnement acoustique. La base VODIS nous servira pour élaborer nos commentaires et la base Aurora, pour les valider.

Sur la figure 7.12 sont rapportées les courbes relatives à l'amélioration du taux de reconnaissance par rapport à *Référence* (le SRAP n'intégre aucun mécanisme de robustesse). Ces courbes sont tracées en fonction du RSB moyen sur la partie bruitée. Les 4 courbes représentées illustrent l'amélioration relative apporté par la méthode *Biais*

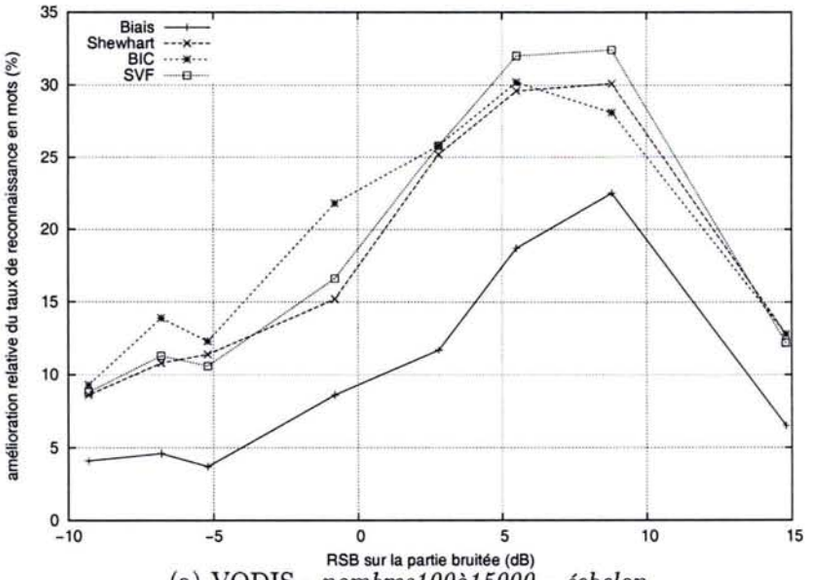
- sans mécanisme de surveillance-réinitialisation (décrite au chapitre 5)
- avec l'algorithme de *Shewhart* pour mécanisme de surveillance-réinitialisation (ce chapitre, section7.3)
- avec l'algorithme utilisant le critère *BIC* pour mécanisme de surveillance-réinitialisation (ce chapitre, section7.4.1)
- avec l'algorithme utilisant l'approche de type *SVF* pour mécanisme de surveillance-réinitialisation (ce chapitre, section7.4.2)

Les deux courbes de la figure 7.12 ont pour point commun leur forme générale :

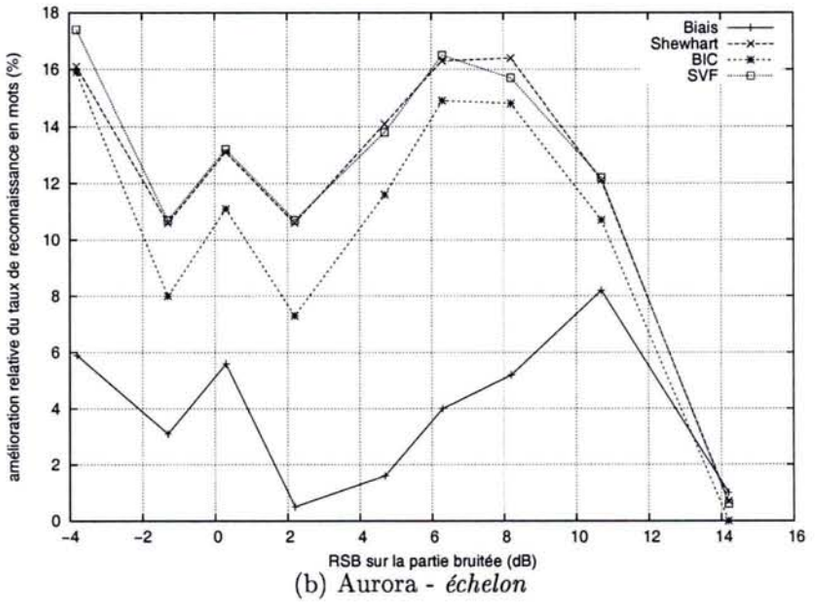
1. pour un RSB fort (les phrases sont peu bruitées), l'amélioration relative est faible. Ceci est dû à deux facteurs. D'une part, la variation dans l'environnement n'est pas bien détectée, voir ignorée. D'autre part, la différence entre les environnements de chaque côté de la rupture sont très semblables pour ces valeurs de RSB. Dans les deux cas, le mécanisme de suivi n'apporte rien et l'amélioration relative apportée par les méthodes de compensation intégrant un mécanisme de surveillance est comparable à celle apportée par la méthode *Biais* seule.
2. pour un RSB faible, l'amélioration relative est aussi faible. Dans ce cas, ce ne sont pas des erreurs de détection qui sont en cause, mais le fait que le mécanisme de compensation ne converge pas toujours vers la bonne valeur de biais après la réinitialisation déclenchée par la détection d'une variation dans l'espace acoustique. Cependant comme l'amélioration relative n'est pas nulle, on peut dire que notre algorithme de compensation combiné avec un mécanisme de surveillance rend le SRAP plus robuste à un environnement variant brusquement, même à des valeurs de RSB très basses.
3. Pour un RSB moyen sur les parties bruitées, l'amélioration relative de notre ensemble de méthodes combinant un processus de surveillance avec notre processus de compensation est très visible. Dans cette zone, les trois méthodes présentées présentent un comportement comparable. On peut dire que, pour ces valeurs de RSB,
   - les trois méthodes de surveillance permettent une bonne détection de la rupture
   - notre méthode de compensation, après avoir été réinitialisée suite à la détection d'une rupture, permet de converger vers un biais efficace.

   Cette observation permet de conclure de l'efficacité de notre approche, pour cette tâche spécifique.

[35] comme décrit en section 7.1.1

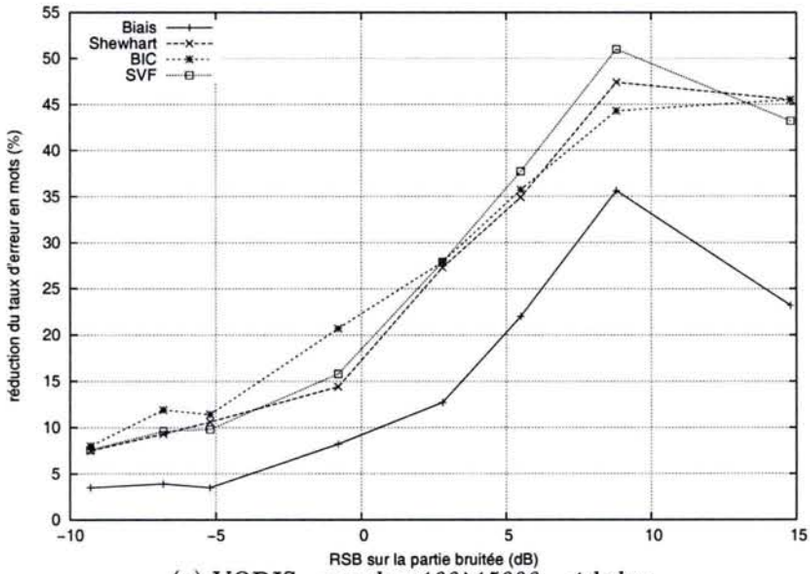


(a) VODIS - *nombres100à15000* - *échelon*

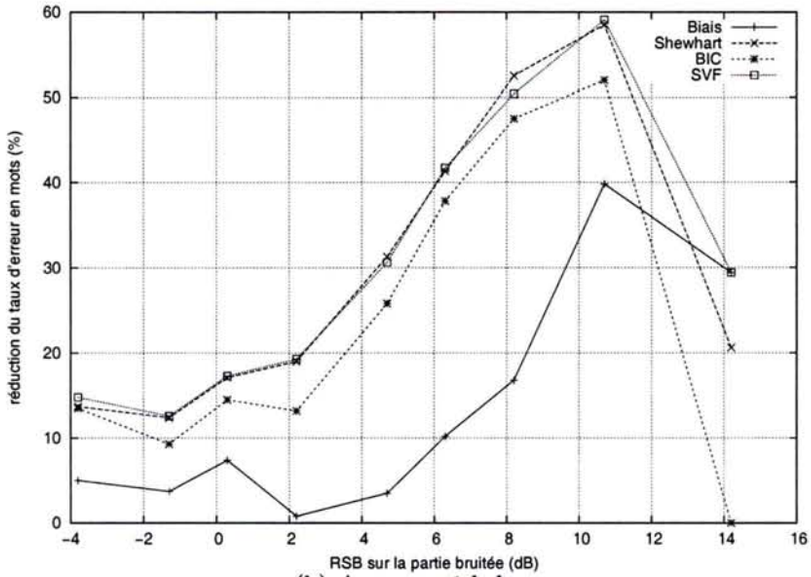


(b) Aurora - *échelon*

FIG. 7.12 – Amélioration du taux de reconnaissance pour l'épreuve *échelon*.



(a) VODIS - *nombres100à15000* - *échelon*



(b) Aurora - *échelon*

FIG. 7.13 – Réduction du taux d'erreur en mots pour l'épreuve *échelon*.

Il existe quelques dissemblances entre les deux courbes de la figure 7.12. On remarquera en particulier que les valeurs de l'amélioration relative pour les expériences sur VODIS et Aurora ne sont pas comparables. Cet écart peut s'expliquer par les différences de taux de reconnaissance que l'on observe entre les performances de *Référence* sur les deux bases de test *échelon* d'Aurora et VODIS (tableaux 7.1 et 7.2). La différence observée vient, entre autres, de la qualité des modèles acoustiques employés.

L'épreuve *aléatoire*

Nous avons procédé aux mêmes expérimentations que précédemment, mais cette fois sur les bases de test corrompues par la tâche *aléatoire*. La figure 7.14 présente les améliorations relatives du taux de reconnaissance sur ces deux bases, lorsque le mécanisme de compensation employé est notre approche *Biais*. La figure 7.15 donne les réductions des taux d'erreur en mots (WER) correspondantes. Là encore, le processus de compensation n'inclut pas de processus de surveillance-réinitialisation (courbe *Biais* sur la figure), ou utilise l'un des trois algorithmes présenté dans ce chapitre (*Shewhart*, *BIC* et *SVF* sur la figure).

Les courbes de la figure 7.14 présentent le même aspect général observé sur les courbes de l'expérience *échelon*. En effet, les taux d'amélioration relative sont faibles pour les valeurs extrêmes de RSB et fortes pour les valeurs de RSB intermédiaires.

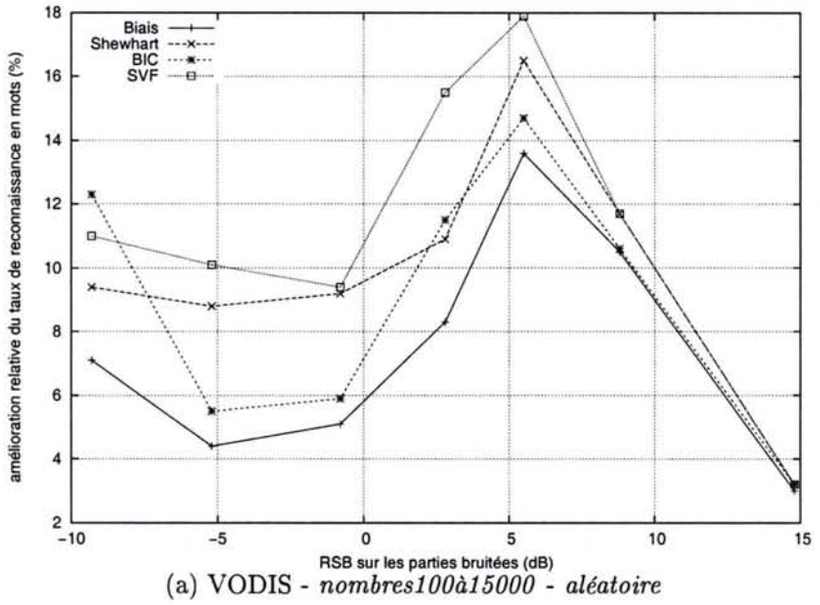
On observe toujours que l'amélioration relative des méthodes de compensation intégrant un processus de surveillance-réinitialisation est toujours plus importante que celle apporté par *Biais* seul. Ceci confirme la validité de notre approche.

On remarque de plus que la valeur maximale de l'amélioration relative est plus faible dans le cas de la tâche *aléatoire* que de la tâche *échelon*. En effet, la tâche *aléatoire* présente une difficulté supplémentaire par rapport à *échelon*. Pour la tâche *échelon*, chaque phrase présente une seule variation abrupte dans son environnement acoustique. Dans le cas de la tâche *aléatoire*, cette variation est présente 2 à 3 fois par phrase et les phases de stabilité de l'environnement acoustique sont parfois courtes (la taille minimale de cette phase étant d'une demi seconde). Par conséquent, l'intervalle de temps sur lequel le processus de compensation a la possibilité de converger vers une valeur de biais correspondant effectivement à l'environnement acoustique est souvent réduit. Malgré cette difficulté supplémentaire notre approche est efficace pour cette tâche et pour la plupart des RSB puisque l'amélioration relative du taux de reconnaissance est le plus souvent positive.

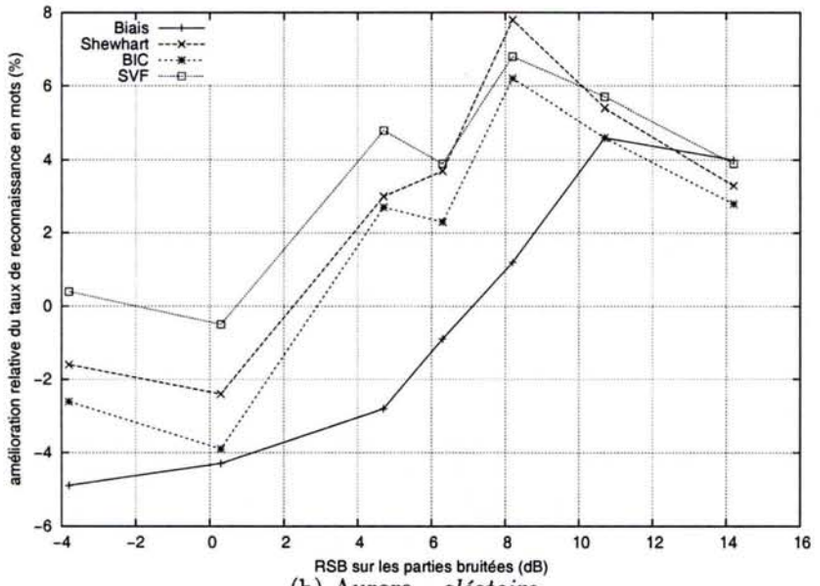
Cependant, le taux d'amélioration pour la tâche *aléatoire* de la base *Aurora* est négatif pour toutes les méthodes de compensation présentées, pour un RSB inférieur à $2dB$ sur la partie bruitée. Ce taux est inférieur à 8% pour les autres valeurs de RSB . Au vu de la différence observée entre les tâches Aurora-*échelon* et Aurora-*aléatoire*, nous pouvons conclure que c'est la brièveté des plages sur lesquelles l'environnement est stable qui est à l'origine de la dégradation des performances de nos algorithmes pour la tâche Aurora-*aléatoire*.

Comparaison des approches

Pour toutes les tâches étudiées, les performances obtenues par notre algorithme de compensation sont comparables, quelle que soit la méthode de surveillance-réinitialisation utilisée. On peut cependant remarquer que la méthode utilisant l'approche SVF donne des résultats supérieurs aux autres méthodes pour l'épreuve *aléatoire* (en particulier). Nous expliquons ce comportement par

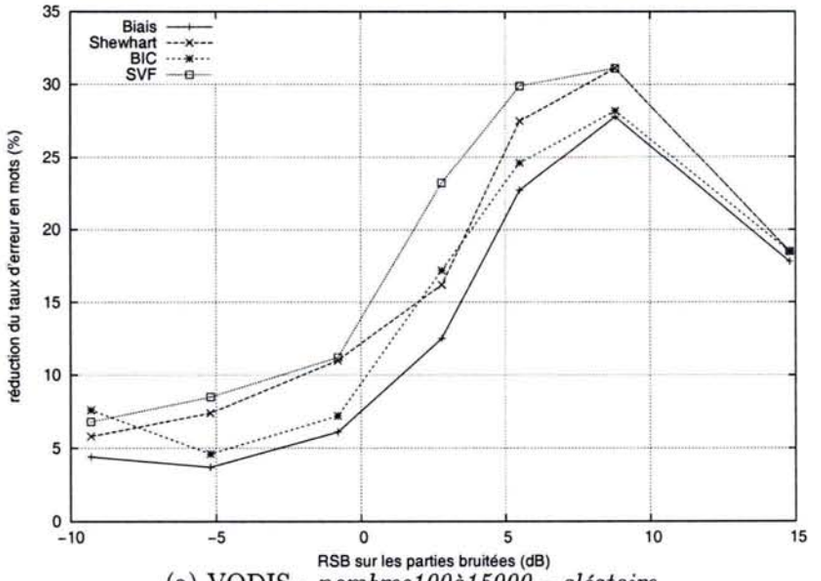


(a) VODIS - *nombres100à15000 - aléatoire*

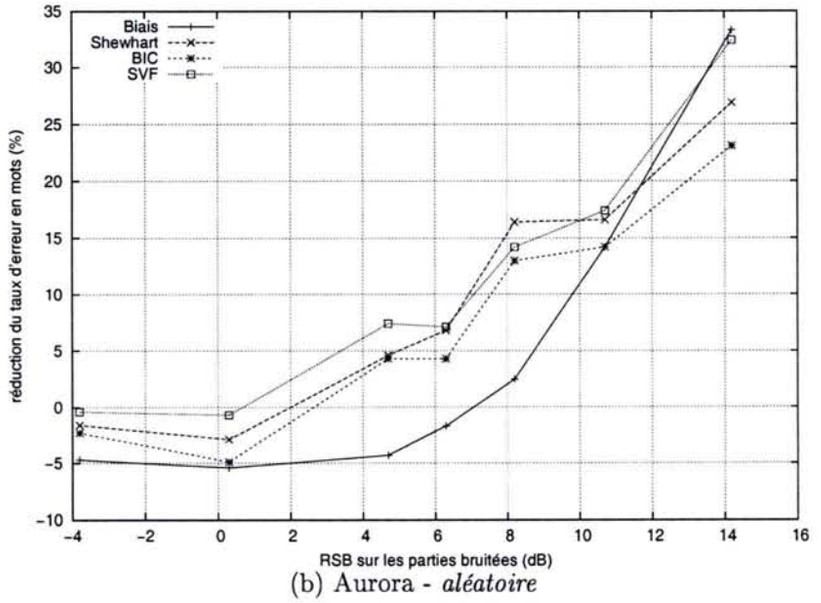


(b) Aurora - *aléatoire*

FIG. 7.14 – Amélioration du taux de reconnaissance pour l'épreuve *aléatoire*.



(a) VODIS - *nombres100à15000 - aléatoire*



(b) Aurora - *aléatoire*

FIG. 7.15 – Réduction du taux d'erreur en mot pour l'épreuve *aléatoire*.

le fait que l'approche SVF est la seule qui ne fasse pas intervenir d'estimation des paramètres de la distribution de δ. Cependant, le calcul des moyennes et des variances d'une distribution gaussienne sur une petite fenêtre, bien qu'étant simple, n'est pas précis et perturbe la détection pour les méthodes utilisant le critère BIC ou l'algorithme de Shewhart.

7.5.2 Précision de la détection

Nous avons vu dans la section 7.3.2 de ce chapitre que le processus de surveillance de Shewhart pouvait déclencher une alarme alors qu'aucun changement n'est intervenu dans l'environnement acoustique. En effet, les indices permettant de repérer un changement dans le contexte acoustique ne sont pas précis et peuvent aussi rendre compte d'un changement de mode d'élocution, par exemple. On distingue donc deux types d'alarmes :

- les bonnes réponses (*GA*, *Good Answer*) sont des alarmes correspondant effectivement à des changements dans l'environnement acoustique.
- les fausses alarmes (*FA*, *False Alarm*) sont des alarmes déclenchées par d'autres événements et qui ne correspondent pas à un changement dans l'environnement acoustique.

Toutes les méthodes de surveillance que nous avons présentées utilisent le même indice, c'est à dire la variable δ pour repérer les changements dans l'environnement acoustique. Par conséquent, toutes ces méthodes doivent déclencher des alarmes (et donc une réinitialisation) alors qu'aucun changement n'est intervenu.

Lorsqu'il est déclenché par une *GA*, le processus de réinitialisation permet à l'algorithme de compensation de ne prendre en compte que les informations fournies par le nouvel environnement pour estimer la fonction de compensation et ainsi d'améliorer le taux de reconnaissance. Dans le cas où elle est déclenchée par une *FA*, la réinitialisation peut entraîner une dégradation des performances de l'algorithme de compensation. Lors d'une réinitialisation sur une *FA*, les informations sur l'historique de l'environnement acoustique sont en effet perdues et les paramètres de la fonction de compensation passent par une nouvelle phase d'oscillation avant de converger à nouveau. Cette phase d'oscillation, si elle est souvent sans conséquence[36] peut réduire voir annuler l'amélioration apportée par le processus de compensation.

Nos expériences ont pu mettre en évidence ce comportement : chaque processus de surveillance proposé dépend d'un paramètre équivalent à un seuil réglant sa sensibilité aux variations dans la variable δ. Nous avons pu observer que pour toutes ces méthodes le taux de reconnaissance dans un environnement variant brusquement est une fonction du seuil. Le comportement du taux de reconnaissance est le même pour toutes ces méthodes : il augmente lorsque le seuil permet de repérer les changements, puis il diminue lorsque les alarmes déclenchées sont trop nombreuses.

La figure 7.16 représente la répartition entre *GA* et *FA* pour des alarmes déclenchées par les trois processus de surveillances.

- en (a), la reconnaissance se fait sur les *nombres100à15000* de la base VODIS corrompus par *échelon*.
- en (b), ces phrases sont corrompues par *aléatoire*.

Les maximums des taux de reconnaissance (en mots) pour les deux exercices ont été rapportés sur les courbes. Les valeurs représentées en abscisses et ordonnées sont les proportions des GA et des FA par rapport au nombre total de changements effectifs.

On utilise ici la version de l'algorithme de Shewhart où la réinitialisation du biais se fait sur toutes les dimensions cepstrales au même instant. Les trois processus de surveillance utilisent une

[36] voir les réflexions sur ce sujet au chapitre 5, section 5.3.3

fenêtre d'analyse de 20 cepstres. Les courbes sont obtenues en faisant varier les seuils respectifs des trois méthodes. Une alarme est comptabilisée comme une *GA* lorsqu'elle intervient dans un intervalle de 25 cepstres suivant le changement effectif dans l'espace acoustique.

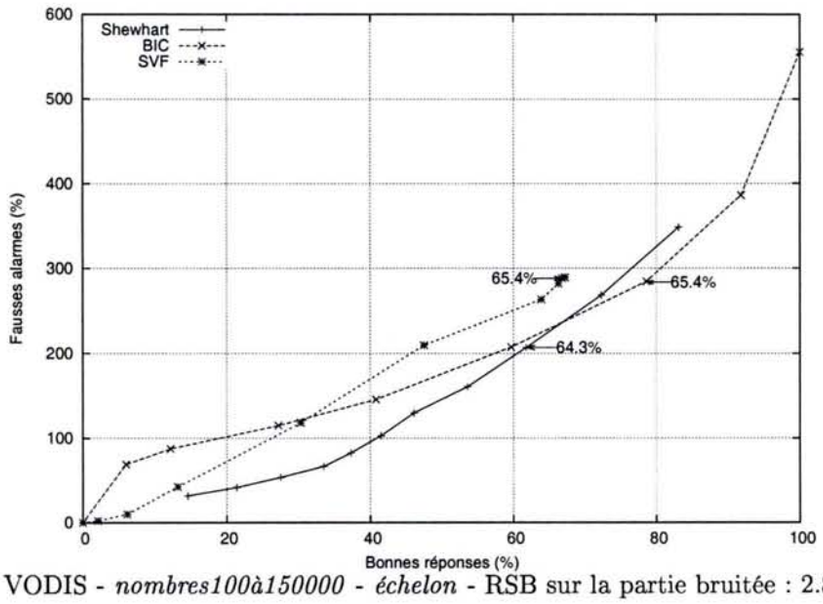
Pour les trois processus de surveillance, la courbe (FA/GA) est croissante. Pour les deux tâches, la disposition relative des courbes et leur forme générale sont identiques. On peut voir que le processus de surveillance de Shewhart permet d'obtenir le meilleur rapport GA/FA, pour les valeurs de GA inférieures à 60%. De plus, pour des valeurs de GA supérieures à 60%, les méthodes de type BIC et Shewhart offrent un rapport GA/FA équivalent et l'approche de type SVF donne toujours le plus grand rapport GA/FA. Enfin, pour l'exercice *aléatoire*, les maximums des taux de reconnaissance sont tous atteints pour un rapport GA/FA proche de 70%/120% alors que, pour l'exercice *échelon*, les optimums sont concentrés dans une zone plus étalée, centrée sur 70%/250%.

7.6 Conclusion

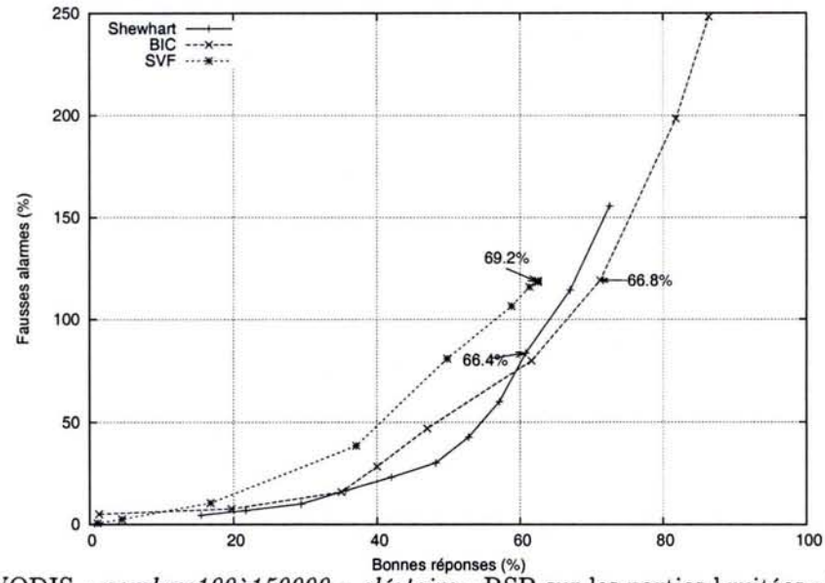
Dans ce chapitre, nous avons présenté une amélioration d'un algorithme de compensation temps réel synchrone à la trame utilisant le cadre de travail de l'association stochastique (*Stochastic Matching*). Nous avons intégré un système de détection de changement d'environnement ne se basant que sur l'étude d'une variable dérivée du processus de compensation. Par conséquent cette amélioration de notre algorithme de compensation ne nécessite aucune information *a priori* sur l'environnement d'utilisation. Ainsi la reconnaissance est améliorée dans un milieu pouvant varier brusquement et inopinément. Nous avons évalué notre algorithme sur des tâches de reconnaissance de nombres pour des phrases issues des bases VODIS et Aurora artificiellement bruitées.

Trois algorithmes de surveillance ont été testés. Ils sont inspirés de l'algorithme de surveillance de Shewhart, du critère d'information bayésien (BIC) et de l'approche appelée SVF. Ces algorithmes sont chargés de repérer une rupture dans le comportement de la variable δ, un indice relatif à l'environnement acoustique et généré par le processus de compensation. Une rupture peut être considérée comme un changement d'environnement et déclenche une réinitialisation du processus de compensation.

L'algorithme de compensation que nous présentons, fourni une amélioration significative du taux de reconnaissance dans des milieux adverses changeant inopinément quelque soit le processus de surveillance utilisé. Le processus de surveillance utilisant l'approche SVF, en particulier, est très bien adapté à la reconnaissance dans un milieu variant fréquemment et abruptement.



(a) VODIS - *nombres100à150000* - *échelon* - RSB sur la partie bruitée : 2.8dB



(b) VODIS - *nombres100à150000* - *aléatoire* - RSB sur les parties bruitées : 2.8dB

FIG. 7.16 – Taux de bonnes détections et de fausses alarmes lors de la reconnaissance (en mots) des tâches *nombres100à150000* de la base VODIS corrompues par *échelon* et *aléatoire* pour les processus de réinitialisation *Shewhart*, *BIC* et *SVF*.

Conclusion générale

Les Systèmes de Reconnaissance Automatique de la Parole (SRAPs) voient leurs performances diminuer de manière significative lorsque les environnements dans lesquels ils ont été entraînés diffèrent de ceux dans lesquels ils sont utilisés. Les travaux de recherche exposés dans ce document portent sur des méthodes permettant de rendre les SRAPs robustes à leur environnement d'utilisation.

Nous avons vu au **chapitre 2** que plusieurs techniques ont été proposées afin d'améliorer la reconnaissance de la parole dans le bruit. Les techniques de *compensation* cherchent à transformer les séquences de parole bruitée en des séquences de parole non-corrompue. Pour parvenir à cet objectif, la plupart d'entre elles utilise une mise en correspondance entre la séquence de parole bruitée et les données observées lors de l'apprentissage. Cette mise en correspondance peut se faire de différentes manières, et peut faire intervenir des données stéréo, une modélisation de la parole propre et/ou une modélisation de l'interaction de l'environnement acoustique avec le signal de parole.

On regroupe sous le terme *Stochastic Matching* un ensemble de méthodes de compensation qui tire parti du mécanisme de reconnaissance pour obtenir cette mise en correspondance. Nous avons vu au **chapitre 3** une approche de type *Maximum Likelihood* du *Stochastic Matching* qui a été proposée par Sankar et Lee en 1996. Les paramètres d'une fonction de compensation y sont estimés afin de maximiser la vraisemblance d'une séquence de parole en fonction de la séquence optimale des modèles acoustiques fournie par le processus de reconnaissance. Depuis, des méthodes utilisant des approches alternatives au *Stochastic Matching* ont été présentées. Sous certaines conditions, ces méthodes peuvent être unifiées, bien qu'elles aient été développées à partir d'hypothèses distinctes. L'aspect le plus intéressant du *Stochastic Matching* est qu'il ne nécessite aucune information sur la nature ou le niveau de bruit ambiant. Il est possible en effet d'effectuer le débruitage d'une phrase de test sans disposer d'autres informations *a priori*.

Les recherches exposées dans ce document utilisent le cadre théorique du *Stochastic Matching*. Au **chapitre 5**, nous en proposons une approche en temps-réel. Les algorithmes temps-réel, ou synchrones à la trame, sont particulièrement intéressants lorsqu'il s'agit de contrer l'influence d'une source de bruit variant lentement. Dans ces conditions, la compensation s'effectue en parallèle avec le processus de reconnaissance. Le processus de compensation ainsi obtenu augmente la robustesse d'un SRAP utilisé dans un environnement non-stationnaire variant lentement.

Un algorithme de compensation utilisant le cadre formel du *Stochastic Matching* tout en étant synchrone à la trame soulève un problème ardu. Dans ce cas en particulier, la séquence optimale d'états nécessaire dans la méthode *Stochastic Matching* exposée plus haut n'est pas disponible au moment de compenser une trame de parole. Afin de nous départir de cet écueil, nous avons envisagé d'approximer ces statistiques manquantes par les seules disponibles lors de la compensation : les probabilités *forward*. Les paramètres d'une fonction de compensation

simple peuvent être réactualisés itérativement à chaque trame en fonction de cette probabilité. Ces paramètres étant obtenus de façon incrémentale, leur estimation sera d'autant plus fine que le nombre d'observations émises consécutivement dans un même environnement acoustique sera important.

Les algorithmes que nous avons développés dans ce cadre donnent des résultats supérieurs à ceux obtenus par les algorithmes classiques de compensation synchrone à la trame (comme la normalisation cepstrale ou la soustraction spectrale). Par exemple, la première version de notre algorithme a obtenu un taux de reconnaissance en mots supérieur de 15.5% à la soustraction spectrale sur les données de la base VODIS. Sur cette même base, on obtient un taux de reconnaissance en mots supérieur de 27.8% par rapport à la normalisation cepstrale.

Afin d'améliorer les résultats obtenus par cette méthode, nous avons proposé au **chapitre 6** une fonction de compensation adoptant la forme d'un arbre binaire de transformations. Cette approche est motivée par plusieurs constatations. Tout d'abord, il est reconnu que des vecteurs de parole similaires sont affectés de la même façon s' ils sont émis dans le même environnement acoustique. Par conséquent, un ensemble de fonctions de compensation spécifiques à des sous-ensembles de l'espace d'observation doit donner de meilleurs résultats qu'une seule et unique fonction de compensation couvrant l'ensemble des vecteurs de parole. Par exemple, dans un même environnement acoustique, deux voyelles seront affectées d'une certaine façon alors qu'une consonne le sera d'une autre. Cependant une fonction de compensation semblable à celle que nous avons proposée, mais spécifique à un sous-ensemble acoustique, rencontre un problème majeur : si la phrase à compenser comporte très peu d'observations dans un de ces sous-ensembles, la transformation associée à ce sous-ensemble sera mal estimée. Pour contourner ce problème, nous organisons l'ensemble des fonctions spécifiques aux sous-ensembles de façon hiérarchique. Ainsi, lorsqu'une observation doit être compensée, on pourra utiliser la transformation associée au plus petit sous-ensemble acoustique contenant cette observation (un nœud de l'arbre). Si cette dernière n'est pas bien estimée on utilisera la transformation associée à l'ensemble acoustique qui le contient (le noeud père).

Au **chapitre 7**, nous avons cherche à améliorer l'efficacité de notre algorithme dans un environnement variant abruptement. En effet lors d'une utilisation réelle un SRAP, l'environnement d'exécution peut subir des variations rapides et inopinées, comme l'ouverture intempestive d'une fenêtre de l'habitacle d'une voiture. Dans ce cas, il est évident qu'on ne dispose d'aucune information sur la nature du bruit ni le moment de son apparition. Un algorithme de compensation doit pourtant identifier et prendre en considération ces changements dans un petit intervalle de temps. Dans cette optique, deux problèmes ont été explorés : celui de la détection des changements dans l'environnement acoustique et celui de la stratégie de compensation à adopter dans le cas d'un changement. Cette étude nous a permis d'améliorer notre algorithme de base afin de débruiter dans un milieu particulièrement difficile, variant rapidement et fréquemment. A chaque trame de temps, la distance entre le vecteur de parole observé et le modèle acoustique le plus probable au sens de la probabilité *forward* est calculé. Or, lorsqu'un changement abrupte se produit dans l'espace acoustique de test, la valeur de cette distance varie rapidement. Nous détectons cette brusque variation par l'intermédiaire de trois algorithmes de suivi tels que celui de Shewhart, le critère d'information bayésien et une adaptation de la fonction de variation spectrale. Lorsque le début d'un nouvel environnement acoustique est détecté, les paramètres de notre fonction de compensation sont réinitialisés à des valeurs obtenues dans l'environnement déjà observé le plus proche. Cette approche donne des améliorations très significatives sur les méthodes de compensation classiques, pour des tests conduits sur des données artificiellement bruitées (un bruit est ajouté au signal de parole propre à partir du milieu de chaque phrase de test). En effet, nous

avons obtenu pour ce type de données une amélioration de 32.4% en reconnaissance de phonèmes par rapport à un système de reconnaissance n'intégrant pas de système de compensation.

Il faut retenir qu'au cours de ces recherches nous avons étudié un algorithme de compensation *temps-réel* permettant d'augmenter la robustesse d'un SRAP pour une grande variété d'environnements. De plus, en rendant la fonction de compensation de cet algorithme dépendant de l'état, nous l'avons rendu non-linéaire. L'algorithme ainsi obtenu peut donc tenir compte de la particularité des régions acoustiques dont sont issues les observations, c'est-à-dire d'une variabilité spatiale du signal. Enfin, en lui intégrant des processus de surveillance et de réinitialisation, nous avons donné à notre approche la possibilité inédite de fonctionner dans un environnement variant inopinément. L'algorithme ainsi obtenu peut donc tenir compte d'une forte variabilité temporelle du signal.

Les travaux que nous avons exposés dans cette thèse ouvrent des perspectives à court terme. Dans un premier temps, nos recherches devraient explorer dans quelles mesure l'approche que nous avons développée pourrait être implémentée dans un SRAP utilisant le paradigme des réseaux bayésiens dynamiques, dont les HMMs ne sont en fait, qu'un cas particulier. Dans un deuxième temps, ces recherches pourraient s'étendre à l'étude de la compatibilité de notre algorithme avec d'autres modélisations acoustiques, comme les modèles segmentaux par exemple.

On peut également proposer des perspectives à long termes qui peuvent se diviser en deux champs d'investigation.

- Au cours des recherches que nous avons décrites dans cette thèse, nous avons proposé des algorithmes *temps-réel* basés sur *Stochastic Matching* permettant une compensation de vecteurs acoustiques. Comme montré dans [Sankar and Lee, 1996], le *Stochastic Matching* peut être aussi utilisé dans le cadre de l'adaptation des modèles acoustiques à l'environnement. Les méthodes d'adaptation des modèles montrent une efficacité certaine pour l'adaptation au locuteur et à un environnement stationnaire. Cependant les calculs déployés dans ces méthodes sont complexes et s'effectuent en *temps-différé*. Nos recherches vont donc s'orienter vers l'étude de l'application de notre approche à l'adaptation *temps-réel* des paramètres des modèles acoustiques. L'objectif que nous cherchons à atteindre par cette voie est de rendre possible l'adaptation des modèles à des environnements non-stationnaires tout en réduisant la complexité des développements. Cet objectif ne pourra être atteint qu'après la résolution du problème de la quantité de données nécessaires à l'adaptation. Le nombre des paramètres à adapter est en effet bien supérieur aux quelques valeurs de biais que nous estimons dans cette thèse. Cela se traduit par une augmentation du nombre de données à observer pour effectuer l'estimation des fonctions devant transformer les paramètres des modèles. L'adaptation des modèles par notre approche ne sera efficace que si cette quantité de donnée reste réduite aux données contenues dans la phrase à reconnaître.
- Un deuxième champ d'investigation concerne l'approche du *Stochastic Matching* que nous avons adoptée. Il s'agit d'utiliser les moyennes des états les plus probables selon la probabilité *avant* pour estimer la fonction de compensation. A propos de l'adaptation incrémentale des modèles acoustiques, Lee fait remarquer dans [Lee, 1998] que l'estimation (non supervisée) basée sur un résultat de reconnaissance faux peut mener à une mauvaise adaptation des modèles (une *dérive* des modèles). Notre approche utilise aussi une estimation incrémentale non supervisée pour effectuer la compensation. Une mauvaise estimation des paramètres de la fonction pour un vecteur d'observation y influence à la fois l'estimation des paramètres pour le vecteur suivant et les statistiques utilisées pour l'alignement de Viterbi. Notre approche doit donc être soumise au même phénomène de *dérive*. Pour limiter cet

effet, Lee suggère d'utiliser une mesure de confiance permettant d'évaluer si l'information sur laquelle porte l'estimation est sûre. Un axe de recherche futur pourrait s'attacher à développer cette mesure. Cette mesure pourrait utiliser par exemple la valeur des probabilités *avant* associées aux états, dont nous n'avons exploité jusqu'à présent que le classement.

L'obtention d'un SRAP robuste est un objectif qui peut être atteint dans le cas où le cadre d'utilisation du SRAP est bien défini. Toutefois les mécanismes qui permettront aux SRAPs de fonctionner correctement dans tout environnement, sous des conditions défavorables inédites variant souvent n'ont pas encore été développés. La recherche et l'étude de tels mécanismes constituent mes perspectives à long terme.

A

Evaluation

A.1 Taux de reconnaissance

Afin de mesurer l'efficacité avec laquelle un SRAP reconnait la parole, on analyse ses erreurs en dégageant 3 types :

- erreurs d'insertion : ajout d'une unité acoustique qui n'existe pas dans le message d'origine.
- erreurs de délétion : non reconnaissance d'une des unités acoustiques du message comme telle.
- erreurs de substitution : confusion d'une unité acoustique du message d'origine avec une autre.

Les taux de reconnaissance (Acc pour *Accuracy*) sont calculés comme suit :

$$Acc = \frac{(\text{nombre d'unités reconnues} - \text{nombres d'unités insérées}) * 100}{\text{nombre total d'unités du message}}$$

On définit aussi :

- le taux d'insertion :

$$Ins = \frac{\text{nombres d'unités insérées} * 100}{\text{nombre total d'unités du message}}$$

- le taux de délétion :

$$Del = \frac{\text{nombres d'unités omises} * 100}{\text{nombre total d'unités du message}}$$

- le taux de substitution :

$$Sub = \frac{\text{nombres d'unités substituées} * 100}{\text{nombre total d'unités du message}}$$

A.2 Intervalle de confiance

Les taux de reconnaissance fournis dans ce document sont ontenus lors d'expériences sur des corpus de taille relativement importante. Cependant comme ces corpus ont des tailles finies ces valeurs ne peuvent être considérées que comme des estimations des valeurs réelles. Il toutefois est possible de fournir un intervalle de valeur dans lequel ce taux de reconnaissance se trouve.

Soient

- n le nombre de phrase dans le corpus de test.
- Acc le taux de reconnaissance obtenus sur ce corpus.

Alors l'intervalle de confiance est donné par :

$$Acc - 2\sqrt{\frac{Acc/100\ (1 - Acc/100)}{n}} \leq \hat{Acc} \leq Acc + 2\sqrt{\frac{Acc/100\ (1 - Acc/100)}{n}}$$

B

Dérivations pour la méthode *Affine*

Les calculs suivants permettent d'aboutir à la détermination des paramètres d'une fonction de compensation affine selon l'approche du *Stochastic Matching* décrite dans [Delphin-Poulat *et al.*, 1998]. Cette approche est décrite en détails au chapitre 3. Les paramètres de la fonction *Affine* décrite à la section 5.2.4 du chapitre 5 en sont dérivés.

Dans le cadre de l'algorithme décrit dans [Delphin-Poulat *et al.*, 1998], on cherche à minimiser :

$$J(\theta) = K_t(\theta_0, \theta_1) = A\ E\left[log\left(p(\mathbf{Y}_t|\theta)\right)|\theta^0\right]$$

- θ_0 est le paramètre optimum.
- A est une constante indépendante de θ et de $\mathbf{Y}$.
- $K_t(\theta_0, \theta_1)$ est l'information de Kullback-Leibler entre deux densités de probabilité p_{θ_0} et p_{θ_1} d'une même variable aléatoire $\mathbf{Y}$

On cherche à déterminer θ de sorte que l'information de Kullback-Leibler $J(\theta)$ soit la plus proche de 0.

Il est possible d'approcher cette valeur optimale de façon incrémentale par intermédiaire de la fonction auxilaire :

$$\begin{aligned} \theta_{t+1} &= \arg\max_{\theta} Q_{t+1}(\Theta_t, \theta) \\ Q_{t+1}(\Theta_t, \theta) &= \sum_{\tau=1}^{t+1} \mathcal{L}_{\tau|t+1}(\Theta_{\tau-1}) \end{aligned}$$

avec $\Theta_t = (\theta_0, \ldots, \theta_t)$.

Considérons les modèles acoustiques comme étant des HMMs à N états, chaque état d'indice n étant caractérisé par un mélange de M Gaussiennes de moyenne $\mu_{n,k}$ et variance $\sigma_{n,k}$ pondérées par le facteur $w_{n,k}$ $(k = 1, \ldots, M)$.

La fonction auxiliaire est définie à partir de l'expression de la vraisemblance suivante.

$$\begin{aligned} \mathcal{L}_{\tau|t+1}(\Theta_{\tau-1}) \ = \ & \log(|f'_\theta(y_\tau)|) - \\ & \tfrac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{M} \gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)\frac{(f_\theta(y_\tau)-\mu_{n,k})^2}{\sigma^2_{n,k}} \end{aligned}$$

Dans laquelle

- $f'_\theta(y_\tau)$ est la dérivée partielle de la fonction de compensation par rapport à l'observation à l'instant τ, y_τ

- $\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)$ est la probabilité que le τ-iéme état de la séquence optimale d'états globale s_τ soit celui d'indice n et que sa principale composante gaussienne g_τ soit celle d'indice k, sachant la séquence partielle d'observations Y_{t+1} et la séquence des estimations précédentes des paramètres : $\Theta_{\tau-1}$.

$$\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k) = P(s_\tau|Y_{t+1},\Theta_{\tau-1})$$

Alors,

$$\theta_{t+1} = \theta_t + \frac{S(\theta_t, y_{t+1})}{I_{t+1}(\theta_t)}$$

avec

$$S(\theta_t, y_{t+1}) = \frac{\partial \mathcal{L}_{t+1|t+1}(\Theta_t,\theta)}{\partial\theta}|_{\theta=\theta_t}$$

soit

$$S(\theta_t, y_{t+1}) = m_{t+1|t+1}(\Theta_t,\theta)\frac{\partial f_\theta(y_t)}{\partial\theta}|_{\theta=\theta_t} + \frac{1}{f'_\theta(y_{t+1})}\frac{f'_\theta(y_{t+1})}{\partial\theta}|_{\theta=\theta_t}$$

avec

$$m_{\tau|t+1}(\Theta_{\tau-1},\theta) = \sum_{n=1}^{N}\sum_{k=1}^{M}\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)\frac{f_\theta(y_\tau)-\mu_{n,k}}{\sigma^2_{n,k}}$$

et

$$I_{t+1}(\theta_t) = -\frac{\partial^2 Q_{t+1}(\Theta_t,\theta)}{\partial\theta^2}|_{\theta=\theta_t}$$

Soit

$$\begin{aligned} I_{t+1}(\theta_t) &= \textstyle\sum_{\tau=1}^{t+1}\Big\{m_{\tau|t+1}(\Theta_{\tau-1},\theta)\frac{\partial^2 f_\theta(y_\tau)}{\partial\theta^2}|_{\theta=\theta_t} \\ &+ n_{\tau|t+1}(\Theta_{\tau-1},\theta)\left(\frac{\partial f_\theta(y_t)}{\partial\theta}|_{\theta=\theta_t}\right)^2 \\ &+ \frac{1}{(f'_\theta(y_{t+1}))^2}\left(\frac{\partial f'_\theta(y_t)}{\partial\theta}|_{\theta=\theta_t}\right)^2 \\ &- \frac{1}{f'_\theta(y_{t+1})}\frac{\partial^2 f'_\theta(y_t)}{\partial\theta^2}|_{\theta=\theta_t}\Big\} \end{aligned}$$

avec

$$n_{\tau|t+1}(\Theta_{\tau-1},\theta) = \sum_{n=1}^{N}\sum_{k=1}^{M}\frac{\gamma_{\tau|t+1,\Theta_{\tau-1}}(n,k)}{\sigma^2_{n,k}}$$

Dans le cas d'une application affine, $f_\theta(y_t) = a\ y_t + b$, c'est-à-dire que $\theta = [b\ a]^T$. Dans ce cas,

$$\begin{array}{llll} f_\theta(y_t) & = a\ y_t + b & f'_\theta(y_t) & = a \\ \frac{\partial f_\theta(y_t)}{\partial\theta} & = [1\ y_t]^T & \frac{\partial^2 f_\theta(y_t)}{\partial\theta^2} & = [0\ 0]^T \\ \frac{\partial f'_\theta(y_t)}{\partial\theta} & = [0\ 1]^T & \frac{\partial^2 f'_\theta(y_t)}{\partial\theta^2} & = [[0\ 0]\ [0\ 0]]^T \end{array}$$

Donc,

$$S(\theta_t, y_{t+1}) = -\begin{bmatrix} m_{t+1|t+1}(\Theta_t,\theta) \\ m_{t+1|t+1}(\Theta_t,\theta)\ y_{t+1} - \frac{1}{a_t} \end{bmatrix}$$

et

$$I_{t+1}(\theta_t) = \begin{bmatrix} \sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta) & \sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta)y_\tau \\ \sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta)y_\tau & \sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta)y_\tau^2 + \frac{t+1}{a_t^2} \end{bmatrix}$$

Dont l'inverse est :

$$I^{-1}_{t+1}(\theta_t) = \frac{1}{\delta_{t+1}}\begin{bmatrix} \sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta)y_\tau^2 + \frac{t+1}{a_t^2} & -\sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta)y_\tau \\ -\sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta)y_\tau & \sum_{\tau=1}^{t+1} n_{\tau|t+1}(\Theta_{\tau-1},\theta) \end{bmatrix}$$

où δ_{t+1} est le déterminant de $I_{t+1}(\theta_t)$.

Sous l'hypothèse formulée au chapitre 5, on peut exprimer les paramètres de la fonction de compensation sous la forme :

$$\begin{aligned} b_{t+1} &= b_t - \frac{1}{\delta_{t+1}} \left(m_{t+1}(y_{t+1}C_{t+1} - D_{t+1}) - \frac{1}{a_t} C_{t+1} \right) \\ a_{t+1} &= a_t - \frac{(m_{t+1} + E_{t+1} + \frac{t+1}{a_t^2} - (y_{t+1} - \frac{1}{a_t}) D_{t+1})}{\delta_{t+1}} \end{aligned}$$

avec :

$$\begin{aligned} \delta_{t+1} &= \frac{1}{C_{t+1}\left(E_{t+1} + \frac{t+1}{a_t^2}\right) - (D_{t+1})^2} \\ E_{t+1} &= \sum_{\tau=1}^{t+1} \frac{y_\tau^2}{\sigma^2_{(n,k)_\tau}} \\ D_{t+1} &= \sum_{\tau=1}^{t+1} \frac{y_\tau}{\sigma^2_{(n,k)_\tau}} \\ C_{t+1} &= \sum_{\tau=1}^{t+1} \frac{1}{\sigma^2_{(n,k)_\tau}} \\ m_{t+1} &= \frac{a_t y_{t+1} + b_t - \mu_{(n,k)_{t+1}}}{\sigma^2_{(n,k)_{t+1}}} \end{aligned}$$

Bibliographie

[Acero and Stern, 1990] A. Acero and R. Stern. Environmental robustness in automatic speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1990.

[Afify, 1999] M. Afify. Sequential bias compensation for robust speech recognition. In *Proceedings of the European Conference on Speech Communication and Technology*, 1999.

[Almera, 2004] J. Almera. *Robust Audio Segmentation.* PhD thesis, IDIAP, 2004.

[Barreaud *et al.*, 2003a] V. Barreaud, I. Illina, and D. Fohr. On-Line Frame-Synchronous Compensation of Non-Stationary noise. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, April 2003.

[Barreaud *et al.*, 2003b] V. Barreaud, I. Illina, D. Fohr, and F Korkmazsky. Structural State-Based Frame Synchronous Compensation. In *Proceedings of the European Conference on Speech Communication and Technology*, 2003.

[Barreaud *et al.*, 2004] V. Barreaud, I. Illina, D. Fohr, and V. Colotte. Compensation en milieu variant abruptement. In *Proceedings of the Journées d'Etudes sur la Parole*, April 2004.

[Basseville and Nikiforov, 1993] M. Basseville and I.V. Nikiforov. *Detection of Abrupt Changes : Therory and Application.* Prentice-Hall, 1993.

[Beatie and Young, 1991] V. Beatie and S. Young. Noisy speech recognition using hidden markov model state-based filtering. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1991.

[Beatie and Young, 1992] V. Beatie and S. Young. Hidden markov model state-based cepstral compensation. In *Proceedings of International Conference on Spoken Language Processing*, 1992.

[Boll, 1979] S.F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE transactions on Acoustics, Speech and Signal Processing*, 1979.

[Bonastre *et al.*, 2000] J-F. Bonastre, P. Delacourt, C. Fredouille, T. Merlin, and C. Wellekens. A speaker tracking system based on speaker turn detection for nist evaluation. In *ICASSP*, 2000.

[Bourlard and Morgan, 1994] H. Bourlard and N. Morgan. *Connectionist Speech Recognition : A Hybrid Approach.* Kluwer Academic Publishers, 1994.

[Brugnara *et al.*, 1992] F. Brugnara, R. De Mori, D. Giuliani, and M. Omologo. Improved connected digit recognition using spectral variation functions. In *Proceedings of the International Conference on Spoken Language Processing*, 1992.

[Buchner and Kellermann, 2002] H. Buchner and W. Kellermann. Improved kalman gain computation for multichannel frequency-domain adaptive filtering and application to acoustic echo

cancellation. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2002.

[Calliope, 1989] Calliope. *La parole et son traitement automatique.* Masson, 1989.

[Chen and Gopalakrishnan, 1998] S.S. Chen and P.S. Gopalakrishnan. Speaker, environment and channel change detection and clustering via the bayesian information criterion. In *DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[Colotte, 2002] V. Colotte. *Techniques d'analyse et de synthèse de la parole appliquées à l'apprentissage des langues.* PhD thesis, UHP - Nancy1, 2002.

[Cooke *et al.*, 2001] M. Cooke, P. Green, L. Josifovski, and A Vizinho. Robust automatic speech recognition with missing and uncertain acoustic data. *Speech Communication*, 2001.

[d'Allessandro, 2002] C. d'Allessandro. *Analyse, Synthèse et Codage de la Parole*, chapter 1. Mariani, J., 2002.

[de la Torre *et al.*, 2002] A. de la Torre, J Segura, M. Benitez, A. Peinado, and A. Rubio. Non-linear transformations of the feature space for robust speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2002.

[Delacourt, 2000] P. Delacourt. *La segmentation et le regroupement par locuteurs pour l'indexation de documents audio.* PhD thesis, EURECOM, 2000.

[Delphin-Poulat *et al.*, 1998] L. Delphin-Poulat, C. Mokbel, and J. Idier. Frame Synchronous Stochastic Matching Based on the Kullback-Leibler Information. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 89–92, 1998.

[Dempster *et al.*, 1977] A. Dempster, N. Laird, and D. Rubin. Maximum-likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1) :1–38, 1977.

[Deng *et al.*, 2001] L. Deng, J. Droppo, and A. Acero. Recursive noise estimation using iterative stochastic approximation for stereo-based robust speech recognition. In *Automatic Speech Recognition and Understanding Workshop*, 2001.

[Deng *et al.*, 2003] L. Deng, J. Droppo, and A. Acero. Recursive estimation of nonstationary noise using iterative stochastic approximation speech recognition. *IEEE Transaction on Speech and Audio Processing*, 11(6) :568–579, November 2003.

[Deviren and Daoudi, 2001] M. Deviren and K. Daoudi. Structural learning of dynamic bayesian networks in speech recognition. In *Proceedings of the European Conference on Speech Communication and Technology*, 2001.

[Deviren, 2004] M. Deviren. *Système de reconnaissance de la parole revisité : Réseaux Bayesiens dynamiques et nouveaux paradigmes.* PhD thesis, UHP - nancy1, 2004.

[Fohr *et al.*, 2000] D. Fohr, O. Mella, and C. Antoine. The automatic speech recognition engine espere : Experiments on telephone speech. In *Proceedings of International Conference on Spoken Language Processing*, 2000.

[Ford and Moore, 1998] J. Ford and B. Moore. On adaptive hmm state estimation. *IEEE Transaction on Speech and Audio Processing*, 1998.

[Furui, 1981] S. Furui. Cepstral analysis technique for automatic speaker verification. *IEEE Transaction on Speech and Audio Processing*, ASSP-29(2) :254–272, 1981.

[Gales, 1995] M.J.F. Gales. *Model-Based Techniques for Noise Robust Speech Recognition.* PhD thesis, Gonville and Caius College, September 1995.

[Gauvain and Lee, 1994] J.-L. Gauvain and C.-H. Lee. Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains. *IEEE Transaction on Speech and Audio Processing*, 2(2) :291–298, 1994.

[Glass *et al.*, 1996] J. Glass, J. Chang, and M. McCandless. A probabilistic framework for feature-based speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 1996.

[Gong, 1995] Yifan Gong. Speech recognition in noisy environments : A survey. *Speech Communication*, 1995.

[Gänsler and Benesty, 2000] T. Gänsler and J. Benesty. Stereophonic acoustic echo cancellation and two-channel adaptative filtering : an overview. *International Journal of Adaptative Control and Signal Processing*, 2000.

[Haton *et al.*, 1991] J-P. Haton, J. Caelen, J-L. Gauvain, G. Perennou, and J-M. Pierrel. *Reconnaissance Automatique de la Parole.* Dunod, 1991.

[Haton, 2002] J-P. Haton. *Reconnaissance de la parole*, chapter Méthodes robustes. Mariani, J., 2002.

[Hazen *et al.*, 2002] T. Hazen, I. Hetherington, H. Shu, and K. Livescu. Prononciation modeling using a finite-state transducer representation. In *ISCA Workshop on Prononciation Modeling and Lexicon Adaptation*, 2002.

[Hilger and Ney, 2001] F. Hilger and H. Ney. Quantile based histogram equalization for noise robust speech recognition. In *Proceedings of the European Conference on Speech Communication and Technology*, 2001.

[Jiang and Wang, 2004] Hui Jiang and Qi Wang. Nonlinear noise compensation in feature domain for speech recognition with numerical methods. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2004.

[Junqua and Haton, 1995] J-C. Junqua and J-P Haton. *Robustness in Automatic Speech Recognition.* Kluwer Academic Publisher, 1995.

[Kermorvant, 1999] C. Kermorvant. A comparison of noise reduction techniques for robust speech recognition. IDIAP-RR 10, IDIAP, 1999. IDIAP-RR 99-10.

[Klatt, 1976] D.H. Klatt. A digital filter-bank for spectral matching. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1976.

[Korkmazsky *et al.*, 2004] F. Korkmazsky, D. Fohr, and I. Illina. Using linear interpolation to improve histogram equalization for speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 2004.

[Kristjanson *et al.*, 2001] T. Kristjanson, B.Frey, L.Deng, and A.Acero. Toward non-stationary model-based noise adaptation for large vocabulary speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2001.

[Landercy and Renard, 1982] A. Landercy and R. Renard. *Eléments de phonétique.* Didier-CIPA, 1982.

[Lee, 1998] Chin-Hui Lee. On stochastic feature and model compensation approaches to robust speech recognition. *Speech Communication*, 1998.

[Leggetter and Woodland, 1995] C.J. Leggetter and P.C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. *Computer Speech and Language*, 9(2) :171–185, 1995.

[Linares, 1999] G. Linares. *Détection de ruptures et classification automatique dans un environnement de bruits impulsifs.* PhD thesis, Université d'Avignon et des pays du Vaucluse, 1999.

[Liu *et al.*, 1994] F. Liu, R. Stern, A. Acero, and P. Moreno. Environment normalization for robust speech recognition using direct cepstral comparison. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1994.

[Meignier, 2002] S. Meignier. *Indexation en locuteurs de documents sonores : Segmentation d'un dociment et appariement d'une collection.* PhD thesis, Université d'Avignon et des pays du Vaucluse, 2002.

[M.Gales and S.Young, 1992] M.Gales and S.Young. An improved approach to the hidden markov model decomposition of speech and noise. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1992.

[Mokbel and Chollet, 1991] C. Mokbel and G. Chollet. Speech recognition in adverse environments : speech enhancement and spectral transformations. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1991.

[Mokbel *et al.*, 1994] C. Mokbel, P. Paches-Leal, D. Jouvet, and J. Monné. Compensation of Telephone Line Effects for Robust Speech Recognition. In *Proceedings of International Conference on Spoken Language Processing*, April 1994.

[Mokbel, 1992] C. Mokbel. *Reconnaissance de la parole dans le bruit : bruitage/débruitage.* PhD thesis, ENST Paris, 1992.

[Mokbel, 2001] Chafic Mokbel. Online adaptation of hmms to real life conditions : a unified framework. *IEEE Transaction on Speech and Audio Processing*, 2001.

[Moleau *et al.*, 2001] S. Moleau, M. Pitz, and H. Ney. Histogram based normalization in the acoustic feature space. In *Automatic Speech Recognition and Understanding Workshop*, 2001.

[Moleau *et al.*, 2003a] S. Moleau, F. Hilger, D. Keyser, and H. Ney. Enhanced histogram normalization in the acoustic feature space. In *Proceedings of International Conference on Spoken Language Processing*, 2003.

[Moleau *et al.*, 2003b] S. Moleau, F. Hilger, and H. Ney. Feature space normalization in adverse acoustic conditions. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2003.

[Moreno *et al.*, 1996] P.J. Moreno, B. Raj, and R.M. Stern. A vector taylor series approach for environment-independent speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1996.

[Moreno, 1996] P. Moreno. *Speech Recognition in Noisy Environments.* PhD thesis, CMU EDE, 1996.

[Morris *et al.*, 1998] A. Morris, M. Cooke, and P. Green. Some solutions to the missing feature problem in data classification, with application to noise robust asr. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1998.

[Neumeyer and Weintraub, 1994] L. Neumeyer and M. Weintraub. Probabilistic optimum filtering for robust speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1994.

[Neumeyer and Weintraub, 1995] L. Neumeyer and M. Weintraub. Robust speech recognition in noise using adaptation and mapping techniques. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1995.

[Obushi and Stern, 2003] Y. Obushi and R. Stern. Normalization of time-derivative parameters using histogram equalization. In *Proceedings of the European Conference on Speech Communication and Technology*, 2003.

[Okazaki *et al.*, 2004] M. Okazaki, T. Kunimoto, and T. Kobayashi. Multi-stage spectral subtraction for enhancement of audio signals. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2004.

[Ostendorf *et al.*, 1996] M. Ostendorf, V. Digalakis, and O.A. Kimball. From hmms to segment models : A unified view of stochastic modelilng for speech recognition. *IEEE Transaction on Speech and Audio Processing*, 1996.

[Rabiner and Juang, 1993] L. Rabiner and B-H. Juang. *Fundamentals of Speech Recogntion.* Prentice Hall, 1993.

[Rahim and Juang, 1996] M. Rahim and B-H Juang. Signal bias removal by maximum likelihood estimation for robust telephone speech recognition. *IEEE Transaction on Speech and Audio Processing*, 4(1) :19–30, January 1996.

[Raj *et al.*, 1996] B. Raj, E. Gouvêa, P. Moreno, and R. Stern. Cepstral compensation by polynomial approximation for environment-independent speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 1996.

[Raj *et al.*, 1997] B. Raj, E. Gouvêa, and R. Stern. Cepstral compensation using statistical linearization. In *Proc. of the ESCA Tutorial and Research Workshop on Robust Speech Recognition for Unknown Communication Channels*, 1997.

[Raj *et al.*, 2001] B. Raj, M. Seltzer, and R. Stern. Robust speech recognition : the case for restoring missing feature. In *Workshop for Consistent and Reliable Acoustic Cues, CRAC*, 2001.

[Reichl and Ruske, 1995] W. Reichl and G. Ruske. Discriminative training for continuous speech recognition. In *eurospeech*, 1995.

[Sankar and Lee, 1995] A. Sankar and H. Lee, C. Robust speech recognition based on stochastic matching. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 121–124, 1995.

[Sankar and Lee, 1996] A. Sankar and C-H. Lee. A Maximum Likelihood Approach to Stochastic Matching for Robust Speech Recognition. *IEEE Transaction on Speech and Audio Processing*, pages 190–202, 1996.

[Segura *et al.*, 2002a] J. Segura, M. Benitez, A. de la Torre, and A. Rubio. Feature extraction combining spectral noise reduction and cepstral histogram equalization for robust asr. In *Proceedings of International Conference on Spoken Language Processing*, 2002.

[Segura *et al.*, 2002b] J.C. Segura, M.C. Benitez, A. de la Torre, and A.J. Rubio. Vts residual noise compensation. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2002.

[Shinoda and Lee, 2001] K. Shinoda and C.-H. Lee. A Structural Bayes Approach to Speaker Adaptation. *IEEE Transaction on Speech and Audio Processing*, 9(3) :276–286, march 2001.

[Siohan *et al.*, 1995] O. Siohan, Y. Gong, and J-P. Haton. Noise adaptation using linear regression for continuous noisy speech recognition. In *Proceedings of the European Conference on Speech Communication and Technology*, 1995.

[Siohan, 1995] O. Siohan. *Reconnaissance Automatique de la Parole Continue en Environnement Bruité : Application à des modèles Stochastiques de Trajectoires.* PhD thesis, Université Henri Poincarré - Nancy1, 1995.

[Stern *et al.*, 1992] R.M Stern, F.-H Liu, Y Ohshima, T.M. Sullivan, and A Acero. Multiple approaches to robust speech speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 1992.

[Ström *et al.*, 1999] N. Ström, L. Hetheringtton, T.J. Hazen, E. Sandness, and J. Glass. Acoustic modeling improvments in a segment-based speech recognizer. In *Automatic Speech Recognition and Understanding Workshop*, 1999.

[Surendran *et al.*, 1996] A-C Surendran, C-H Lee, and M. Rahim. Maximum-likelihood stochastic matching approach to non-linear equalization for robust speech recognition. In *Proceedings of International Conference on Spoken Language Processing*, 1996.

[Tamura and Waibel, 1988] S. Tamura and A. Waibel. Noise reduction using connectionist models. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1988.

[Treurniet and Gong, 1994] WC Treurniet and Y. Gong. Noise independant speech recognition for a variety of noise types. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1994.

[Van Hamme, 2004] H. Van Hamme. Robust speech recognition using cepstral domain missing data techniques and noisy masks. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2004.

[Varga and Moore, 1990] P. Varga and R Moore. Hidden markov model decomposition of speech and noise. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1990.

[Varga and Moore, 1991] P. Varga and R Moore. Simultaneaous recognition of concurent speech signals using hidden markov model decomposition. In *Proceedings of the European Conference on Speech Communication and Technology*, 1991.

[Varga *et al.*, 1988] A. Varga, R. Moore, J. Bridle, K. Ponting, and M. Russel. Noise compensation algorithms for use with hidden markov model based speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 1988.

[Waibel *et al.*, 1989] A. Waibel, T. Hanazawa, G. Hinton, K Shikano, and K. Lang. Phoneme recognition using time-delay neural networks. *IEEE Transaction on Speech and Audio Processing*, 1989.

[Yuk *et al.*, 1999] D. Yuk, J. Flanagan, M. Krishnamoorthy, and K. Dayanidhi. Adaptation to environment and speaker using maximum likelihood neural networks. In *Proceedings of the European Conference on Speech Communication and Technology*, 1999.

[Zhang *et al.*, 2004] Z. Zhang, T. Sugimura, and S. Furui. A tree-structured clustering method integrating noise and snr for piecewise linear-transformation-based noise adaptation. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2004.

Monsieur **BARREAUD Vincent**

DOCTORAT de l'UNIVERSITE HENRI POINCARE, NANCY 1

en INFORMATIQUE

VU, APPROUVÉ ET PERMIS D'IMPRIMER N° 1001

Nancy, le 6 décembre 2004

Le Président de l'Université

UNIVERSITÉ HENRI POINCARÉ, NANCY 1 * LE PRÉSIDENT

J.P. FINANCE

Université Henri Poincaré, Nancy I
24-30 rue Lionnois - B.P. 3069 - 54013 NANCY Cédex
Tél. : 03 83 68 20 00 - Fax : 03 83 68 21 00

Résumé

Les performances d'un système de reconnaissance automatique de la parole se dégradent lorsque les conditions de test et d'entraînement diffèrent. L'approche classique du Stochastic Matching (SM) propose une estimation en temps-différé d'une fonction de compensation qui maximise la vraisemblance de la parole compensée par rapport à la séquence de modèles proposée par le processus de reconnaissance. Nous proposons des techniques temps-réel basées sur SM : la compensation s'effectue en parallèle avec le processus de reconnaissance. Cela permet de compenser dans un environnement variant lentement. Deux améliorations ont été apportées : -Une structure arborescente de transformations permet de construire une fonction de compensation non-linéaire dépendant du type acoustique de la parole. -Un processus surveillant les changements dans l'environnement déclenche la réinitialisation du processus de compensation. Cela permet à notre algorithme de fonctionner dans des environnements variant abruptement.

Mots-clés: Reconnaissance Automatique de la parole, Compensation, Robustesse, Stochastic Matching, temps-réel

Abstract

Performances of an automatic speech recognition system degrade when test and training conditions do not match. Classical Stochastic Matching (SM) method proposes an off-line estimation of a compensation function that maximizes the likelihood of the compensated speech, given the optimal sequence of models proposed by the recognition process. We developed a new frame-synchronous technic based on SM : compensation is performed in parallel with the recognition. This is suitable to cope with slowly varying noise. We proposed two additional versions of our approach : -a tree structure of transformations is used to build a state-dependant non-linear compensation function. This is motivated by the fact that similar observations will be affected similarly by the environment. -a surveillance process monitoring the fluctuations in the environment is used to trigger the reinitialisation of the compensation process. This enables our algorithm to cope with environments experiencing sudden occurrences of noise.

Keywords: Automatic Speech Recognition, Robustness, Stochastic Matching, Frame Synchronous