UFR mathématiques et informatique — École doctorale IAEM Lorraine

Département de formation doctorale en informatique

Adaptation au locuteur de modèles acoustiques markoviens pour la reconnaissance automatique de la parole

THÈSE

présentée et soutenue publiquement le 11 Octobre 2004

pour l'obtention du

Doctorat de l'université Nancy 2

(spécialité informatique)

par

Fabrice LAURI

Composition du jury

Président :	Jeanine SOUQUIERES, Maître de conférence, Nancy 2
Rapporteurs :	Noureddine ELLOUZE, Professeur à l'Ecole Nationale d'Ingénieurs de Tunis Gérard CHOLLET, Directeur de recherche CNRS
Directeur de thèse :	Jean-Paul HATON, Professeur à l'Université Henri-Poincaré, Nancy 1
Examinateurs :	Claude MONTACIE, Professeur à l'Université de Paris 4 Dominique FOHR, Chercheur CNRS

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

Résumé

Les systèmes actuels de reconnaissance automatique de la parole souffrent d'une dégradation des performances lorsqu'ils sont utilisés dans des conditions acoustiques différentes de celles employées lors de l'apprentissage. Les techniques d'adaptation permettent de pallier ce problème. Elles permettent de réduire les différences acoustiques entre les conditions d'apprentissage et les conditions d'utilisation du système, ce qui améliore le taux de reconnaissance du système pour le nouveau locuteur considéré (adaptation au locuteur) ou le nouvel environnement visé (adaptation à l'environnement).
Les travaux décrits dans ce mémoire portent essentiellement sur l'adaptation au locuteur des modèles acoustiques d'un système de reconnaissance automatique de la parole (*SRAP*). Toutes les techniques étudiées et proposées dans ce cadre furent implantées et évaluées à l'aide du moteur de reconnaissance automatique de la parole *ESPERE*. Elles furent évaluées sur un ensemble de phrases prononcées par seize locuteurs et issues du corpus de parole *Resource Management* (*RM*).

Dans un premier temps, nous avons étudié, implanté et évalué trois des techniques d'adaptation les plus largement utilisées actuellement : *Structural Maximum Likelihood Linear Regression* (*SMLLR*), *Structural Maximum A Posteriori* (*SMAP*) et *EigenVoices* (*EV*). De ces premières expériences ont pu être mises en lumière les constatations suivantes. *SMLLR* permet d'améliorer de manière significative les performances du *SRAP* à partir de trois phrases d'adaptation. *SMAP* permet d'améliorer marginalement les performances du *SRAP* dès la première phrase d'adaptation, mais l'amélioration des performances est moins marquée que dans le cas de *SMLLR*. Enfin, *EigenVoices* permet d'améliorer de manière significative les performances du *SRAP* dès la première phrase d'adaptation. Toutefois, cette amélioration plafonne lorsque la quantité de données d'adaptation croît.

Dans un second temps, dans l'idée de résoudre le problème de saturation des performances de *EV*, nous avons proposé une amélioration des *EigenVoices* : *Structural EigenVoices* (*SEV*). *SEV* reprend un des concepts de *SMLLR*, qui consiste à accroître le nombre de paramètres qui peuvent être estimés de manière fiable en fonction de la quantité de données d'adaptation disponible. Utilisé dans *EV*, ce concept permet d'améliorer plus longtemps les performances du *SRAP* lorsque de nouvelles phrases sont disponibles.
Afin de disposer d'une technique d'adaptation qui soit efficace quelle que soit la quantité de phrases d'adaptation disponible, nous avons ensuite proposé quatre nouvelles techniques, qui allient les avantages de *SMLLR* et de *EV*. Ces techniques sont basées soit sur la combinaison de *SMLLR* et de *EV*, soit sur la combinaison de *SMLLR* et de *SEV*. Toutes ces techniques se sont révélées au moins équivalentes, en terme d'amélioration des performances, aux techniques *SMLLR*, *EV* et *SEV*, quelle que soit le nombre de phrases d'adaptation utilisées.

Enfin, la dernière partie de nos travaux a porté sur l'emploi d'algorithmes génétiques pour adapter les modèles acoustiques d'un *SRAP*. Toutes les techniques d'adaptation classiques fournissent une solution optimale locale, alors que les algorithmes génétiques sont capables de fournir plusieurs solutions optimales, dont certaines d'entre elles peuvent théoriquement être des solutions optimales globales. Nous avons montré expérimentalement, d'une part, qu'un algorithme génétique peut fournir une amélioration des performances similaire à *EV* en utilisant une seule phrase d'adaptation. D'autre part, la combinaison de l'algorithme génétique et de *EV* permet d'améliorer encore ces performances.

Mots clés : adaptation au locuteur, modèles de Markov cachés, reconnaissance automatique de la parole, ESPERE, *Structural Maximum Likelihood Linear Regression*, *SMLLR*, *Structural Maximum A Posteriori*, *SMAP*, *EigenVoices*, *EV*, algorithme génétique.

Abstract

Automatic speech recognition systems suffer from a degradation of performance when they are used in acoustic conditions which are different than those employed during the training phase. Adaptation methods can resolve this problem. They can reduce the acoustic mismatch between the training and using conditions, so that they can improve the recognition rate of the system for a new speaker (speaker adaptation) or for a new environment (environment adaptation).
This thesis focus on speaker adaptation of the acoustic models of an automatic speech recognition system (ASRS). All of the studied and proposed techniques were developed and evaluated using the speech recognition engine *ESPERE*. They were evaluated on a set of sentences pronounced by sixteen speakers and available in the speech corpus *Resource Management* (*RM*).

First of all, we have studied, developed and evaluated the most three currently widely used adaptation techniques : *Structural Maximum Likelihood Linear Regression* (*SMLLR*), *Structural Maximum A Posteriori* (*SMAP*) and *EigenVoices* (*EV*). From these experiences we made several conclusions. *SMLLR* can significantly improve the performances of an ASRS as soon as three sentences are available. *SMAP* can improve the performances of an ASRS as soon as only one sentence is used, but the improvement is less significant than with *SMLLR*. Finally, *EV* can significantly improve the performances of an ASRS with only one sentence. Yet, this improvement reach a plateau when the number of adaptation sentences increase.

Secondly, in order to resolve the problem of saturation of *EV*, we have proposed a improvement of *EV* : *Structural EigenVoices* (*SEV*). This technique is based on a concept of *SMLLR*. This concept enables *SEV* to increase the number of parameters which can be robustly estimated in function of the available amount of adaptation data. Thus, the performances of the ASRS can be improved much longer when more adaptation sentences become available.
To have an adaptation technique which are efficient whatever the number of adaptation sentences is used, we then proposed four new methods combining the advantages of *SMLLR* and *EV*-based techniques. These techniques are based either on the combination of *SMLLR* and *EV*, or on the combination of *SMLLR* and *SEV*. We showed that all of them were able to improve the performances of an ASRS as significantly as *SMLLR*, *EV* and *SEV*, whatever the available amount of adaptation sentences.

Finally, the last part of our works was focused on the use of genetic algorithm to adapt the acoustic models of an ASRS. All of the standard adaptation methods provide an optimal and local solution, whereas genetic algorithms are theoretically able to provide several optimal and global solutions. We experimentally showed that, on the one hand, a genetic algorithm and *EV* can both equivalently improve the performances of an ASRS. On the other hand, combining a genetic algorithm and *EV* can further improve the performances of the ASRS.

Keywords : speaker adaptation, Hidden Markov Models, automatic speech recognition, ESPERE, *Maximum Likelihood Linear Regression*, *Structural Maximum A Posteriori*, *EigenVoices*, genetic algorithm.

Remerciements

Il peut être très difficile de se décider à commencer un doctorat. Un doctorat constitue une aventure scientifique, humaine et introspective de plusieurs années, en général de trois ans, qui a un impact sensible sur la carrière d'un futur chercheur, surtout en cette période actuelle où la reconnaissance de l'état vis-à-vis de la recherche n'est pas très démonstrative. Plusieurs éléments sont alors à mettre dans la balance avant de se lancer dans un tel projet : sa propre motivation, celle des acteurs du projet, ses aspirations, ses convictions... En ce qui me concerne, l'une des principales raisons qui m'a poussé à me lancer dans cette aventure était de montrer, à moi-même et aux autres, que j'en étais capable.
Les personnes que je vais citer ci-dessous ont toutes contribué, à leur manière, à cristalliser cette reconnaissance naissante.

Mes remerciements s'adressent dans un premier temps aux membres de mon jury, à mon directeur de thèse, à mes encadrants de thèse, à mes responsables de stage de DEA et à notre assistante de projet au sein de l'équipe.
Je voudrais ainsi tout d'abord remercier *Jeanine Souquières*, *Noureddine Ellouze*, *Gérard Chollet* et *Claude Montacié*, pour avoir accepté de faire partie de mon jury de thèse. Toutes ces personnes ont consacré du temps pour relire un document de plus d'une centaine de pages et pour me transmettre leurs remarques et leurs conseils afin d'améliorer sa compréhension, sa clarté et la présentation des résultats.
Je tiens à remercier mon directeur de thèse, *Jean-Paul Haton*, pour m'avoir proposé ce sujet de thèse lors d'une période de doute, alors que la recherche laborieuse d'un financement rendait la perspective de commencer un doctorat très incertaine. Il m'a ainsi permis de l'effectuer dans des conditions très satisfaisantes.
Mes remerciements vont bien sûr également à mes encadrants de thèse, *Irina Ilina* et *Dominique Fohr*, pour leur expérience solide dans le domaine de la reconnaissance automatique de la parole, pour leur disponibilité, leur écoute et leurs recommandations constructives. Nos réunions fréquentes m'ont souvent permis de clarifier mes idées, d'améliorer mes analyses de résultats ou d'orienter mes recherches.
Je voudrais adresser un merci tout particulier à mes responsables de stage pendant mon DEA, *Anne Boyer* et *François Charpillet*, qui m'ont fait connaître le vaste domaine engouffrant de l'Intelligence Artificielle, et en particulier celui très passionnant des systèmes multi-agents.
Merci également à *Martine Kuhlmann*, notre assistante de projets au sein de l'équipe Parole, pour sa gentillesse infinie, son dévouement et son efficacité à accomplir les tâches administratives rencontrées pendant la thèse.

Je voudrais maintenant transmettre mes remerciements à mes amis et collègues que j'ai cotôyé au LORIA tout au long de cette aventure scientifique.
Merci à *Filipp Korkmazsky*, pour son aide et ses conseils sur le paramétrage de l'algorithme génétique utilisé pour adapter directement les moyennes d'un système de *RAP*. Merci à *Vincent Barreaud* et *David Langlois* pour avoir consacré du temps à lire et à annoter un manuscrit qui n'avait pas encore toute sa maturité. Vos suggestions m'ont permis de

présenter mes travaux d'une manière plus accessible et moins rébarbative. Merci aussi à *Vincent Thomas* et *Vincent Barreaud* pour avoir tenu bon et pour vos conseils lors de ma présoutenance improvisée. Merci également à *Jean-Luc Metzger*, *Martine Cadot* et *Joseph Di Martino*, pour les dialogues endiablés qu'ils ont fait naître et ont entretenu dans notre bureau. Ceux-ci constituaient des intermèdes vivifiants et très humoristiques... Un autre merci à *Joseph*, adversaire tenace et solide aux échecs, pour avoir pris part à mon passe-temps favori [1].

Par ailleurs, je remercie ma famille, en particulier mes parents, qui ont sûrement plus d'une fois douté sur les perspectives offertes par un doctorat, mais qui m'ont finalement fait confiance et toujours soutenu.
Enfin, je tiens à adresser mes remerciements les plus respectueux à ma femme et à mon fils qui, par leur présence, leur attention et leur amour, m'ont sans cesse revitaliser et donner du courage pour atteindre l'autre bout du tunnel...

1. J'en profite pour saluer *Kaïssa*, muse des échecs, qui a su me résonner afin que ce passe-temps ne devienne pas trop envahissant. ;-)

A toute ma famille.

Table des matières

Table des figures

Liste des tableaux

Glossaire

EM	***Expectation-Maximisation*** (algorithme d'estimation des paramètres de HMM)
EV	***EigenVoices*** (technique d'adaptation des Voix Propres)
GA	**Genetic Algorithm** (technique d'adaptation proposée)
HMM	***Hidden Markov Models*** (modèles acoustiques)
RAP	**Reconnaissance automatique de la parole**
RM	***Resource Management*** (corpus de parole)
SEV	***Structural EigenVoices*** (technique d'adaptation proposée)
SMAP	***Structural Maximum A Posteriori*** (technique d'adaptation)
SMLLR	***Structural Maximum Likelihood Linear Regression*** (technique d'adaptation)
SRAP	**Système de reconnaissance automatique de la parole**

Introduction

Depuis la construction du premier ordinateur, les moyens de lui transmettre des informations n'ont jamais cessé de s'améliorer. De la méthode fastidieuse d'insertion de cartes programmées à l'utilisation de claviers, souris, stylos optiques, tablettes graphiques, écrans tactiles ou encore de systèmes vocaux, les périphériques d'entrée de données deviennent de plus en plus intuitifs, faciles et rapides à utiliser.

La communication orale présente cependant de nombreux avantages par rapport aux autres outils de communication entre l'homme et la machine. La parole constitue le moyen le plus naturel de communiquer entre personnes, et donc *a fortiori* entre une personne et un système informatique. Sa puissance d'expression, caractérisée par une compacité de l'information, fait de la parole le moyen le plus rapide de transmettre et de recevoir des informations. En outre, la communication orale ne requiert aucune formation particulière de la part de l'utilisateur, ce qui n'est pas le cas lorsqu'il doit saisir des informations au clavier ou en manipulant une souris. La parole est également pratique lorsque l'utilisateur a les yeux ou les mains occupées, que ce soit lors de la conduite d'un véhicule ou lors d'une activité d'assemblage ou de maintenance. La parole est enfin conviviale lorsqu'un clavier n'est pas envisageable, ce qui est le cas pour les personnes handicapées ou les personnes ayant une activité mobile (comme les commerciaux par exemple).

Le besoin de disposer d'un système de reconnaissance automatique de la parole ($SRAP$), capable de reconnaître n'importe quelle phrase prononcée par n'importe quel locuteur dans n'importe quel environnement, est aujourd'hui grandissant. Un tel système révolutionnerait ainsi la communication entre l'homme et la machine, aussi bien du point de vue de la vitesse que de la simplicité de la transmission des informations.

La complexité inhérente des langues naturelles, tant au niveau acoustique que linguistique, rend toutefois très difficile la tâche de conception d'un tel système de reconnaissance automatique de la parole. Le signal de parole numérisé à partir d'une phrase prononcée par un locuteur est en effet très variable : il dépend fortement de ses conditions d'émission, qui sont dues au locuteur, à l'environnement ou au microphone. Par exemple, les signaux de parole extraits d'une même phrase prononcée par deux locuteurs distincts seront totalement différents. C'est également le cas si un locuteur a prononcé successivement deux fois la même phrase, ou s'il parle successivement dans deux environnements différents (en environnement calme et en environnement bruité par exemple).

Pour fournir l'énoncé de la phrase qui a été prononcée par un locuteur, les systèmes de RAP actuels utilisent dans un premier temps des modèles acoustiques. Chacun de ces modèles représente la signature acoustique d'un son de parole (généralement un phonème), c'est-à-dire la manière dont ce son est prononcé. Le $SRAP$ utilise ces modèles acoustiques (des modèles de Markov cachés généralement) pour identifier, à partir d'un signal de parole, à quel instant et dans quel ordre tel son de parole a été prononcé. Une fois qu'une ou que plusieurs suites de sons de parole possibles ont été trouvées, le $SRAP$ utilise alors un modèle de langage, afin de déterminer la suite de mots la plus probable correspondant à l'énoncé de la phrase qui a été prononcée.

1 Problématique

Avant d'être utilisé, tout système de *RAP* est soumis à une phase d'apprentissage. Cette phase consiste à utiliser un corpus de parole d'une part pour apprendre les paramètres des modèles acoustiques, et un corpus textuel d'autre part pour apprendre les paramètres du modèle de langage. La phase d'apprentissage dicte les conditions d'utilisation du *SRAP*. Ses performances sont effectivement largement influencées par cette phase. Si le corpus de parole utilisé en apprentissage regroupe les phrases de plusieurs locuteurs et si un nombre suffisant de phrases par locuteur est disponible, le *SRAP* obtenu après apprentissage des modèles sera capable de reconnaître correctement les phrases prononcées par un grand nombre de locuteurs. A l'inverse, si le corpus de parole utilisé est essentiellement constitué de phrases provenant d'un seul locuteur, le *SRAP* délivrera de très bonnes performances pour ce locuteur, mais sera inutilisable pour un autre locuteur. Les mêmes remarques peuvent être émises lorsque l'environnement change entre la phase d'apprentissage et la phase d'utilisation du *SRAP*.
Pour pallier ce problème de dégradation des performances d'un *SRAP* lorsque les conditions d'apprentissage et les conditions d'utilisation du *SRAP* sont différentes, des techniques d'adaptation sont communément utilisées.

2 Contributions

Cette thèse s'inscrit dans le domaine de l'adaptation au locuteur de modèles acoustiques markoviens, en vue d'une reconnaissance automatique de la parole. Au cours de cette thèse, nous nous sommes intéressés essentiellement à des techniques capables d'améliorer le taux de reconnaissance en mot d'un *SRAP* lorsqu'il est utilisé par un nouveau locuteur, c'est-à-dire un locuteur n'ayant pas été utilisé pour apprendre les modèles acoustiques du *SRAP*.

Les contributions de cette thèse portent sur trois points :

1. Nous avons fournit dans un premier temps une étude comparative et détaillée de trois techniques d'adaptation au locuteur. Ces trois techniques furent implantées dans le même moteur de *RAP* (*ESPERE*), si bien que les résultats expérimentaux obtenus peuvent être comparés de manière cohérente et donner lieu à des conclusions directement exploitables.
   Cette étude est originale dans le sens où ces trois techniques furent comparées selon plusieurs critères : selon l'amélioration du taux d'erreur en mot du *SIL*, selon la complexité et selon la place mémoire nécessaire à leur implémentation. Cette étude nous a ainsi permis d'orienter nos recherches ultérieures.
2. C'est ainsi que nous avons décidé d'améliorer dans un second temps une des techniques d'adaptation au locuteur existante (*EigenVoices*), et de proposer quatre techniques capables de délivrer, quelle que soit la quantité disponible de données d'adaptation, des performances au moins égales à celles délivrées par la meilleure des techniques précédentes.

3. Enfin, nous avons réalisé des investigations sur l'emploi d'algorithmes génétiques pour adapter les modèles acoustiques d'un *SRAP* dans le cas d'une adaptation rapide, lorsque moins de trois secondes de parole sont disponibles. Ces travaux ont permis de montrer que les algorithmes génétiques sont capables de délivrer des performances supérieures à celles obtenues avec les techniques précédentes lorsqu'une seule phrase d'adaptation est disponible.

3 Plan du mémoire de thèse

Nous avons divisé ce mémoire en trois parties :

La première partie est une introduction au domaine de l'adaptation au locuteur pour la reconnaissance automatique de la parole. Il comporte deux chapitres :

Le premier chapitre porte sur le domaine de la reconnaissance automatique de la parole. La complexité du signal de parole est tout d'abord abordé, et les facteurs de variabilité du signal de parole sont passés en revue. Nous décrivons ensuite quels sont les éléments qui sont actuellement utilisés par la plupart des systèmes de *RAP* afin de leur permettre de fournir l'énoncé d'une phrase prononcée par un locuteur. Les modèles de Markov cachés, ainsi que les stratégies de reconnaissance et les principes utilisés dans le cadre de l'apprentissage des modèles acoustiques sont présentés plus particulièrement.

Le deuxième chapitre est destiné à fournir les bases théoriques nécessaires à la compréhension des principes formulés dans le cadre de l'adaptation au locuteur des modèles acoustiques dans le but d'une reconnaissance automatique de la parole ultérieure.

La deuxième partie présente trois techniques d'adaptation au locuteur qui sont actuellement les plus utilisées dans le cadre de l'adaptation au locuteur des modèles acoustiques. Nous avons implanté ces trois techniques dans le moteur de reconnaissance *ESPERE* et nous les avons évaluées à l'aide du corpus *Resource Management*. Cette étude théorique et pratique de quelques techniques existantes d'adaptation nous a semblé indispensable afin de mieux cerner les problèmes inhérents au domaine de l'adaptation pour la *RAP*. Cette partie s'étend du chapitre 3 au chapitre 7.

Le troisième chapitre décrit en particulier dans quelles conditions furent évaluées toutes les techniques d'adaptation, aussi bien celles étudiées que celles proposées dans cette thèse. Nous y présentons les caractéristiques du corpus de parole utilisé pour l'apprentissage et l'adaptation des modèles acoustiques d'une part, et pour les tests de reconnaissance d'autre part. Le moteur de reconnaissance *ESPERE*, développé dans l'équipe PAROLE au LORIA et destiné à résoudre des tâches de reconnaissance vocale pour petit vocabulaire (moins de mille mots), est également présenté dans ce chapitre.

Le quatrième chapitre présente la technique d'adaptation *Structural Maximum Likelihood Linear Regression* (*SMLLR*), ainsi que les résultats expérimentaux obtenus avec cette technique.

Le cinquième chapitre décrit la technique *Structural Maximum A Posteriori*(*SMAP*) et présente ses résultats expérimentaux.

Le sixième chapitre présente enfin les concepts adoptés par la technique *Eigen-Voices* (*EV*) ainsi que les résultats des expériences réalisées avec cette technique.

Dans le septième chapitre nous dressons un bilan comparatif des techniques étudiées dans les chapitres 4, 5 et 6. Ces trois techniques sont comparées en terme d'amélioration du taux d'erreur en mot, en terme de complexité en temps de calcul et en terme de complexité en place mémoire.

La troisième et dernière partie porte enfin sur nos contributions dans le cadre de l'adaptation au locuteur pour la reconnaissance automatique de la parole. Nous avons proposé dans un premier temps une technique d'adaptation continue, c'est-à-dire une technique qui permet d'améliorer les performances d'un *SRAP* quel que soit le nombre de phrases d'adaptation disponibles. Dans un second temps, nous avons proposé une technique performante pour l'adaptation rapide d'un *SRAP*, lorsque peu de phrases d'adaptation sont disponibles pour un locuteur. L'ensemble de ces travaux est présenté dans les chapitres 8 et 9.

Le huitième chapitre concerne l'adaptation continue. Nous y présentons tout d'abord la technique structurelle de *EigenVoices*, appelée *Structural Eigen-Voices* (*SEV*). Cette technique fut proposée pour améliorer les performances de *EV* lorsque la quantité disponible de données d'adaptation devient importante. Quatre techniques d'adaptation combinant les concepts de *SMLLR* d'une part, et de *EV* ou de *SEV* d'autre part, sont également proposées, afin de disposer d'une technique efficace quelle que soit le nombre de phrases d'adaptation disponibles.

Le neuvième chapitre porte sur l'utilisation d'algorithmes génétiques pour adapter les modèles acoustiques d'un *SRAP* dans le cas où la quantité de données d'adaptation disponibles est faible. Les algorithmes génétiques ont été utilisés avec succès dans de nombreux domaines, et entre autres dans le cadre de l'apprentissage. Néanmoins, leur utilisation dans le domaine de l'adaptation de modèles acoustiques markoviens pour la reconnaissance automatique de la parole n'a jamais été explorée jusqu'à présent. Nous avons imaginé, implanté et évalué deux techniques d'adaptation au locuteur basées sur les principes des algorithmes génétiques.

Nous donnerons enfin dans le dernier chapitre de ce mémoire, nos conclusions sur les travaux que nous avons effectués, ainsi que les perspectives que nous envisageons pour la poursuite de notre travail.

Première partie

Introduction à l'adaptation pour la reconnaissance automatique de la parole

Chapitre 1

La reconnaissance automatique de la parole

Avant d'aborder le sujet de cette thèse, à savoir l'adaptation pour la reconnaissance automatique de la parole, il est indispensable de comprendre comment un système de reconnaissance automatique de la parole opère pour fournir l'énoncé de la phrase qui a été prononcée par un locuteur.

La difficulté de la tâche de reconnaissance automatique de la parole est essentiellement due au caractère très variable du signal de parole. Pour prendre le plus fidèlement en compte la variabilité inhérente à la production de la parole, les systèmes actuels de reconnaissance automatique de la parole, aussi bien commerciaux que ceux développés et évalués dans les laboratoires de recherche, sont pratiquement tous fondés sur la même architecture. Ils utilisent les mêmes familles d'algorithmes de reconnaissance et d'apprentissage, basés sur les modèles de Markov cachés, et exploitent les mêmes types de composants pour reconnaître une phrase. De tels systèmes disposent des modules de numérisation et de paramétrisation du signal, qui transforme le signal de parole capté par le microphone en une suite de trames acoustiques pouvant être plus aisément manipulées par l'ordinateur, et d'un moteur de reconnaissance, qui utilise les trames acoustiques et exploitent les connaissances acoustiques et linguistiques modélisées par ses divers composants pour délivrer l'énoncé de la phrase qui a été prononcée. Ces composants sont au nombre de trois et incluent : le vocabulaire [2] de formes graphiques, qui contient, en plus de la liste des mots qui peuvent être effectivement reconnus, une ou plusieurs transcriptions possibles de chaque mot du vocabulaire en terme d'unité acoustique [3] ; les modèles acoustiques caractérisant les réalisations possibles des unités acoustiques composant les mots du vocabulaire ; et enfin un modèle de langage, qui définit l'ensemble des phrases pouvant être reconnues.

Le caractère variable du signal de parole est abordé dans la première section de ce chapitre, en insistant surtout sur les variabilités dues au locuteur, à savoir les variabilités inter-locuteurs et intra-locuteur. Nous présenterons ensuite dans la deuxième section l'ap-

2. appelé également dictionnaire de prononciation ou lexique de prononciation
3. Une unité acoustique est le plus petit élément de parole reconnaissable par un *SRAP*. Il peut représenter une syllabe, un phonème, etc.

proche générale communément adoptée par les systèmes de reconnaissance automatique de la parole actuels pour reconnaître l'énoncé d'une phrase qui a été prononcée par un locuteur, à partir du signal de parole de cette phrase. Nous détaillerons ensuite les différents composants énumérés précédemment et qui entrent en jeu dans le processus de décision d'un *SRAP*. La quatrième section portera sur la manière dont tous ces composants sont utilisés pour reconnaître le texte d'une phrase. Nous aborderons enfin dans la dernière section l'apprentissage des modèles acoustiques et linguistiques, en mettant plus particulièrement l'accent sur les modèles acoustiques, centres d'étude de cette thèse.

1.1 Complexité du signal de parole

La complexité du signal acoustique de parole résulte de l'interaction de nombreux facteurs de variabilité (figure 1.1).

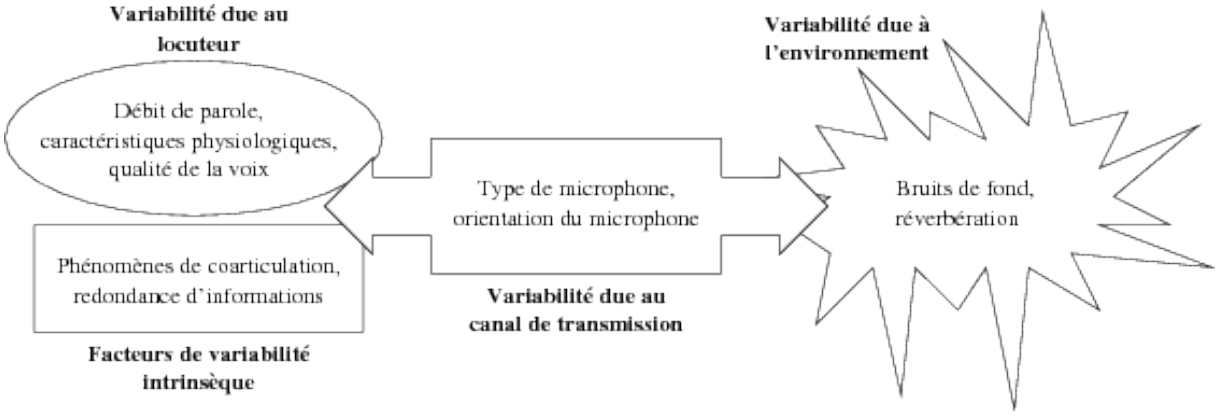


FIGURE 1.1 – Sources de variabilité du signal de parole

Certains sont inhérents au signal de parole, comme la redondance des informations qui y sont contenues ou les effets de la coarticulation. Les autres facteurs correspondent aux sources de variabilités, qui peuvent rendre la représentation de deux signaux acoustiques correspondant au même message très différentes. Ces sources de variabilités au niveau du signal acoustique sont dues au locuteur lui-même, à l'environnement ou au canal de transmission du signal (microphone).

1.1.1 Redondance des informations contenues dans le signal

La représentation dans le domaine temporel du signal acoustique numérisé est caractérisée par une redondance d'informations qui ne sont pas fondamentalement nécessaires pour reconnaître correctement le message qui a été prononcé. Outre le message proprement dit, la communication parlée véhicule effectivement de nombreuses autres informations para-linguistiques, comme le sexe du locuteur, son identité, son état de santé, son état émotionnel, etc. Pour un *SRAP*, ce flux d'informations représente une quantité colossale

de données à exploiter. Par exemple, un signal échantillonné à 16 kHz sur 16 bits (paramètres qui sont habituellement utilisés pour la voix) représente un débit de 256 KBits/s, ce qui implique que le *SRAP* doit traiter 32000 octets de données par seconde. Pour des raisons de rapidité d'exécution, tout *SRAP* cherchera donc à minimiser ce flux important de données en ayant recours à une étape de prétraitement du signal, afin de le débarasser des informations superflues et inutiles pour la reconnaissance d'un message.

1.1.2 Phénomènes de coarticulation

Tout message peut être décomposé en une suite de mots, qui peuvent à leur tour être décrits comme une suite d'unités acoustiques. Cela laisse supposer que la parole est un processus séquentiel, au cours duquel des unités élémentaires et indépendantes se succèdent. Toutefois, les phonéticiens eux-mêmes éprouvent parfois des difficultés à identifier individuellement ces sons caractéristiques du langage dans un signal de parole, même si quelques évenements acoustiques particuliers peuvent être détectés. La parole est en réalité un continuum sonore, où il n'existe pas de pause perceptible entre les mots qui pourrait faciliter leur localisation automatique par un *SRAP*.
En outre, lors de la production d'un message, l'inertie de l'appareil phonatoire et l'anticipation du geste articulatoire influencent la production de chaque son, si bien que la réalisation acoustique d'un son est fortement perturbée par les sons qui le précèdent mais également par ceux qui le suivent. Ces effets s'étendent sur la durée d'une syllabe, voire même au-delà, et sont amplifiés par un rythme d'élocution soutenu.

Le choix de l'unité acoustique directement identifiable par un *SRAP* est alors primordiale. On distingue habituellement trois classes d'unités acoustiques :

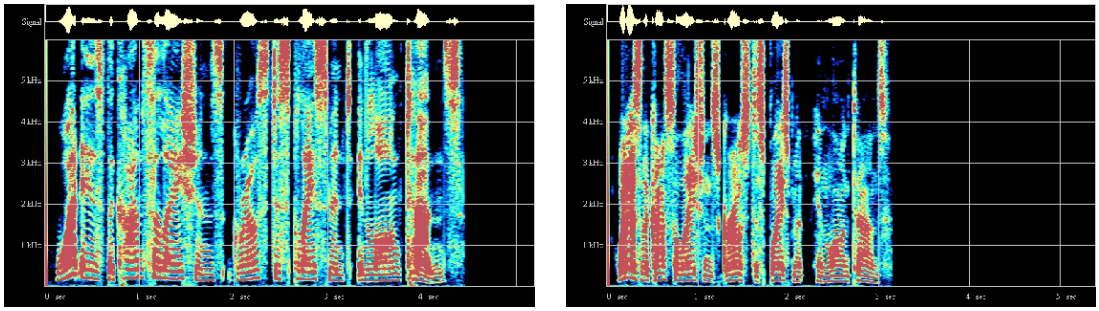
- les phonèmes,
- les unités courtes infra-phonémiques (ou phones) et
- les unités longues supra-phonémiques (diphones, triphones, semi-syllabes, syllabes, mots).

Une unité courte peut être en général mieux identifiée, mais ne possèdant pas de statut linguistique particulier, leur concaténation pour former des unités plus longues est problématique. L'utilisation de phonèmes souffre d'une mauvaise modélisation des effets de coarticulation et d'une difficulté pour les localiser. Toutefois leur nombre assez faible (une quarantaine dans la langue française) facilite la mise en œuvre du *SRAP*. En ce qui concerne les unités longues enfin, leur utilisation permet une meilleure modélisation des effets de la coarticulation interne, mais la mise en œuvre du *SRAP* n'est pas aisée en raison de leur nombre important.

1.1.3 Variabilités inter-locuteurs et intra-locuteur

La variabilité inter-locuteurs, qui est généralement considérée comme étant *a priori* la plus importante, suggère que la prononciation d'un même énoncé par deux personnes est différente. Les différences physiologiques entre locuteurs de l'appareil phonatoire, comme la longueur du conduit vocal, la forme et le volume des cavités résonnantes, ou la forme

des lèvres, influencent la réalisation acoustique d'un même message. Pour s'en convaincre, il suffit de considérer par exemple les voix d'enfants et d'adultes, qui sont les plus reconnaissables car les caractéristiques de leurs appareils phonatoires sont les plus différenciées [75].
A ces différences physiologiques s'ajoutent les habitudes acquises au sein du milieu social et géographique, comme la vitesse d'élocution, ou les accents régionaux. Dans la figure 1.2, deux locuteurs ont prononcé le même message, le premier avec un débit de parole normale, le second avec un débit de parole rapide. Ces différences au niveau de la réalisation d'un même message sont clairement observables sur les signaux de parole et sur les spectrogrammes représentés dans la figure.



Première voix d'homme Deuxième voix d'homme

FIGURE 1.2 – Variabilité inter-locuteurs

La variabilité intra-locuteurs, loin d'être négligeable, suggère qu'un locuteur ne peut prononcer deux fois à l'identique le même énoncé. L'état physique (comme la fatigue ou un rhume), l'état émotionnel (comme la gaieté ou la mélancolie) ou l'état psychologique (comme le stress) d'une personne influencent effectivement son expression orale, entraînant des variations complexes de la vitesse d'élocution, de la prosodie et de l'intensité du discours.

Tous ces phénomènes sont encore mal modélisés à ce jour et compliquent la conception des systèmes de reconnaissance automatique de la parole.

1.1.4 Variabilités dues à l'environnement et au canal de transmission

L'absence de bruit de fond est dans la pratique impossible. A moins d'être dans une chambre anéchoïque, n'importe lequel des appareils que nous utilisons émet un bourdonnement qui est la plupart du temps audible et qui génère des parasites dans le signal acoustique. Dans certains cas, ce bruit de fond peut être si élevé qu'il influe directement sur la prononciation du locuteur, le poussant à ralentir son élocution et à augmenter l'intensité sonore de son discours (effet Lombard [60]).

Par ailleurs, le microphone utilisé par le locuteur pour transmettre son message au système

possède des caractéristiques spécifiques [4] et peut alors avoir des qualités d'acquisition plus ou moins bonnes de certaines fréquences. L'acquisition de certaines fréquences peut également être rendue imparfaite selon l'angle et la distance du microphone lors de son utilisation [15]. Enfin, le canal de transmission (fil, ondes radio, etc.) peut introduire des parasites dans le signal.

1.2 Paradigme du processus décisionnel des *SRAP*

Pour qu'un système de reconnaissance automatique de la parole puisse déterminer de manière fiable et robuste le texte exact de la phrase qui a été prononcée, il devrait pouvoir prendre en compte l'ensemble de ces facteurs de variabilité. Pour cela, de nombreuses connaissances, portant sur les aspects acoustiques et linguistiques de la parole, doivent être modélisées. Alors que les connaissances acoustiques décrivent les propriétés spectrales structurelles et temporelles du signal vocal, les connaissances linguistiques décrivent plutôt les processus mentaux mis en œuvre dans l'élaboration d'un message, ces processus mentaux mettant en jeu des connaissances pragmatiques, sémantiques, syntaxiques, lexicales, morphologiques, phonologiques et phonétiques [13].
Nous allons découvrir dans les paragraphes suivants comment la plupart de ces connaissances sont intégrées et utilisées dans un système de reconnaissance automatique de la parole afin qu'il puisse reconnaître la phrase prononcée.

1.2.1 Etapes intervenant dans le processus décisionnel d'un *SRAP*

L'architecture typique d'un système de reconnaissance automatique de la parole (figure 1.3) est constituée de trois modules, qui sont le module de numérisation du signal de parole, le module de paramétrisation du signal et le décodeur, également appelé moteur de reconnaissance.

Etape de numérisation Le locuteur prononce une phrase comportant N_M mots $M = m_1, m_2, \cdots, m_{N_M}$ par l'intermédiaire d'un microphone.
Ce message M est transmis dans le canal du microphone sous la forme d'ondes électriques, qui sont transformées en un signal acoustique S par un processus de numérisation. Le signal ainsi numérisé est représenté dans le domaine temporel.

Etape de paramétrisation Le signal acoustique S est décomposé en une séquence de N_T trames de parole $T = (t_1, t_2, \cdots, t_{N_T})$ de durée égale. Pour chaque trame t_i, pour $i = 1, 2, \cdots, N_T$, est calculé un vecteur d'observations acoustiques o_i, formant ainsi une séquence d'observations acoustiques $O = (o_1, o_2, \cdots, o_{N_T})$. Cette étape de *traitement du signal vocal* permet de transformer le signal brut en séquence de paramètres plus robustes et plus significatifs, fondés sur des critères perceptifs. Elle permet de fournir une représentation du signal moins redondante, si bien que le flux d'informations acoustiques à traiter ultérieurement par le moteur de reconnaissance est réduit.

4. Parfois deux microphones de même modèle et de même fabricant peuvent avoir des caractéristiques différentes.

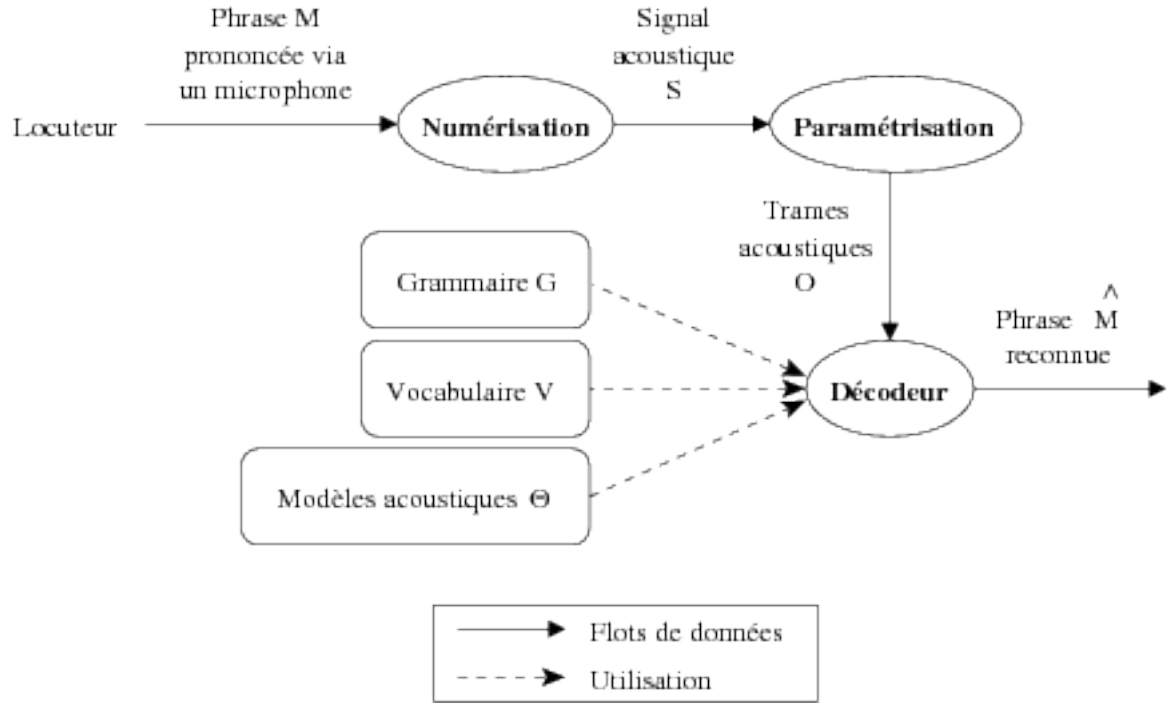


FIGURE 1.3 – Processus de décision d'un système de reconnaissance automatique de la parole

Etape de décodage Considérons que l'on a défini :

- un vocabulaire V des mots (ou formes graphiques) pouvant être effectivement reconnus par le système de reconnaissance automatique de la parole,
- un ensemble G, généré par une grammaire, comportant toutes les phrases (ou séquences de mots) qui peuvent être prononcées et reconnues par le *SRAP*,
- un ensemble de modèles acoustiques caractérisant chacun la manière de prononcer un son spécifique. Ce son peut représenter soit une unité acoustique courte (comme un phone), soit un phonème, soit une unité acoustique longue (comme une syllabe par exemple).

L'objectif du moteur de reconnaissance (ou décodeur) est de déterminer, parmi l'ensemble des phrases de G, la séquence de mots $\hat{M} = (m_1, m_2, \cdots, m_N)$ qui correspond le mieux au signal acoustique S et telle que $m_i \in V \quad \forall i = 1, 2, \cdots, N$ et $\hat{M} \in G$. Pour cela, le moteur de reconnaissance effectue un décodage acoustico-phonétique qui consiste à aligner les trames acoustiques O avec les modèles acoustiques Θ. Une fois que la séquence de modèles acoustiques est connue, la séquence de formes graphiques (donc la phrase à reconnaître) peut être reconstituée en utilisant le dictionnaire de prononciation.

Nous allons dans la section suivante focaliser notre attention sur la manière dont opère le moteur de reconnaissance d'un système de reconnaissance automatique de la parole afin de fournir l'énoncé d'une phrase.

1.2.2 Approche globale probabiliste du décodage

Le moteur de reconnaissance de la plupart des systèmes de reconnaissance automatique de la parole actuels est basé sur l'approche globale probabiliste, proposée par J.-K. Baker et F. Jelinek [4; 55].
Cette approche consiste à effectuer un raisonnement probabiliste, en utilisant conjointement des connaissances acoustiques et linguistiques, pour déterminer le texte de la phrase prononcée par un locuteur. Par raisonnement probabiliste nous voulons signifier qu'il s'agit d'un raisonnement qui associe des probabilités à des événements acoustiques.
Pour déterminer la séquence de mots $\hat{M}$ la plus probable parmi l'ensemble de toutes les séquences de mots M possibles, l'approche globale probabiliste s'appuie sur l'équation suivante :

$$\hat{M} = \underset{M}{\operatorname{argmax}}\ p(M/O) \tag{1.1}$$

où $p(M/O)$ représente la probabilité de production de la séquence de mots M sachant que la suite d'observations O a été générée.

Comme le fait remarquer Ueberla dans [106], les réalisations acoustiques de M font partie d'un espace continu qu'il est très difficile de modéliser. En utilisant la formule de Bayes [3], cette équation peut être réécrite sous la forme :

$$\hat{M} = \underset{M}{\operatorname{argmax}}\ \frac{p(O/M)\,p(M)}{p(O)} \tag{1.2}$$

où :
- $p(O/M)$ représente la probabilité *a posteriori* d'émettre la séquence d'observations O en sachant que la phrase M a été prononcée. Ce terme modélise l'aspect acoustique du processus de production de la parole.
- $p(M)$ représente la probabilité *a priori* de générer la séquence de mots M. Ce terme modélise l'aspect linguistique du processus de production de la parole.
- $p(O)$ représente la probabilité de générer la séquence d'observations O.

Comme $p(O)$ ne dépend pas de M, ce terme peut être omis, si bien que l'équation précédente se réduit à :

$$\hat{M} = \underset{M}{\operatorname{argmax}}\ p(O/M)\,p(M) \tag{1.3}$$

L'utilisation de ces deux termes $p(O/M)$ et $p(M)$ ne requiert "plus" que la modélisation d'un espace discret, qui consiste à définir les suites possibles de mots du vocabulaire.
Selon cette formule, la tâche de la reconnaissance automatique de la parole est ainsi considérée comme un processus de décodage global, intégrant à la fois les connaissances acoustiques, qui sont représentées par les modèles acoustiques $p(O/M)$, et les connaissances linguistiques, qui sont caractérisées par le modèle de langage $p(M)$.

1.3 Composants intervenant dans le processus décisionnel d'un *SRAP*

1.3.1 Vocabulaire

Le vocabulaire d'un *SRAP* permet de modéliser les connaissances phonétiques et phonologiques nécessaires à la reconnaissance d'une phrase. Il contient les connaissances nécessaires pour transformer une séquence d'unités acoustiques données (syllabes, phonèmes, phones, etc.) en une séquente de mots.
Un vocabulaire est constitué :

- d'une part, de la liste des mots qui peuvent être effectivement reconnus par le système de reconnaissance automatique de la parole,
- d'autre part, de la ou des séquences possibles de prononciation (en terme d'unités acoustiques) de chacun des mots présents.

Par exemple, si le phonème a été choisi en tant qu'unité acoustique, le vocabulaire sera alors constitué, pour chacun des mots le constituant, de la ou des séquences phonétiques possibles qui représentent chacune une prononciation possible de ce mot.

La taille du vocabulaire, ainsi que la qualité et l'exhaustivité des prononciations associées aux mots influencent les performances en reconnaissance d'un *SRAP*. Selon la taille de son vocabulaire, on distingue habituellement trois types de systèmes de reconnaissance automatique de la parole :

- les systèmes à petit vocabulaire (moins de mille mots),
- les systèmes à vocabulaire moyen (moins de dix mille mots),
- les systèmes à grand vocabulaire (plus de dix mille mots).

Un mot ne faisant pas partie du vocabulaire ne pourra jamais être reconnu par le système [5]. Ce problème ne se pose pas pour les systèmes à petit ou moyen vocabulaire, qui sont destinés à des applications "de commandes" où le vocabulaire et la syntaxe des phrases sont très contraints. Dans le cas des systèmes à grand vocabulaire cependant, la prononciation d'un mot hors vocabulaire pourra causer en reconnaissance une erreur de remplacement de ce mot par un autre, qui engendrera par la suite d'autres types d'erreurs (comme l'insertion d'un autre mot ou l'omission d'un mot). Le choix de la liste des mots faisant partie du vocabulaire est donc primordiale, et dépend ainsi étroitement de l'application.

La conception d'un vocabulaire nécessite la définition de la ou des prononciations possibles de chacun des mots le constituant. Il peut en effet exister plusieurs prononciations pour un même mot, selon l'état psychologique, le dialecte ou l'origine géographique d'un locuteur, par exemple. Une connaissance sur les conditions d'utilisation d'un *SRAP* comme, par exemple, les caractéristiques régionales des locuteurs susceptibles d'utiliser le *SRAP*, peut alors aider à la définition de ces prononciations et permettre ainsi d'améliorer considérablement les capacités de reconnaissance pour la population de locuteurs considérée.

5. A moins que le système donne la possibilité à l'utilisateur d'épeler les mots qu'il a jugé comme ayant été reconnus avec un certain degré d'erreur.

Précisons enfin que la taille du vocabulaire ainsi que le nombre de prononciations possibles pour chaque mot du vocabulaire est un compromis entre les capacités de reconnaissance désirées du *SRAP* et son temps de réponse.

1.3.2 Modèles acoustiques

Un modèle acoustique permet de prédire la réalisation acoustique d'une unité telle qu'une syllabe, un phonème, un phone, etc. Depuis un peu plus d'une vingtaine d'années, les modèles acoustiques sont habituellement représentés par des modèles de Markov cachés (*Hidden Markov Model* ou *HMM*) [57; 80; 58; 13; 92].
Ces modèles sont actuellement parmi ceux qui permettent de caractériser le mieux possible le processus de production de la parole. Les modèles acoustiques peuvent également être représentés par des modèles stochastiques de mélanges de trajectoires, par des réseaux de neurones artificiels ou par des modèles hybrides combinant les modèles markoviens et les réseaux neuromimétiques. Nous ne les présenterons pas ici. Le lecteur pourra consulter les travaux de Illina [54] pour plus de détails sur les modèles stochastiques de mélanges de trajectoires et sur ses variantes. Il pourra également se reporter à l'état de l'art des modèles acoustiques neuromimétiques et des modèles hybrides dans [46].

1.3.2.1 Qu'est ce qu'un *HMM* ?

Un *HMM* est un automate probabiliste d'états finis (figure 1.4). Il est constitué d'états (les nœuds), reliés entre eux par des transitions (les arcs). Une transition entre un état s_i et un état s_j rend possible le passage entre ces deux états.

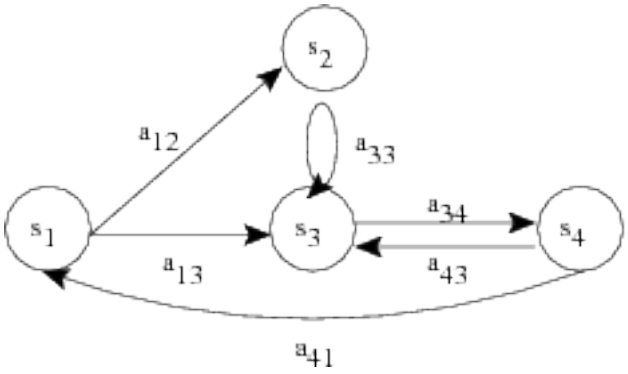


FIGURE 1.4 – Exemple d'un *HMM* d'ordre 1 à 4 états

A chaque instant t, un *HMM* se trouve dans un état $q_t \in \mathcal{S}$ avec $\mathcal{S} = \{s_1, s_2, \cdots, s_N\}$ où N est le nombre total d'états du modèle, et il émet l'observation $o_t \in \mathcal{V}$ où $\mathcal{V}$ est un ensemble éventuellement continu d'observations que le *HMM* peut potentiellement générer. A chaque état s_i est associée une distribution de probabilité d'observations $b_i(o_t)$ et à chaque transition d'un état s_i vers un autre état est associée une probabilité de transition

$p(q_t = s_i/q_1\ q_2\ \cdots\ q_{t-1})$. La distribution de probabilité représente la probabilité de générer l'observation o_t à l'état s_i au temps t ; la probabilité de transition représente la probabilité de passer d'un état vers un autre état sachant la suite d'états précédents. Cette probabilité constitue une sorte d'historique des états par lesquels le *HMM* est passé. La longueur n de cette historique $q_1 q_2 \cdots q_n$ détermine l'ordre du *HMM*. Un *HMM* d'ordre 1 ne conservera ainsi que l'état précédemment visité à l'instant $t-1$. Dans la figure 1.4, cette probabilité de transition se réduit à $a_{ji} = p(q_t = s_i/q_{t-1} = s_j)$ où seul l'état visité au temps $t-1$ est mémorisé.
A l'issue de ce processus, ce modèle permet ainsi de générer une séquence de T observations $O = (o_1, o_2, \cdots, o_T)$. Seule la séquence d'observations O est connue, la séquence d'états $Q = (q_1, q_2, \cdots, q_T)$ ayant permis de la générer restant inconnue, ce qui explique le substantif "caché" des modèles de Markov.

Un *HMM* [92] θ est défini par le quintuplet de paramètres $\theta = (\mathcal{S}, \mathcal{V}, \pi, \mathcal{A}, \mathcal{B})$ où :

- $\mathcal{S} = \{s_1, s_2, \cdots, s_N\}$ est un ensemble de N états.

- $\mathcal{V}$ est l'ensemble discret ou continu des observations qu'un *HMM* peut générer.

- $\pi = (\pi_1, \pi_2, \cdots, \pi_N)$ est l'ensemble des probabilités initiales. $\pi_i = p(q_1 = s_i)\ \ \forall i = 1, 2, \cdots, N$, représente la probabilité de se trouver dans l'état s_i à l'instant 1, avec $\pi_i\ >=\ 0\ \ \forall i$ et $\sum_{i=1}^{N}\ \pi_i = 1$.

- $\mathcal{A} = \{p(q_t = s_i/q_1 q_2 \cdots q_{t-1})\}\ \ \forall i = 1, 2, \cdots, N$ est l'ensemble des probabilités de transition.

- $\mathcal{B} = \{b_i(o)\}\ \ \forall i = 1, 2, \cdots, N$, est l'ensemble des fonctions de densité de probabilité d'observations $b_i(o)$. $b_i(o) = p(o/q_t = s_i)$ est la probabilité de générer au temps t l'observation o à l'état s_i.
  La somme des probabilités sur l'ensemble des observations continues est égale à 1, c'est-à-dire $\oint\ b_i(o)\ \delta o = 1\ \ \forall i$.

1.3.2.2 Propriétés des *HMMs* acoustiques utilisés en *RAP*

Pour réduire le temps de calcul, des hypothèses simplificatrices sont communément émises dans le cadre de la *RAP* en ce qui concerne les propriétés des modèles de Markov cachés. Ainsi :
- les observations sont considérées comme indépendantes, c'est-à-dire que $p(o_t/q_1 q_2 \cdots q_t, o_1 o_2 \cdots o_{t-1}) = p(o_t/q_1 q_2 \cdots q_t)$,

- la probabilité d'émission d'une observation ne dépend que de l'état courant. Ainsi $p(o_t/q_1 q_2 \cdots q_t) = p(o_t/q_t)$,

- les probabilités de transition sont considérées comme stationnaires, si bien que $p(q_t = s_i/q_{t-1} = s_j) = p(q_{t+v} = s_i/q_{t+v-1} = s_j)\ \ \forall v$.

En outre, dans la plupart des systèmes de reconnaissance automatique de la parole actuels, les modèles acoustiques sont représentés par des *HMMs* d'ordre 1, c'est-à-dire que la probabilité d'être dans un état donné s_i à l'instant t, en sachant que $t-1$ états ont été visités, est égal à la probabilité d'être dans l'état s_i en ne considérant que l'état s_j précédemment visité. En d'autre termes, $p(q_t = s_i/q_1 q_2 \cdots q_{t-1} = s_j) = p(q_t = s_i/q_{t-1} = s_j)$. Dans ce cas, on peut noter $a_{ji} = p(q_t = s_i/q_{t-1} = s_j)$ la probabilité de passer de l'état s_j à l'état s_i, avec $\forall i, j = 1, 2, \cdots, N$. L'ensemble des probabilités de transition du *HMM* est alors caractérisé par la matrice $N \times N$ suivante :

$$\mathcal{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{pmatrix}$$

avec $a_{ji} >= 0 \ \forall i, j$ et $\sum_{i=1}^{N} a_{ji} = 1 \ \forall j$.
Pour une utilisation de *HMMs* d'ordre 2 en *RAP*, les travaux de Jean-François Mari [84; 83] pourront être consultés.

Enfin, afin de mieux modéliser la variabilité du signal, la fonction de densité de probabilités d'observation associé à un état s_i est traditionnellement représenté par un mélange de M gaussiennes, c'est-à-dire que :

$$b_i(o_t) = \sum_{k=1}^{M} c_{i,k} \, \mathcal{N}(o_t, \mu_{i,k}, \sigma_{i,k})$$

où

$$\mathcal{N}(o_t, \mu_{i,k}, \sigma_{i,k}) = \frac{1}{\sqrt{(2\pi)^n \ |\sigma_{i,k}|}} \ exp \ \left[-\frac{1}{2} \ (o_t - \mu_{i,k})' \ {\sigma_{i,k}}^{-1} \ (o_t - \mu_{i,k}) \right]$$

et $\sum_{k=1}^{M} c_{i,k} = 1 \ \forall i$

1.3.2.3 Topologie des *HMMs* acoustiques

Considérant le caractère irréversible de l'aspect temporel de la parole, R. Bakis [5] a proposé d'utiliser un modèle type de *HMM* pour représenter une unité acoustique.

Ce modèle type (figure 1.5), appelé encore modèle "gauche-droit", permet de caractériser des progressions acoustiques qui peuvent être soit **stationnaires**, soit **normales**. Une progression acoustique stationnaire est représentée par une transition de type t_s bouclant sur l'état courant, tandis qu'une progression acoustique normale est symbolisée par une transition de type t_n entre l'état courant et l'état suivant. Le nombre d'états d'un tel *HMM* gauche-droit est choisi proportionnellement à la durée moyenne de l'unité acoustique qu'il représente. Pour modéliser un phonème, dont la réalisation acoustique est de courte durée, un *HMM* à trois états de type Bakis est habituellement utilisé. Pour une vue d'ensemble d'autres topologies, on pourra consulter [73].

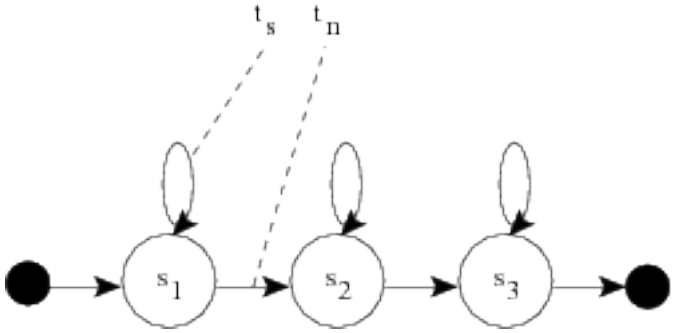


FIGURE 1.5 – Modèle type d'un *HMM* acoustique à 3 états selon R. Bakis

Les états représentés par des points noirs sont des états fictifs, qui ne génèrent aucune observation. Ils sont utilisés lors de la concaténation de *HMM* pour former des unités acoustiques plus longues. Cette concaténation de *HMMs* est nécessaire pour reconnaître la phrase prononcée, comme nous le verrons dans la section consacrée à la stratégie de reconnaissance employée dans un *SRAP* (paragraphe 1.4).

1.3.3 Modèles de langage

Les modèles de langage permettent de caractériser l'ensemble des phrases admissibles par un système de reconnaissance automatique de la parole.
Soit le second terme $p(M)$ de l'équation 1.3 qui représente la probabilité *a priori* de la séquence M de mots. La probabilité d'une phrase M constituée de N_M mots $M = m_1, m_2, \cdots, m_{N_M}$ peut être calculé comme un produit de probabilités conditionnelles [56] :

$$p(M) \quad = \quad p(m_1) \times \prod_{i=2}^{N_M} p(m_i / m_1 \cdots m_{i-1}) \tag{1.4}$$

En pratique, cette approche n'est cependant pas réalisable. Elle nécessiterait en effet d'un trop grand nombre de corpora textuels pour pouvoir réaliser un apprentissage fiable des probabilités de toutes les séquences M de mots possibles. Les ressources matérielles informatiques nécessaires seraient considérables. C'est pourquoi les systèmes de reconnaissance automatique de la parole actuels utilisent de manière pratiquement systématique les modèles probabilistes *n-grammes* : au lieu de considérer que la probabilité d'un mot m_i dépend de tous les mots précédents $m_1, m_2, \cdots, m_{i-1}$ à partir du début de la phrase, un *n-gramme* suppose que la probabilité de ce mot m_i ne dépend que des $n-1$ mots précédents $m_{i-n+1}, m_{i-n+2}, \cdots, m_{i-1}$.
Ainsi, dans le cas d'un trigramme où seuls les deux précédents mots sont pris en compte, l'équation 1.4 peut se réécrire :

$$p(M) = p(m_1) \times p(m_2/m_1) \times \prod_{i=3}^{N_M} p(m_i / m_{i-2}\, m_{i-1}) \tag{1.5}$$

De cette équation nous pouvons remarquer que l'utilisation de trigrammes nécessite l'estimation des probabilités des modèles unigrammes $p(m_i)$, bigrammes $p(m_i/m_{i-1})$ et trigrammes $p(m_i/m_{i-2}\, m_{i-1})$.
Pour un état de l'art sur les modèles de langage, les travaux de Brun [9] et de Langlois [68] pourront être consultés.

1.4 Stratégies de reconnaissance

Le but du moteur de reconnaissance (figure 1.3) est de fournir le texte $\hat{M}$ de la phrase qui correspond le mieux à la séquence d'observations acoustiques O, c'est-à-dire au signal acoustique S.

Tous les algorithmes de reconnaissance utilisés actuellement pour délivrer l'énoncé $\hat{M}$ construisent dans un premier temps un treillis de mots L_W à partir du modèle de langage. Ce treillis L_W représente l'ensemble des phrases (ou séquences de mots) pouvant être potentiellement reconnues par le système de reconnaissance automatique de la parole. Le dictionnaire de prononciation, les modèles acoustiques et le modèle de langage sont ensuite utilisés pour transformer ce treillis L_W en un treillis L_P. L_P représente alors le "métamodèle acoustique" du système de reconnaissance automatique de la parole, construit par concaténation des *HMMs* individuels en respectant les contraintes lexicales et linguistiques imposées par le lexique de prononciation et le modèle de langage. Il peut être symbolisé comme étant l'appareil de phonation d'un locuteur, permettant de prononcer l'ensemble G des phrases défini dans le *SRAP*.

Reconnaître la phrase qui a été prononcée, c'est-à-dire résoudre l'équation 1.3, revient alors à rechercher dans ce métamodèle L_P le chemin (la meilleure séquence d'états) de probabilité optimale $p(O|M)\ p(M)$ qui est susceptible d'avoir généré la séquence d'observations O. La séquence d'états permet ensuite de retrouver la séquence d'unités acoustiques et donc la séquence de mots de la phrase prononcée.

On distingue habituellement deux familles d'algorithmes de reconnaissance :
- les algorithmes de reconnaissance en une passe, qui considèrent que la meilleure séquence d'états déterminée à partir du treillis de phonèmes L_P représente l'énoncé $\hat{M}$ de la phrase prononcée.
- les algorithmes de reconnaissance en deux passes, qui recherchent les N meilleurs chemins (ou hypothèses) et qui déterminent l'énoncé $\hat{M}$ à partir de ces N hypothèses.

Une technique simpliste de reconnaissance en une passe consisterait à énumérer l'ensemble des chemins possibles, à calculer pour chacun d'eux sa probabilité, puis à considérer que le chemin possèdant la plus grande probabilité représente la séquence de mots délivrée par le système. Néanmoins, pour les systèmes de *RAP* actuels à grand vocabulaire, le nombre de chemins possibles à traiter demanderait un temps de calcul rédhibitoire.

Plusieurs algorithmes contemporains de reconnaissance permettent d'optimiser la recherche du chemin optimal. Le plus connu est l'algorithme de *Viterbi* [107; 33], qui emprunte le principe d'optimalité de Bellman afin de calculer par récurrence la probabilité associée aux chemins possibles. Pour limiter davantage l'espace de recherche et améliorer ainsi le temps en reconnaissance, l'algorithme de Viterbi a donné lieu à plusieurs variantes. L'une d'elles consiste à élaguer les chemins les moins prometteurs à l'aide d'un seuil. Si la probabilité associée à un chemin est inférieure à ce seuil, ce chemin n'est alors plus pris en compte dans la recherche. D'autres algorithmes de reconnaissance en une passe sont basés sur l'algorithme A^*. Ils utilisent une fonction heuristique pour ne parcourir que les chemins prometteurs et déterminer le meilleur chemin.

En ce qui concerne les algorithmes de reconnaissance en deux passes, la plupart sont également basés sur l'algorithme de Viterbi ou sur l'algorithme A^*. Une variante de l'algorithme de Viterbi, l'algorithme *N-Best* [94], peut effectivement être utilisé afin de déterminer non plus le meilleur chemin mais les N meilleurs chemins. Un algorithme de reconnaissance basé sur A^* peut ensuite être employé pour déterminer le meilleur chemin dans le treillis construit à partir des N meilleures hypothèses émises lors de la première passe.

Pour un état de l'art des algorithmes de reconnaissance, le lecteur intéressé pourra se reporter aux travaux de Lee *et al.* [73] et de Woszczyna [108].

1.5 Apprentissage des modèles acoustiques

Les performances d'un système de reconnaissance automatique de la parole sont conditionnées par la phase d'apprentissage des modèles acoustiques et du modèle de langage. Cet apprentissage nécessite des corpora textuels dans le cas du modèle de langage et des corpora de parole dans le cas des modèles acoustiques markoviens.

Nous n'aborderons ici que le cas de l'apprentissage des modèles acoustiques, l'apprentissage du modèle de langage sortant du cadre de cette thèse.

Soit $O = (o_1, o_2, \cdots, o_T)$ l'ensemble des T vecteurs d'observation (trames acoustiques) issus des données d'apprentissage. Soit $\Theta = \{\theta_1, \theta_2, \cdots, \theta_{N_H}\}$ l'ensemble des paramètres des *HMMs*, où θ_i représente les paramètres du i-ème *HMM*, pour $i = 1, 2, \cdots, N_H$. Ces paramètres incluent les probabilités de transition ainsi que les paramètres des densités de probabilité d'observations, c'est-à-dire les vecteurs de moyennes et les matrices de covariance des gaussiennes, et les poids associés aux gaussiennes dans les mélanges de gaussiennes (voir paragraphe 1.3.2.2).
L'apprentissage des paramètres Θ des *HMMs* est traditionnellement réalisé selon le critère du maximum de vraisemblance [6] des données d'apprentissage O, à l'aide de l'algorithme de Baum-Welch [7]. D'autres algorithmes peuvent néanmoins être employés pour l'apprentissage de *HMMs*, comme l'algorithme de Viterbi [33], dont l'utilisation pour l'apprentissage de modèles a été proposée dans [25].

6. ou *MLE* pour *Maximum Likelihood Estimation*

1.5.1 Apprentissage *MLE* à l'aide de l'algorithme de Baum-Welch

L'apprentissage au maximum de vraisemblance permet de déterminer les paramètres localement optimaux Θ^{ML} des modèles acoustiques Θ afin qu'il maximise la vraisemblance sur l'ensemble des observations O du corpus d'apprentissage :

$$\Theta^{ML} = \underset{\Theta}{\operatorname{argmax}} \; p(O/\Theta) \tag{1.6}$$

Les paramètres optimaux des modèles acoustiques sont obtenus de manière itérative avec l'algorithme de Baum-Welch. Les paramètres $\Theta^{(i)}$ de Θ à l'itération i sont réestimés à partir des paramètres du modèle $\Theta^{(i-1)}$ de l'itération précédente, de manière à ce que le nouveau modèle $\Theta^{(i)}$ améliore la vraisemblance sur l'ensemble des observations acoustiques O du corpus, c'est-à-dire tel que :

$$p(O/\Theta^{(i)}) \geq p(O/\Theta^{(i-1)}) \tag{1.7}$$

L'algorithme s'arrête lorsqu'un optimum local est atteint, c'est-à-dire lorsque :

$$p(O/\Theta^{(i)}) = p(O/\Theta^{(i-1)}) \tag{1.8}$$

Toutes les formules de réestimation des paramètres des modèles acoustiques à l'aide de l'algorithme de Baum-Welch sont dérivées dans l'annexe A.

1.5.2 Propriétés de l'apprentissage *MLE*

La phase préliminaire de l'apprentissage des paramètres des *HMMs* consiste en une étape (parfois fastidieuse) d'**étiquetage** du corpus d'apprentissage. L'étiquetage revient à indiquer pour chaque signal de parole du corpus quelles sont ses transcriptions exactes en terme d'unités acoustiques. Il permet d'effectuer un alignement correct des trames acoustiques avec les unités acoustiques, afin de réaliser une estimation cohérente des paramètres de chacun des *HMMs*. L'alignement revient alors à constituer l'ensemble A tel que :

$$A = \{A_1, A_2, \cdots, A_{N_H}\} \tag{1.9}$$

Chaque sous-ensemble A_i, pour $i = 1, 2, \cdots, N_H$, représente l'ensemble des vecteurs d'observation associés au *HMM* θ_i et permet d'estimer les paramètres de θ_i au maximum de vraisemblance. En utilisant cet ensemble A, l'estimation des modèles Θ est réalisée en reformulant l'équation 1.6 telle que :

$$\Theta^{ML} = \underset{\Theta}{\text{argmax}} \; p(O/\Theta) = \underset{\Theta}{\text{argmax}} \; p(A/\Theta) \tag{1.10}$$

L'estimation au maximum de vraisemblance des paramètres d'un modèle particulier θ_i est alors donnée par la formule :

$$\theta_i^{ML} = \underset{\theta_i}{\text{argmax}} \; p(A_i/\theta_i) \tag{1.11}$$

Cette équation nous renseigne qu'en l'absence de données A_i, il est impossible d'estimer les paramètres de θ_i en utilisant le critère du maximum de vraisemblance.

Ce problème d'estimation en l'absence de données, plus connu sous le terme de **problème des données manquantes**, s'applique également dans le cas où d'autres critères statistiques d'estimation (comme le critère du maximum *a posteriori* par exemple) sont employés. Ce point est essentiel et conditionne ainsi la difficulté de la tâche d'apprentissage. Une quantité suffisante de données d'apprentissage A_i est donc nécessaire pour apprendre de manière robuste les paramètres de chaque modèle acoustique θ_i. Cette quantité est proportionnelle au nombre de paramètres à estimer. Etant donné que les modèles acoustiques des *SRAP* actuels comportent plusieurs milliers de paramètres, une quantité considérable de données acoustiques se révèle ainsi indispensable pour leur apprentissage afin d'éviter le problème des données manquantes.

1.6 Système indépendant du locuteur et système dépendant du locuteur

L'apprentissage des modèles acoustiques peut être réalisé en utilisant soit la totalité des phrases prononcées par l'ensemble des locuteurs d'apprentissage, soit l'ensemble des phrases prononcées par un seul locuteur (figure 1.6).

Dans le premier cas, le système de *RAP* est dit indépendant du locuteur. Etant donné qu'un système indépendant du locuteur a été appris à l'aide de phrases provenant de plusieurs locuteurs, il est donc capable de prendre en compte de manière plus ou moins robuste la variabilité inter-locuteurs. Dans le cas où plusieurs phrases sont disponibles pour chaque locuteur d'apprentissage, la variabilité intra-locuteur peut également être prise en compte de manière assez fidèle. Un tel système fournit d'assez bonnes performances si le locuteur qui l'utilise possède des caractéristiques acoustiques assez proches de celles des locuteurs d'apprentissage. Pour certains locuteurs, appelés **locuteurs témoins** [7], les performances en reconnaissance du système peuvent néanmoins se dégrader de manière significative.

7. outlier speaker

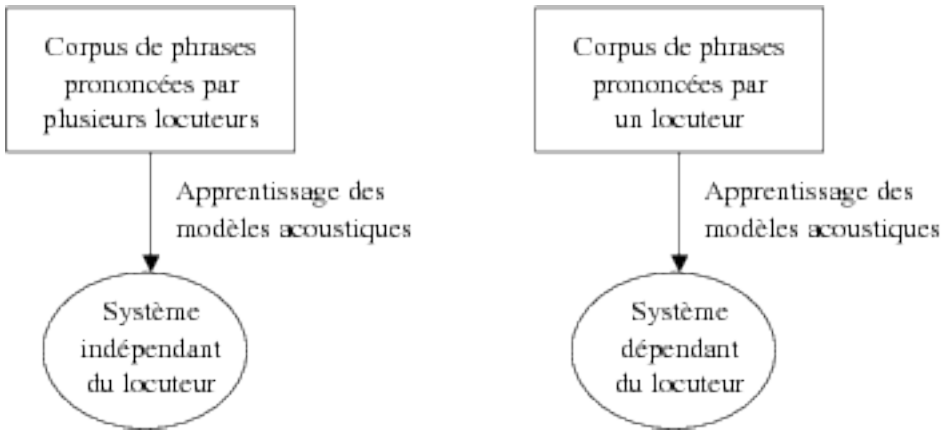


FIGURE 1.6 – Système indépendant du locuteur et système dépendant du locuteur

La quantité colossale de données nécessaire à l'apprentissage d'un système indépendant du locuteur ne facilite généralement pas sa mise en œuvre.

Dans le cas où seules les phrases prononcées par un locuteur sont employées pour l'apprentissage, le système de *RAP* généré, appelé système dépendant du locuteur, est alors particulièrement performant pour reconnaître les phrases émises par ce locuteur. La variabilité intra-locuteur est en effet prise en compte de manière plus fidèle que dans un système indépendant du locuteur. La reconnaissance vocale d'autres locuteurs est par contre susceptible d'être désastreuse.
Comme pour un système indépendant du locuteur, la difficulté de mettre en œuvre un système dépendant du locuteur réside dans la constitution d'un corpus d'apprentissage contenant un assez grand nombre de phrases (au moins plusieurs centaines). Dans le cas de l'apprentissage d'un système dépendant du locuteur, ces phrases doivent en outre être prononcée par le même locuteur.

L'utilisation d'un système indépendant du locuteur ou d'un système dépendant du locuteur, et donc la constitution du corpus de phrases d'apprentissage, dépend de la taille du panel de locuteurs susceptibles d'exploiter le système de *RAP*. Si le système de *RAP* n'est utilisé que par un nombre restreint de N locuteurs connus, l'apprentissage de N systèmes dépendant du locuteur peut dans ce cas être envisageable. L'identification du locuteur permet alors de choisir parmi les N systèmes de *RAP* lequel est utilisé. Dans les autres cas, la plupart des applications récentes en reconnaissance automatique de la parole utilisent des systèmes indépendant du locuteur, qui sont beaucoup plus flexibles que les systèmes dépendant du locuteur.

1.7 Conclusions

Nous avons présenté dans ce chapitre les modules [8], les composants [9] ainsi que les algorithmes d'apprentissage et de reconnaissance utilisés par la plupart des systèmes de *RAP* actuels pour faire face à la variabilité intrinsèque (redondance, phénomènes de coarticulation) et à la variabilité extrinsèque (due à l'environnement, au locuteur ou au canal de transmission) du signal de parole.

L'architecture générale du système de reconnaissance automatique de la parole que nous vous avons décrit correspond à un *SRAP* à petit vocabulaire. Nous avons choisi d'en donner une description assez détaillée étant donné que nous avons utilisé un système analogue pendant cette thèse (*ESPERE* [32]).

La difficulté de concevoir et de mettre en œuvre un *SRAP* est apparue en filigrane tout au long de ce chapitre. Le choix des unités acoustiques, des mots à inclure dans le vocabulaire, des transcriptions à associer à ces mots, ainsi que l'apprentissage des modèles acoustiques et du modèle de langage conditionnent les circonstances pour lesquelles le système de reconnaissance automatique de la parole délivrera ses meilleures performances. Par exemple, les performances d'un *SRAP* appris à l'aide de phrases prononcées uniquement par des femmes seront effectivement désastreuses lorsqu'un homme tentera de l'utiliser.

A cet égard, les techniques d'adaptation permettent de minimiser ces différences acoustiques entre les conditions d'apprentissage et les conditions d'utilisation d'un *SRAP*, pour lui permettre de délivrer des performances acceptables quels que soient les changements acoustiques survenus au cours de son exploitation. Avant d'étudier de telles techniques, il est important de comprendre les principes généraux et les bases théoriques qui sont employés dans le cadre de l'adaptation appliquée à la reconnaissance automatique de la parole. C'est ce que se propose de décrire le chapitre suivant.

8. le module de numérisation, le module de paramétrisation et le moteur de reconnaissance
9. le vocabulaire, les modèles acoustiques et le modèle de langage

Chapitre 2

Adaptation au locuteur des modèles acoustiques pour la *RAP*

Nous avons décrit dans le premier chapitre les divers composants qui sont employés par le moteur d'un système de reconnaissance automatique de la parole pour déterminer l'énoncé de la phrase qui a été prononcée par un locuteur. Ces composants sont le module de paramétrisation, les modèles acoustiques, le modèle de langage ainsi que le dictionnaire de prononciation. Les performances en reconnaissance du *SRAP* dépendent étroitement des connaissances utilisées pour générer le dictionnaire de prononciation, du choix de la paramétrisation du signal de parole, ainsi que des corpora textuels et acoustiques employés pour l'apprentissage du modèle de langage et des modèles acoustiques.
En effet, lorsqu'un système de reconnaissance automatique de la parole est utilisé dans les mêmes conditions que celles employées dans sa phase d'apprentissage (même environnement, même microphone, même locuteur), ses performances en terme de taux de reconnaissance dépassent les 95% de mots correctement reconnus. Mais qu'une condition vienne à changer, et la robustesse du système est immédiatement mise en défaut.

Comme tout système (artificiel ou biologique) qui a appris à exhiber certains types de comportements dans des situations spécifiques et qui adapte ses comportements face à une situation nouvelle, un système de reconnaissance automatique de la parole doit donc pouvoir s'adapter à de nouvelles conditions d'exploitation. L'adaptation dans un *SRAP* consiste à utiliser quelques phrases prononcées par un locuteur afin de modifier les paramètres de certains de ses composants internes et pallier ainsi aux différences acoustiques ou linguistiques survenues au cours de son utilisation. La fonction primordiale de l'adaptation dans un *SRAP* est de lui permettre de délivrer des performances acceptables quelles que soient les conditions dans lesquelles il est utilisé.

L'objectif de ce chapitre est principalement destiné à fournir les bases nécessaires à la compréhension des modèles mathématiques et des principes formulés dans le cadre de l'adaptation des modèles acoustiques. La première partie souligne la distinction qui est effectuée dans le domaine de l'adaptation pour la *RAP* entre l'adaptation au locuteur et l'adaptation à l'environnement. Nous indiquons ensuite où se situe la phase d'adaptation dans l'architecture d'un *SRAP*, en précisant notamment quels sont les composants

du *SRAP* qui peuvent faire l'objet d'une adaptation. Le cadre de travail mathématique et les concepts sous-jacents à l'adaptation des modèles acoustiques sont donnés dans la troisième partie. La quatrième section a pour but de préciser les différents modes possibles d'adaptation. Nous discutons enfin dans la cinquième et dans la sixième section des techniques qui font partie de l'état de l'art de l'adaptation au locuteur des modèles acoustiques.

2.1 Adaptation au locuteur et adaptation à l'environnement

Du point de vue d'un système de reconnaissance automatique de la parole, les principales sources de variabilité qui influent sur la complexité du signal acoustique sont dues à l'environnement ou au locuteur (voir paragraphe 1.1.3 et paragraphe 1.1.4, page 11). Les sources de variabilité dues au canal de transmission peuvent effectivement être omises si l'on considère que l'acquisition du signal s'opère selon les mêmes conditions lors de la phase d'apprentissage et lors de la phase d'utilisation du système.
Les variabilités dues au locuteur et à l'environnement devraient donc être conjointement prises en compte lors de la conception et de la mise en œuvre d'une technique d'adaptation. Cependant, plutôt que d'affronter simultanément toutes ces difficultés, les techniques d'adaptation actuelles tentent en général de pallier aux différences acoustiques engendrées préférentiellement par l'une de ces deux sources de variabilité. Pour cette raison, l'état de l'art relatif aux techniques d'adaptation pour la *RAP* distingue habituellement les techniques d'adaptation au locuteur des techniques d'adaptation à l'environnement.

2.2 Introduction de la phase d'adaptation dans l'architecture d'un *SRAP*

Afin de pallier les différences acoustiques ou linguistiques survenues au cours de l'utilisation du *SRAP* et causées par un changement des caractéristiques de l'environnement ou de celles du locuteur, quelques composants ou l'ensemble de ses composants peuvent faire l'objet d'une adaptation (figure 2.1).

L'adaptation d'un *SRAP* est réalisée en utilisant un corpus (aussi petit que possible) de phrases d'adaptation qui ont été prononcées par le locuteur utilisant actuellement le système. Ce corpus d'adaptation peut subir un prétraitement idoine afin de faciliter son exploitation pour la modification des paramètres d'un composant spécifique du *SRAP*.

L'architecture d'un système de reconnaissance automatique de la parole est telle que chaque composant modélise une partie des connaissances ou des informations qui entrent en jeu dans le processus de reconnaissance d'une phrase. Pour améliorer la robustesse d'un *SRAP* qui est confronté à de nouvelles conditions acoustiques ou linguistiques, chacun de ses composants est donc suceptible d'être adapté. Toutefois, selon que l'adaptation est

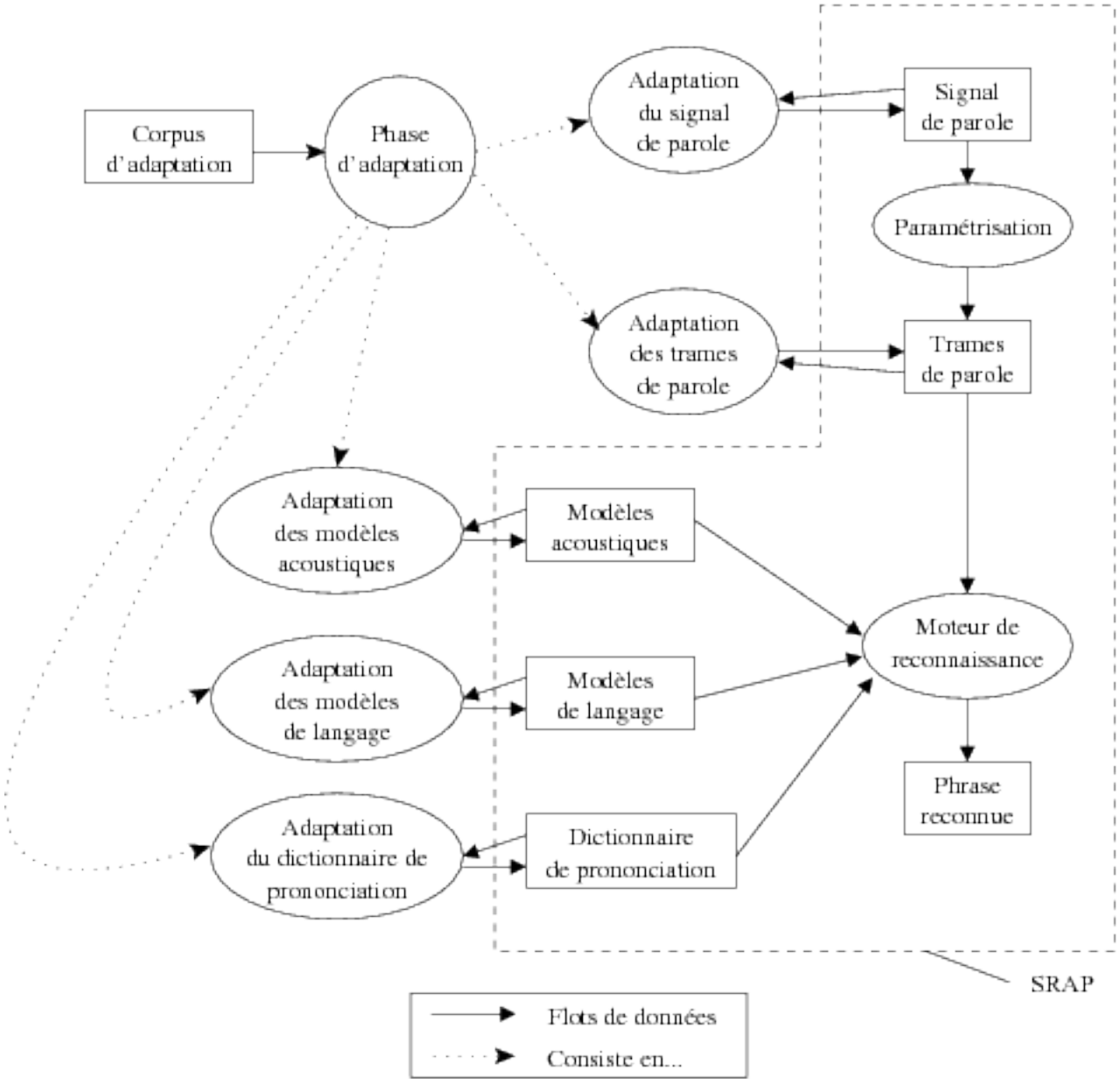


FIGURE 2.1 – Introduction de la phase d'adaptation dans l'architecture d'un *SRAP*

destinée à pallier des changements dues au locuteur ou à l'environnement, il sera plus judicieux d'adapter certains d'entre eux et de laisser certains autres inchangés, comme il est indiqué dans le tableau 2.1 :

Traitement éventuel	Adaptation au locuteur	Adaptation à l'environnement
Signal de parole		×
Trames acoustiques	×	×
Modèles acoustiques	×	×
Modèles de langage	×	
Dictionnaire de prononciation	×	

TABLE 2.1 – Composants susceptibles d'être modifiés dans le cas d'une adaptation au locuteur et dans le cas d'une adaptation à l'environnement

Dans le cas de l'adaptation à l'environnement, même s'il est mathématiquement possible de manipuler directement le signal de parole afin de le débarasser des effets nuisibles causés par les bruits ambiants et masquant le signal de parole, la majorité des techniques actuelles interviennent plutôt au niveau des trames acoustiques ou des modèles acoustiques. Il est en effet plus facile de concevoir et de mettre en œuvre ces techniques en manipulant les trames ou les modèles acoustiques, car l'espace de représentation est relativement réduit par rapport au domaine temporel du signal brut.

Dans le cas d'une adaptation au locuteur, la représentation moins complexe des modèles acoustiques ainsi que des trames acoustiques en font également les meilleurs candidats à une adaptation. Les modèles de langage ainsi que le dictionnaire de prononciation, parce qu'ils caractérisent les aspects linguistiques de la reconnaissance automatique de la parole et qu'ils représentent donc une source systématique et non négligeable de variabilité inter- et intra-locuteurs, sont à juste titre destinés éventuellement à être adaptés également. Il existe cependant très peu de méthodes d'adaptation au locuteur où l'adaptation porte sur le dictionnaire de prononciation ou sur les modèles de langage. L'hypothèse selon laquelle les variations systématiques dans la prononciation entre locuteurs peuvent être capturées par les modèles acoustiques pousse de nombreux chercheurs à s'orienter plutôt vers des techniques d'adaptation des modèles acoustiques. Il paraît alors judicieux d'adapter ces composants (dictionnaire de prononciation et modèles de langage) uniquement dans le cas où il existe de fortes probabilités pour que le système soit utilisé par une population de locuteurs ayant acquis des habitudes linguistiques différentes de celles modélisées dans le dictionnaire de prononciation et dans les modèles de langage actuellement utilisés par le *SRAP*.

La grande majorité des techniques actuelles d'adaptation au locuteur porte sur l'adaptation des modèles acoustiques uniquement. Il existe en effet peu de techniques qui s'orientent vers l'adaptation de deux composants ou plus. Parmi les plus connues, on peut citer [51; 91; 109; 105]. La première méthode [51] a montré qu'il est bénéfique de modifier

le dictionnaire de prononciation conjointement à une adaptation des modèles acoustiques lorsqu'il existe une différence de prononciation très sévère entre la phase d'apprentissage et la phase d'utilisation du système. Ceci peut être le cas si des locuteurs non natifs ou possédant un accent régional particulier utilisent le système qui a été appris pour des locuteurs natifs. Les trois dernières techniques [91; 109; 105] ont montré expérimentalement que la combinaison d'une technique de normalisation du locuteur (VTLN [31; 74; 61] en l'occurence) et d'une technique d'adaptation des modèles acoustiques (*MLLR*), pouvait avoir des effets bénéfiques sur les performances d'un *SRAP*.

2.3 Adaptation au locuteur des modèles acoustiques

Dans le cadre de cette thèse, nous nous sommes intéressés essentiellement à l'adaptation au locuteur des modèles acoustiques d'un système de reconnaissance automatique de la parole indépendant du locuteur, lorsque les modèles acoustiques sont représentés par des modèles de Markov cachés (*HMMs*).

Le problème de l'adaptation des *HMMs* dans un *SRAP* indépendant du locuteur est typiquement un problème d'estimation de paramètres. L'adaptation des *HMMs* consiste en effet à modifer la valeur de l'ensemble ou d'un sous-ensemble des paramètres des *HMMs*[10] afin d'améliorer les performances du système pour un nouveau locuteur. A cet égard, la plupart des principes valables dans le cadre de l'apprentissage des modèles acoustiques s'appliquent également au problème de l'adaptation de ces modèles. Cette affirmation peut être étayée par le fait que les mêmes algorithmes d'estimation des paramètres, basés sur l'algorithme *EM*, sont habituellement employés aussi bien dans le cadre de l'apprentissage que de l'adaptation des *HMMs*.

L'un de ces principes majeurs à garder à l'esprit lorsque l'on s'attaque au problème de l'adaptation de *HMMs* est qu'un modèle avec beaucoup de paramètres nécessite une quantité importante de données d'adaptation pour espérer obtenir des estimations robustes de ces paramètres. A l'inverse, une quantité faible de données d'adaptation peut être utilisée pour estimer de manière robuste les paramètres d'un modèle qui en comporte très peu. Dans ce sens, il est illusoire de penser qu'une petite quantité de données d'adaptation puisse être utilisée pour fournir des estimations précises d'un grand nombre de paramètres d'un modèle. Pour cette raison, les techniques actuelles d'adaptation tentent rarement d'estimer l'ensemble des paramètres des modèles acoustiques.

En effet, la majorité des techniques actuelles d'adaptation des modèles acoustiques supposent que les différences acoustiques entre les conditions d'apprentissage et les conditions d'utilisation d'un *SRAP* peuvent être réduites en manipulant seulement les moyennes des gaussiennes des modèles acoustiques. Cette hypothèse permet ainsi de limiter la mise à jour des paramètres des modèles acoustiques aux seules moyennes des gaussiennes, ce

10. Ces paramètres sont les probabilités de transitions entre états des *HMMs*, les moyennes et les variances-covariances des gaussiennes (modélisant la densité de probabilité d'observation associée à un état), et les coefficients associés aux gaussiennes dans un mélange de gaussiennes

qui réduit considérablement la quantité de données d'adaptation requise pour estimer ces paramètres de manière robuste.

En outre, plutôt que d'estimer directement les moyennes des gaussiennes (qui sont les paramètres à adapter), les techniques actuelles utilisent plutôt des ensembles de *variables d'adaptation*. Ces variables sont estimées à l'aide d'un critère statistique (le plus souvent au maximum de vraisemblance ou au maximum *a posteriori*) en utilisant les données d'adaptation. Elles sont ensuite employées, éventuellement avec d'autres informations (relatives à la variabilité due au locuteur par exemple), pour obtenir les moyennes des gaussiennes du système adapté.

Nous allons par la suite tenter d'exprimer ce concept de manière plus générale et plus formelle, en supposant que tous les paramètres (pas seulement les moyennes des gaussiennes) des modèles acoustiques peuvent être adaptés.

2.3.1 Notation utilisée

Avant d'aborder les modèles mathématiques utilisés dans le cadre de l'adaptation des modèles acoustiques d'un *SRAP*, il est indispensable de définir la topologie des modèles acoustiques utilisés ainsi qu'une notation des paramètres qui les constituent. Pour toute la suite de ce manuscrit, nous employerons les symboles suivants :

- $\Theta = \{\theta_1, \theta_2, \cdots, \theta_{N_H}\}$ est l'ensemble des N_H *HMMs* acoustiques θ_i utilisés par le *SRAP*.
- N_i est le nombre total d'états du *HMM* θ_i.
- $b_j^{(i)}(o)$ est la fonction de densité de probabilité d'observation o associée à l'état j du *HMM* θ_i.
  $b_j^{(i)}(o)$ est représentée par un mélange de N_P lois gaussiennes telle que :

$$\begin{aligned} b_j^{(i)}(o) &= \sum_{k=1}^{N_P} c_{j,k}^{(i)}\, \mathcal{N}(o, \mu_{j,k}^{(i)}, \sigma_{j,k}^{(i)}) \\ &= \sum_{k=1}^{N_P} c_{j,k}^{(i)}\, \frac{1}{\sqrt{(2\pi)^{N_D}\ |\sigma_{j,k}^{(i)}|}}\ exp\ \left[-\frac{1}{2}\ (o-\mu_{j,k}^{(i)})'\ {\sigma_{j,k}^{(i)}}^{-1}\ (o-\mu_{j,k}^{(i)})\right] \end{aligned} \quad (2.1)$$

- $c_{j,k}^{(i)}$ est le facteur de pondération de la k-ème loi gaussienne associée à l'état j du *HMM* θ_i.
- $\mu_{j,k}^{(i)}$ est le vecteur de moyenne de la k-ème gaussienne de l'état j du *HMM* θ_i.
- $\sigma_{j,k}^{(i)}$ est la matrice de variances-covariances de la k-ème gaussienne de l'état j du *HMM* θ_i.
- N_D est la taille d'un vecteur d'observation o.
- $g_{j,k}^{(i)} = (\mu_{j,k}^{(i)}, \sigma_{j,k}^{(i)})$ est la k-ème gaussienne de l'état j du *HMM* θ_i.
- N_S est le nombre total d'états de tous les *HMMs*, tel que $N_S = \sum_{i=1}^{N_H}\ N_i$.
- N_G est le nombre total de gaussiennes utilisées par le système de *RAP*, tel que $N_G = N_S \times N_P$.

Pour ne pas compliquer les notations, nous avons donc supposé que tous les états de tous les *HMMs* ont le même nombre de gaussiennes.

D'autre part, afin de ne pas alourdir les équations qui employeront ces symboles, l'écriture de chaque symbole s de la forme $s_j^{(i)}$ ou de la forme $s_{j,k}^{(i)}$, où i fait référence à un *HMM*, j à un état de ce *HMM* et k à une gaussienne de cet état de ce *HMM*, peut être simplifiée. Définissons une fonction f_1 telle que :

$$f_1(i,j) = \sum_{l=1}^{i-1} N_l \; + \; j \tag{2.2}$$

où $i \in [1; N_H]$, $j \in [1; N_i]$ et $f_1(.) \in [1; N_S]$. En utilisant cette fonction, un symbole s de la forme $s_j^{(i)}$ peut être renommé en un symbole de la forme s_x où $x = f_1(i,j)$. De la même manière, un symbole de la forme $s_{j,k}^{(i)}$ peut être renommé en un symbole de la forme $s_{x,k}$ où $x = f_1(i,j)$.
En définissant une fonction f_2 telle que :

$$f_2(i,j,k) = \sum_{l=1}^{i-1} N_l \, N_P \; + \; \sum_{m=1}^{j-1} N_P \; + \; k \tag{2.3}$$

où $i \in [1; N_H]$, $j \in [1; N_i] \quad \forall i \in [1; N_H]$, $k \in [1; N_P]$ et $f_2(.) \in [1; N_G]$, nous pouvons également simplifier la numérotation des symboles faisant référence à une gaussienne. Dans ce cas, les symboles de la forme $s_{j,k}^{(i)}$ peuvent alors être renommés en des symboles de la forme s_y où $y = f_2(i,j,k)$.
Cette notation permet ainsi de faire référence à une gaussienne particulière d'un état d'un *HMM* en utilisant uniquement un indice. Pour retrouver le numéro de la gaussienne r dans un certain état, et le numéro de l'état [11] dans lequel elle se trouve, nous utiliserons respectivement les deux fonctions f_g et f_s telles que :

$$f_g(r) \;=\; (r \;\%\; N_P) \tag{2.4}$$

$$\text{et } f_s(r) \;=\; (r \;/\; N_P) \tag{2.5}$$

où $a \;\%\; b$ est le reste de la division de a par b.

2.3.2 Représentation mathématique d'une technique d'adaptation

Comme nous l'avons évoqué précédemment, toute technique d'adaptation des modèles acoustiques manipule un ensemble de variables d'adaptation, qui sont estimées à l'aide

11. Ce numéro ne correspond pas au numéro de l'état d'un *HMM* dans lequel est située physiquement une gaussienne, mais le numéro de l'état parmi la totalité des N_S états des *HMMs*.

d'une méthode statistique puis utilisées pour mettre à jour les paramètres des modèles acoustiques du système indépendant du locuteur afin d'obtenir ceux du système adapté. Formellement, une technique d'adaptation peut être représentée par le triplet (Λ, ψ, Φ). Λ représente l'ensemble des variables d'adaptation. ψ indique la méthode qui est utilisée pour estimer les variables d'adaptation à l'aide des données d'adaptation. Φ est la fonction d'application de ces variables d'adaptation sur les paramètres des modèles acoustiques du système indépendant du locuteur afin d'obtenir les paramètres Θ correspondant des modèles acoustiques du système adapté, tels que :

$$\Theta = \Phi(\Lambda) \tag{2.6}$$

En utilisant ce formalisme, l'estimée au maximum de vraisemblance Λ^{ML} de Λ peut être obtenue à partir des données d'adaptation O selon la formule suivante :

$$\Lambda^{ML} = \underset{\Lambda}{\operatorname{argmax}}\ p(O|\Theta) = \underset{\Lambda}{\operatorname{argmax}}\ p(O|\Phi(\Lambda)) \tag{2.7}$$

Dans ce cas $\psi = ML$.
De la même manière, si le critère du maximum *a posteriori* est plutôt utilisé pour estimer les paramètres Λ ($\psi = MAP$), alors la forme générale de l'estimée au Λ^{MAP} de Λ peut être exprimée par :

$$\Lambda^{MAP} = \underset{\Lambda}{\operatorname{argmax}}\ p(O|\Theta)\, p(\Lambda) = \underset{\Lambda}{\operatorname{argmax}}\ p(O|\Phi(\Lambda))\, p(\Lambda) \tag{2.8}$$

2.3.2.1 Technique d'adaptation fortement contrainte

La formule générale 2.6 fait référence à une technique d'adaptation qui est fortement contrainte, dans le sens où les paramètres Θ des modèles acoustiques du système adapté sont tous obtenus de la même manière, en utilisant une seule fonction Φ pour tous les paramètres.

2.3.2.2 Technique d'adaptation faiblement contrainte

En supposant que Λ peut être décomposé en N sous-ensembles de variables d'adaptation tel que $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_N\}$, que Θ se décompose également en N sous-ensembles de paramètres $\Theta = \{\vartheta_1, \vartheta_2, \ldots, \vartheta_N\}$ et que Φ représente un ensemble de N fonctions telle que $\Phi = \{\phi_1, \phi_2, \ldots, \phi_N\}$, une technique d'adaptation moins contrainte peut être obtenue en considérant que :

$$\begin{cases} \vartheta_1 = \phi_1(\lambda_1) \\ \vartheta_2 = \phi_2(\lambda_2) \\ \vdots \\ \vartheta_N = \phi_N(\lambda_N) \end{cases} \tag{2.9}$$

Dans ce cas, chaque sous-ensemble λ_i est estimé à partir des données d'adaptation O selon l'un des critères statistiques (au maximum de vraisemblance ou au maximum *a posteriori* par exemple). La fonction ϕ_i est ensuite appliquée à λ_i pour obtenir l'ensemble des paramètres correspondant ϑ_i du système adapté.

2.3.3 Complexité d'une technique d'adaptation

L'efficacité d'une technique d'adaptation dépend de la complexité de sa représentation mathématique. Cette complexité résulte du nombre de variables d'adaptation définies, de la manière dont elles sont estimées et de la manière dont elles sont utilisées pour obtenir les paramètres des modèles acoustiques du système adapté.

Trois problèmes doivent alors être résolus dans le but de concevoir une technique d'adaptation fiable :

1. Premièrement, il est préférable que le nombre de variables d'adaptation soit déterminé en fonction de la quantité de données d'adaptation disponibles, afin de bénéficier d'estimations robustes, fiables. Plus le nombre de données d'adaptation s'accroît, plus la taille de Λ pourra s'accroître en conséquence.
2. Deuxièmement, Λ devrait être défini avec une certaine granularité. Chaque sous-ensemble λ_i de Λ est utilisé pour la mise à jour de l'ensemble de paramètres ϑ_i des modèles acoustiques. Ces sous-ensembles devraient alors être assez nombreux pour permettre une plus grande liberté d'adaptation, tout en associant une quantité de données d'adaptation suffisamment importante pour estimer de manière robuste les variables d'adaptation qui y sont regroupées.
3. Troisièmement, il serait souhaitable que Λ puisse prendre en compte efficacement les variabilités possibles inter-locuteurs et intra-locuteur. Pour cela, chaque paramètre d'adaptation devrait coder une source systématique de variabilité due au locuteur (comme le genre du locuteur, sa vitesse d'élocution, etc.).

Un phénomène néfaste, bien connu dans le domaine de l'apprentissage, pourrait survenir dans le cas où les deux premiers problèmes sont mal résolus : celui du sur-apprentissage. Le phénomène de sur-apprentissage survient si deux conditions sont réunies :

- si la quantité de données d'adaptation disponibles permet à une technique d'adaptation d'estimer de manière robuste la totalité de ses variables d'adaptation,
- si la technique d'adaptation dispose de plusieurs ensembles de variables d'adaptation et que chacun d'eux est utilisé pour adapter une seule gaussienne ou un nombre restreint de gaussiennes.

Dans ce cas en effet, les données d'adaptation pourraît avoir été trop bien apprises par les modèles acoustiques du système adapté pour un locuteur spécifique, ce qui entrainerait une dégradation des performances lors de la reconnaissance d'une nouvelle phrase, même si elle est prononcée par le même locuteur.

Concevoir une technique d'adaptation nécessite la résolution (au moins partielle) des trois problèmes précités. Dans ce cas, une technique d'adaptation serait alors plus à même de remplir sa fonction ultime : celle de trouver les paramètres des modèles acoustiques du système dépendant du locuteur sous-jacent, à l'aide de très peu de données d'adaptation. Elle serait alors capable d'améliorer rapidement et efficacement les performances d'un *SRAP*.

2.4 Modes d'adaptation des modèles acoustiques

En fonction des besoins de la tâche demandée au système de reconnaissance automatique de la parole, l'adaptation du système pourra être réalisée selon différents modes. On distingue habituellement deux critères qui permettent de définir le mode dans lequel s'applique l'adaptation :

- le critère d'**adaptation supervisée** ou d'**adaptation non supervisée**,
- le critère d'**adaptation par lot**[12] ou d'**adaptation incrémentale**.

Dans le cas de l'adaptation supervisée, la transcription de chaque phrase du corpus d'adaptation est connue, ce qui n'est pas le cas pour une adaptation non supervisée. L'incertitude liée à la transcription de la phrase reconnue peut ainsi avoir des effets néfastes sur l'efficacité d'une adaptation non supervisée.

En mode par lot, l'adaptation du système est réalisée une fois qu'un certain nombre de phrases d'adaptation ont été prononcées. Une fois utilisé, ce lot de phrases est vidé de son contenu et une nouvelle étape de récolte de phrases d'adaptation est initiée. En mode incrémental, l'adaptation est réalisée chaque fois qu'une nouvelle phrase a été prononcée. L'adaptation incrémentale est donc un cas particulier de l'adaptation par lot : une technique d'adaptation par lot pourra ainsi également être utilisée en mode incrémentale. Toutefois, comme nous le verrons dans l'état de l'art au paragraphe 2.6 et plus en détail lorsque nous étudierons la technique *SMLLR* (chapitre 4, paragraphe 4.2.2, page 71), des algorithmes existent pour améliorer dans le mode incrémental les performances d'une technique d'adaptation conçue pour être utilisée normalement en mode par lot.

De ces quatres modes d'adaptation, à savoir l'adaptation par lot supervisée, l'adaptation par lot non supervisée, l'adaptation incrémentale supervisée et l'adaptation incrémentale non supervisée, la plus simple à réaliser est la première. L'adaptation par lot supervisée est utilisée pour les sytèmes de dictée vocale [53; 30], par exemple. L'investissement concédé par un locuteur pour fournir au système un corpus de plusieurs dizaines de phrases d'adaptation compensent en effet largement les nombreuses heures pendant lesquelles le système sera utilisé. D'autre part, il existe diverses applications pour lesquelles l'utilisateur n'interagit que pendant très peu de temps avec le système de reconnaissance automatique de la parole. C'est le cas par exemple pour les bornes interactives vocales de vente de billets ou les serveurs vocaux d'accès à des informations distantes. Pour ces applications,

12. Batch adaptation

le système devra s'adapter rapidement à ses nouvelles conditions d'utilisation en ayant alors recours à une adaptation incrémentale non supervisée, la plus compliquée à mettre en œuvre.

Dans le cadre de cette thèse nous nous sommes intéressés à l'adaptation par lot supervisée et à l'adaptation incrémentale non supervisée. Ces deux styles d'adaptation sont complémentaires et conviennent comme nous l'avons vu à une large panoplie d'applications.

2.5 Taxinomie des techniques d'adaptation des modèles acoustiques

Comme indiqué dans la figure 2.2, les techniques d'adaptation au locuteur des modèles acoustiques peuvent être classées selon cinq familles : la famille des techniques utilisant des transformations, la famille des techniques ayant recours au critère bayésien du Maximum A Posteriori, la famille des techniques utilisant des modèles prédictifs, la famille des techniques utilisant des informations relatives aux locuteurs et la famille des techniques hybrides, combinant les concepts de deux ou plusieurs des techniques précédentes d'adaptation des modèles acoustiques.

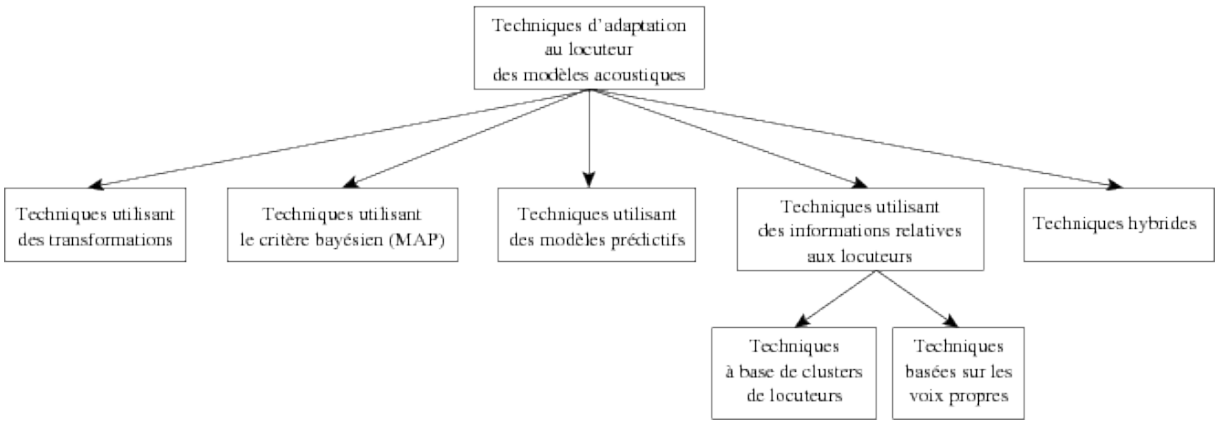


FIGURE 2.2 – Taxinomie des techniques d'adaptation des modèles acoustiques

Les techniques les plus représentatives de chaque famille figurent dans l'état de l'art qui suit.

2.6 Etat de l'art des méthodes d'adaptation des modèles acoustiques

Depuis une vingtaine d'années, de nombreux chercheurs proposent continuellement soit des nouvelles méthodes originales d'adaptation des modèles acoustiques, soit des

techniques qui améliorent l'efficacité des méthodes existantes. A cet égard, étant donné que plusieurs centaines de techniques existent actuellement, il serait inconcevable d'en donner une présentation approfondie pour chacune d'elle.
Nous nous contenterons donc de fournir ci-après une description détaillée uniquement pour les techniques qui nous ont semblées incontournables ou les plus originales.
En outre, les techniques d'adaptation des modèles acoustiques décrites ci-après sont destinées essentiellement à une adaptation des paramètres de *HMMs* à densités continues.

2.6.1 Techniques basées sur le critère du maximum *a posteriori*

Les techniques qui utilisent le critère du maximum *a posteriori* pour estimer les variables d'adaptation supposent qu'il existe une distribution *a priori* de ces paramètres. Lorsque le critère du maximum de vraisemblance est utilisé, les variables d'adaptation Λ sont déterminées de telle sorte à maximiser la vraisemblance $p(O|\Lambda)$ sur les données d'adaptation O. Dans le cas d'une estimation au maximum *a posteriori*, les variables d'adaptation Λ sont choisies comme étant celles qui maximisent la distribution *a posteriori* $p(O|\Lambda) \times p(\Lambda)$ sur les données d'adaptation O. $p(\Lambda)$ représente la distribution *a priori* qui modélise nos connaissances sur les valeurs possibles des variables d'adaptation.

L'utilisation d'une distribution *a priori* permet donc d'utiliser une quantité moins importante de données d'adaptation pour obtenir des estimations robustes des paramètres. C'est pour cette raison que ces méthodes figurent actuellement parmi celles qui sont les plus utilisées dans le domaine de l'adaptation des modèles acoustiques.

2.6.1.1 *Maximum A Posteriori*

Dans la version classique de *MAP* [42], le vecteur de moyenne $\hat{\mu}_r$ de chaque gaussienne g_r du système adapté est obtenu par interpolation entre le vecteur de moyenne $\bar{\mu}_r$ du *SIL* et l'estimation au maximum de vraisemblance de μ_r. L'estimée $\hat{\mu}_r$ au *MAP* de μ_r s'obtient alors selon la formule :

$$\hat{\mu}_r = \frac{\tau\ \bar{\mu}_r\ +\ \sum_{t=1}^{T}\ \gamma_t(r)\ o_t}{\tau\ +\ \sum_{t=1}^{T}\ \gamma_t(r)}$$

$\bar{\mu}_r$ est le vecteur de moyenne *a priori* de la gaussienne g_r (celui du *SIL*) ; o_t est l'un des T vecteurs d'observation au temps t ; $\gamma_t(r)$ est la probabilité *a posteriori* d'occuper la gaussienne g_r au temps t en sachant que la séquence d'observations O a été générée. τ est l'hyperparamètre qui détermine l'influence du vecteur de moyenne *a priori* sur l'estimation au maximum de vraisemblance de μ_r : $\sum_{t=1}^{T}\ \gamma_t(r)\ o_t$.
L'avantage majeur de cette technique est que l'estimation au *MAP* converge vers l'estimation au maximum de vraisemblance lorsque la quantité de données d'adaptation est très importante. Cette technique est donc capable d'estimer les paramètres des modèles acoustiques du système adapté de manière à ce qu'ils convergent vers les estimations des modèles acoustiques du système dépendant du locuteur lorsque une quantité importante de données d'adaptation est employée. Toutefois, étant donné que seules les gaussiennes pour lesquelles des données d'adaptation sont disponibles sont adaptées, cette convergence est plutôt lente. Dans le cas où les modèles acoustiques comportent un grand nombre de

gaussiennes, beaucoup d'entre elles ne disposent pas de données d'adaptation et ne sont donc pas adaptées. L'adaptation des quelques autres gaussiennes n'a ainsi qu'un impact mineur sur les performances du *SRAP* généré.

2.6.1.2 *Structural Maximum A Posteriori*

La technique *SMAP* présentée dans [96; 97; 98] permet d'accroître la rapidité d'adaptation par rapport à la version classique de *MAP* tout en conservant la propriété de convergence vers l'estimation au maximum de vraisemblance lorsque la quantité disponible de données d'adaptation est importante. L'approche adoptée par *SMAP* est la suivante. Toutes les gaussiennes des modèles acoustiques du système indépendant du locuteur sont organisées dans une structure arborescente, appelée arbre de gaussiennes. Chaque nœud feuille ne contient qu'une seule gaussienne. Dans le cas de l'adaptation des moyennes des gaussiennes, un vecteur de biais est récursivement estimé dans chaque nœud de l'arbre et pour chaque niveau, en commençant par la racine. A chaque niveau de l'arbre, l'estimée du vecteur de biais du nœud au niveau supérieur est utilisée comme distribution *a priori*. Ce vecteur de biais est également estimé dans chaque nœud feuille puis appliqué au vecteur de moyenne de la gaussienne qui y est présente pour obtenir la gaussienne correspondante du système adapté. Une description plus détaillé de cette technique est donnée dans le chapitre 5, étant donné qu'elle a fait l'objet d'une étude approfondie dans le cadre de cette thèse. Lorsque peu de données d'adaptation sont disponibles, cette technique permet d'obtenir une amélioration significative des performances par rapport à celle obtenue avec la méthode classique *MAP*.

2.6.2 Techniques utilisant des modèles prédictifs

Une approche ambitieuse pour adapter rapidement les paramètres des modèles acoustiques est de considérer que les paramètres mal estimés et les paramètres non adaptés peuvent être mis à jour à partir des paramètres adaptés, en utilisant un ou plusieurs modèles prédictifs. Un modèle prédictif est typiquement représenté par une matrice de corrélation. Celle-ci contient l'ensemble des corrélations qui ont été définies soit entre unités de parole, soit entre certains paramètres des modèles acoustiques d'un *SRAP*. Dans le dernier cas, si l'adaptation porte sur les moyennes des gaussiennes, ces corrélations sont alors définies entre les vecteurs de moyennes des gaussiennes. La recherche des corrélations est réalisée à partir d'un ensemble de systèmes dépendant du locuteur. Cette phase, initié avant le processus d'adaptation (*offline*), nécessite un grand nombre de systèmes afin de fournir des estimations robustes de ces corrélations.
Les techniques qui font partie de cette famille sont développées, entre autres, par Doh et Stern [29], Afify *et al.* [1], Ahadi et Woodland [2], Chen et DeSouza [16], Cox [21; 22] et Hazen [47; 48]. Nous présentons ci-après les deux techniques les plus connues de cette famille.

2.6.2.1 *Extended Maximum A Posteriori*

Dans *EMAP* [69], Lasry et Stern utilisent des corrélations entre des unités de parole. Une observation d'une unité acoustique U permet alors non seulement de modifier les moyennes des gaussiennes des modèles de cette unité, mais également celles des modèles des unités non observées U' qui sont corrélées avec U.
Cox [22] utilisa cette méthode pour une tâche de reconnaissance de l'alphabet. Avec trois phrases d'adaptation contenant chacune 13 lettres, le taux d'erreur passa de 17% pour les modèles indépendant du locuteur à 3% pour les modèles adaptés.

2.6.2.2 *Regression-Based Model Prediction*

L'approche *RMP* [2] utilise des corrélations définies entre les moyennes des gaussiennes des modèles acoustiques d'un *SRAP*. Ces corrélations (représentées sous la forme de matrices de régression linéaire) permettent de mettre à jour les moyennes non adaptées ou les moyennes mal adaptées sur la base des moyennes bien adaptées.
RMP adapte tout d'abord les moyennes des gaussiennes à l'aide de *MAP*. Les gaussiennes qui ont reçues une quantité suffisante de données d'adaptation sont ensuite utilisées pour estimer les moyennes mal adaptées (n'ayant pas reçues suffisamment de données) et les moyennes non observées.
RMP converge vers les performances de *MAP* lorsque la quantité de données d'adaptation est importante. Elle surpasse *MAP* dans le cas où peu de données d'adaptation sont disponibles. Par exemple, une réduction de 8% du taux d'erreur en mots est obtenu en utilisant *RMP* avec une phrase d'adaptation de l'ordre de 3 secondes [2]. *MAP* ne donne aucune amélioration dans ce cas, puisque seulement très peu des paramètres du système sont observés.

2.6.3 Techniques employant des transformations

L'idée maîtresse des techniques basées sur des transformations réside dans le fait que des gaussiennes similaires peuvent être adaptées de la même manière, c'est-à-dire en utilisant la même transformation. De cette manière, il est désormais possible d'adapter des modèles acoustiques pour lesquels aucune donnée d'adaptation n'a été observé. Considérons que $\mathcal{C} = \{g_1, g_2, \ldots, g_{N_G}\}$ représente l'ensemble des gaussiennes des modèles acoustiques du *SIL*. Les techniques à base de transformations supposent que cet ensemble $\mathcal{C}$ peut être divisé en N sous-ensembles tel que :

$$\mathcal{C} \quad = \quad \{C_1, C_2, \ldots, C_N\} \tag{2.10}$$

Chaque gaussienne est supposée n'appartenir qu'à une et une seule classe C_i, c'est-à-dire que :

$$C_1 \cup C_2 \cup \ldots \cup C_N = \mathcal{C} \text{ et } C_i \cap C_j = \emptyset \;\; \forall i \neq j \tag{2.11}$$

Chaque classe C_i regroupe un ensemble de gaussiennes similaires au sens d'une certaine mesure de distance (selon la distance de Mahalanobis ou la divergence de Kullback-Leibler par exemple). Pendant l'adaptation, chaque classe utilise un ensemble de variables d'adaptation. Elles constituent une transformation. Une transformation est partagée par l'ensemble des gaussiennes d'une classe et est utilisée pour mettre à jour les moyennes et/ou les variances-covariances de chaque gaussienne appartenant à cette classe. Le nombre N de classes, la répartition des gaussiennes dans ces classes, la structure des transformations employées ainsi que la méthode d'estimation des variables d'adaptation de ces transformations dépendent de la technique d'adaptation.

2.6.3.1 *Maximum Likelihood Linear Regression*

Maximum Likelihood Linear Regression (*MLLR*) est la technique la plus représentative de la famille des techniques employant des transformations. Elle fût proposé par Leggetter et Woodland [77; 79; 76]. Elle constitue actuellement la technique d'adaptation la plus largement utilisée et fait continuellement l'objet d'améliorations. Elle permet typiquement de réduire de 15% le taux d'erreur en mots d'un système indépendant du locuteur pour une tâche de *RAP* grand vocabulaire lorsqu'une minute de parole non bruitée est disponible pour l'adaptation. Lorsque près de trente minutes sont disponibles, des performances atteignant celles d'un système dépendant du locuteur sont envisageables.
Dans la version classique de *MLLR*, seuls les moyennes des gaussiennes des modèles acoustiques sont adaptées. Le nombre N de classes et la répartition des gaussiennes dans chaque classe sont déterminés avant le processus d'adaptation. Le vecteur de moyenne μ_r de chaque gaussienne g_r appartenant à la classe C_i, pour $i = 1, 2, \ldots, N$, est adapté selon la formule :

$$\hat{\mu}_r = A_i \, \mu_r \; + \; b_i$$

où A_i est une matrice de dimension $n \times n$ et b_i est un vecteur de biais de dimension n. A_i et b_i sont associés à la classe C_i. Habituellement, cette équation est réécrite sous la forme :

$$\hat{\mu}_r = W_i \, \xi_r$$

W_i est la matrice de transformation linéaire de dimension $n \times (n+1)$ associée à C_i et telle que $W_i = (A_i \; b_i)$. ξ_r est le vecteur de moyenne étendu tel que $\xi'_r = (\mu'_r \; 1)$. Dans *MLLR*, les paramètres d'une transformation W_i sont estimés au maximum de vraisemblance des données d'adaptation, à l'aide de l'algorithme *EM* (voir annexe **??**). Un vecteur de moyenne peut subir une rotation et une homothétie (à l'aide de la matrice A_i), ainsi qu'une translation (avec le vecteur b_i). Cette technique dispose donc d'un large potentiel d'adaptation lorsque la quantité de données d'adaptation est importante. Dans ce cas en effet, chaque gaussienne peut être adaptée en utilisant une transformation différente. A l'inverse, lorsque peu de données d'adaptation sont disponibles, une seule transformation peut être estimée et appliquée à chacune des gaussiennes du système indépendant du locuteur. Une adaptation rapide peut donc théoriquement être réalisée dans ce cas. Toutefois, étant donné que le nombre de classes (et donc le nombre de transformation à estimer) est

déterminé avant de connaître la quantité de données d'adaptation qui est disponible, les variables d'adaptation peuvent être mal estimées [70].

2.6.3.2 *Structural Maximum Likelihood Linear Regression*

La technique *Structural Maximum Likelihood Linear Regression* (*SMLLR*) proposée dans [78] permet de pallier ce problème. Cette technique utilise une structure arborescente afin de déterminer de manière dynamique, c'est-à-dire pendant le processus d'adaptation, en fonction de la quantité de données d'adaptation disponible, le nombre de classes à employer. Cet arbre regroupe dans chacun de ses nœuds un ensemble de gaussiennes qui sont acoustiquement similaires selon une certaine mesure de distance. *SMLLR* sera décrite plus en détail dans le chapitre 4, étant donné qu'elle a fait l'objet d'une étude approfondie dans le cadre de cette thèse. Dans la suite de ce chapitre, nous employerons le terme de *MLLR* pour désigner indifféremment les techniques *MLLR* et *SMLLR*.

2.6.3.3 Réduction du nombre de variables d'adaptation à estimer

La technique *MLLR* suggère qu'une adaptation rapide est possible. Cependant il a été montré dans [79; 109; 70] que *MLLR* n'est pas capable d'adapter efficacement un *SIL* lorsque le nombre de phrases d'adaptation est trop petit (moins de trois phrases). Ceci est du au trop grand nombre de paramètres à estimer lorsque la matrice W est pleine. Lorsque la quantité disponible de phrases d'adaptation est plus faible, des matrices de transformation diagonales peuvent être utilisées, bien que l'amélioration des performances dans ce cas reste très marginale.
En outre, l'utilisation de matrices blocs-diagonales permet d'atteindre approximativement les mêmes performances que lorsque des matrices pleines sont employées, en dépit du nombre plus restreint de variables d'adaptation à estimer [76]. Dans cette variante, les matrices de transformation W ont la forme :

$$W = \begin{bmatrix} W_1 & \cdots & \cdots & 0 \\ \vdots & W_2 & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & W_B \end{bmatrix}$$

Soit μ le vecteur de moyenne d'une gaussienne tel que :

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_{N_D} \end{bmatrix} = \begin{bmatrix} \nu_1 \\ \nu_2 \\ \vdots \\ \nu_B \end{bmatrix}$$

En faisant l'hypothèse que les N_D coefficients constitutifs de μ : μ_i $(i = 1, 2, \ldots, D)$ qui composent les B blocs $\nu_1, \nu_2, \cdots, \nu_B$ sont dépendants uniquement à l'intérieur d'un bloc, les B matrices $W_1, W_2, \cdots, W_B$ permettent alors respectivement d'adapter les blocs $\nu_1, \nu_2, \cdots, \nu_B$. Par exemple, dans le cas d'une matrice de transformation blocs-diagonales

composée de 3 blocs W_1, W_2 et W_3, W_1 pourrait être utilisé pour adapter les coefficients statiques des vecteurs de moyenne des gaussiennes, W_2 les dérivées premières (coefficients delta) et W_3 les dérivées secondes (coefficients delta-delta).
Alors qu'une matrice de transformation pleine de dimension $n \times (n+1)$ nécessite l'estimation de $n * (n+1)$ paramètres, une matrice bloc-diagonale constituée de 3 matrices pleines W_1, W_2, W_3, de dimension respective $n_1 \times (n_1+1), n_2 \times (n_2+1), n_3 \times (n_3+1)$ avec $n_1+n_2+n_3 = n$, ne permet plus que d'estimer $n_1 \times (n_1+1)+n_2 \times (n_2+1)+n_3 \times (n_3+1) << n \times (n+1)$. Une matrice de transformation de dimension 35×36 requiert l'estimation de 1260 paramètres, à comparer aux 444 paramètres devant être estimés dans le cas d'une matrice bloc-diagonale composée de 3 matrices respectivement de dimension 11×12, 12×13 et 12×13.
Plus récemment, Goel *et al.* [43] proposent d'utiliser des transformations qui sont des combinaisons linéaires de matrices de rang unitaire. Ils ont montré que, pour un même nombre de variables d'adaptation estimées, l'emploi de telles transformations pouvait fournir des performances significativement meilleures que lorsque des matrices diagonales ou blocs-diagonales sont utilisées.
Kim et Chung [62] proposent de réduire le nombre des variables d'adaptation constituant une matrice de régression linéaire, en utilisant une technique de réduction de dimension. Soit *PCA* (*Principal Component Analysis*), soit *ICA* (Independant Component Analysis) est appliquée aux modèles acoustiques du système indépendant du locuteur. Les modèles acoustiques obtenus sont constitués de moins de paramètres, ce qui permet alors de réduire le nombre de coefficient d'une matrice de régression linéaire. *PCA* et *ICA* ne permettent pas seulement de réduire le nombre de variables à estimer. Ils permettent également de rendre les paramètres plus discriminants. Lorsque peu de phrases d'adaptation sont utilisées (moins de quatre phrases), l'emploi de *PCA* ou de *ICA* permet ainsi d'obtenir des performances supérieures à celles fournies par la version classique de *MLLR*.

2.6.3.4 Estimation des matrices de variances-covariances des gaussiennes

La version classique de *MLLR* n'adapte que les moyennes des gaussiennes des modèles acoustiques. Seule la position des gaussiennes dans l'espace acoustique est donc modifiée. L'aspect des gaussiennes ne change donc pas. Dans le cas où l'on dispose d'une quantité importante de données d'adaptation (équivalent à une cinquantaine de phrases), les matrices de variances-covariances des gaussiennes peuvent également être adaptées en utilisant une transformation spécifique [39; 40; 37]. Ceci permet d'améliorer sensiblement les performances du système adapté par rapport à celles obtenues en adaptant uniquement les moyennes des gaussiennes.

2.6.3.5 *MLLR* contraint

La formulation *MLLR* décrite ci-dessus suppose que des matrices différentes sont estimées pour les vecteurs de moyenne et pour les matrices de variances-covariances des gaussiennes. L'estimation de ces transformations est dans ce cas non contrainte. Une estimation contrainte des transformations peut être réalisée [27; 37], afin de réduire le nombre de variables d'adaptation et donc la quantité de données d'adaptation requise pour une

estimation robuste de ces variables. Dans ce cas, le vecteur de moyenne μ et la matrice de variances-covariances σ d'une gaussienne sont adaptés en utilisant la même transformation de la manière suivante :

$$\hat{\mu} = A\,\mu\, - b$$

$$\hat{\sigma} = A'\,\sigma\,A$$

où A est la transformation à estimer. Dans [27], la matrice de transformation A est diagonale, tandis que dans [37] le cas de l'estimation d'une matrice de transformation pleine est présentée. L'estimation de la transformation A nécessite plusieurs itérations de *EM*. Cette variante de *MLLR* donne cependant des performances similaires à celles obtenues en utilisant des transformations non contraintes de même structure.

2.6.3.6 *Discounted Likelihood Linear Regression*

Byrne et Gunawardana ont montré dans [10; 11; 12] que *MLLR* a tendance à sur-apprendre les paramètres des transformations lorsque plusieurs itérations de *EM* sont employées alors que peu de phrases d'adaptation sont disponibles. Dans la méthode *DLLR* qu'ils proposent, une variante de *EM* est utilisée pour estimer les variables d'adaptation. Cette variante de *EM*, l'algorithme d'interpolation du moment, détermine les variables d'adaptation afin qu'elles maximisent une vraisemblance "réduite"[13] sur les données d'adaptation. Alors que les performances obtenues avec *MLLR* se dégradent lorsque plus d'une itération de *EM* est employée, *DLLR* permet de les améliorer légèrement à chaque fois qu'une nouvelle itération de la variante de *EM* est réalisée.

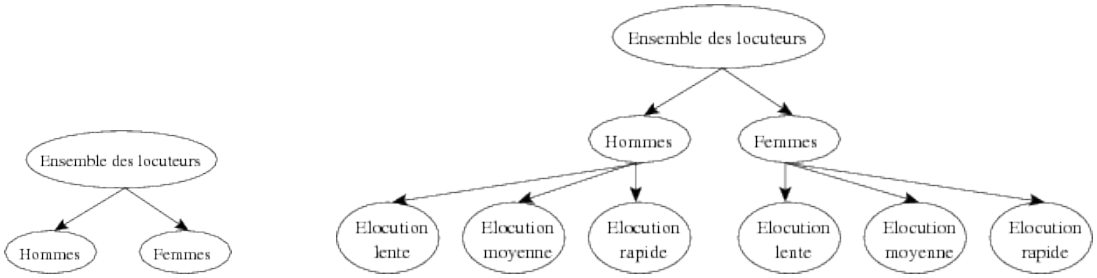
2.6.3.7 (*Structural*) *Maximum A Posteriori Linear Regression*

Dans [17; 18], les auteurs améliorent les performances de *MLLR* lorsque peu de phrases d'adaptation sont disponibles en utilisant une distribution *a priori* sur les paramètres des matrices de transformation. Les variables d'adaptation dans cette technique *MAPLR* ne sont donc plus estimées au maximum de vraisemblance, comme c'est le cas dans *MLLR*, mais au maximum *a posteriori*.
MAPLR peut encore être étendue et améliorée en utilisant une structure hiérarchique de distributions *a priori* du type de celle utilisée dans *SMAP*. La technique *SMAPLR* [101; 100] permet dans ce cas d'adapter un *SIL* en utilisant éventuellement plusieurs matrices de transformations qui sont estimées au maximum *a posteriori*.
Chou et He [19] adaptent également les variances des gaussiennes à l'aide de matrices de régression linéaire qui sont estimées au maximum *a posteriori*. L'adaptation *MAPLR* des variances des gaussiennes permet d'obtenir des résultats significativement supérieurs à ceux fournis par une adaptation *MLLR* des variances.

13. *discounted likelihood*

2.6.4 Techniques modélisant des caractéristiques relatives aux locuteurs

Aucune des techniques que nous avons décrit précédemment n'utilise d'informations explicites sur des caractéristiques spécifiques aux locuteurs. Toutes ces techniques peuvent donc être utilisées aussi bien pour une adaptation au locuteur que pour une adaptation à l'environnement [14]. A l'inverse, les techniques employant des classes de locuteurs et les techniques basées sur les *EigenVoices* (ou "voix propres") ne peuvent être utilisées que pour une adaptation au locuteur. Ces techniques sont effectivement capables de prendre en compte la variabilité inter-locuteurs et la variabilité intra-locuteur en utilisant un ensemble de systèmes de référence. Pour générer ces systèmes, les locuteurs du corpus d'apprentissage sont organisés en clusters (ou classes), dans une structure arborescente. Le nœud racine de cet arbre contient l'ensemble des locuteurs. Chaque nœud de l'arbre regroupe des locuteurs partageant les mêmes particularités acoustiques, au sens d'une certaine mesure de similarité. Deux exemples de regroupement de locuteurs sont donnés dans la figure 2.3, le regroupement de locuteurs le plus simple étant représenté par le regroupement par genre homme-femme.



Locuteurs classés par genre — Locuteurs classés par genre et par vitesse d'élocution

FIGURE 2.3 – Exemples de regroupement de locuteurs

Le système associé à chaque nœud est entrainé à l'aide des phrases prononcées par les locuteurs regroupés dans ce nœud. Chaque système de référence ainsi généré est supposé reconnaître un certain type de locuteur.
La plupart de ces techniques sont généralement très efficaces dans le cas d'une adaptation rapide.

2.6.4.1 Multi-modèles

L'approche des multi-modèles considère que la variabilité inter-locuteurs peut être plus fidèlement pris en compte en employant un ensemble de modèles acoustiques provenant de plusieurs systèmes de référence plutôt que d'un unique système indépendant du locuteur.

14. exceptées peut être les techniques employant des modèles prédictifs, dans le sens où les informations de corrélations entre paramètres sont introduites pour prendre en compte des caractéristiques propres aux locuteurs. Toutefois, on peut supposer que ces corrélations peuvent également être définies afin qu'elles puissent prendre en compte n'importe quel événement acoustique.

Le processus d'adaptation d'un système de *RAP* peut être considéré comme un processus de sélection parmi les systèmes de références du meilleur système de reconnaissance pour un nouveau locuteur. Dans ce cas, les modèles acoustiques de plusieurs systèmes sont utilisés dans le processus de reconnaissance d'une phrase prononcée par le nouveau locuteur. Ce sont alors les modèles acoustiques du système de référence ayant fourni la meilleure vraisemblance sur les données d'adaptation qui sont ensuite utilisés pour la phase de reconnaissance.
Cette approche des multi-modèles a été largement utilisée, notamment par Furui [34], Mathan et Miclet [85], Van Compernolle *et al.* [20], Ljolje [82], Jouvet *et al.* [59], Kosaka *et al.* [63; 64], Sankar *et al.* [93], Barras [6] et Padmanabhan *et al.* [90; 41]. Les techniques proposées dans ces articles se distinguent les unes des autres par la manière de constituer les clusters de locuteurs, d'apprendre les systèmes de référence et d'effectuer la reconnaissance.

2.6.4.2 *Cluster Adaptive Training*

L'apprentissage adaptatif par multi-modèles (ou *CAT*) [36; 38] considère l'existence d'un espace des locuteurs qui représente la variabilité inter-locuteurs. Cet espace des locuteurs est construit à partir d'un ensemble de systèmes de référence. Chaque locuteur est représenté dans cet espace par un vecteur de coordonnées. Ces coordonnées sont extraites à partir des paramètres des modèles acoustiques du système de référence correspondant à ce locuteur. L'apprentissage adaptatif par multi-modèles consiste alors à localiser le nouveau locuteur dans l'espace des locuteurs. Les coordonnées du nouveau locuteur permettent ensuite de générer le système adapté.
La localisation du nouveau locuteur consiste tout d'abord à choisir K systèmes parmi les T systèmes de références. Le vecteur de coordonnées v du nouveau locuteur est supposé être une combinaison linéaire des K vecteurs de coordonnées $v_1, v_2, \ldots, v_K$ associés aux K systèmes de référence retenus, tel que :

$$v = \sum_{i=1}^{K} w_i \, v_i$$

Localiser le nouveau locuteur dans l'espace des locuteurs consiste à déterminer les K poids $w_1, w_2, \ldots, w_K$.

2.6.4.3 *EigenVoices*

Dans les techniques précédentes, la sélection des K systèmes de référence entrant en jeu dans le processus de localisation du nouveau locuteur est cruciale et influence grandement les performances de l'adaptation. L'approche adoptée par les *EigenVoices* [66; 89; 67] consiste à réduire l'espace initial des locuteurs pour obtenir un espace plus petit en dimension conservant toutefois la plus grande variabilité des données. Ces techniques se distinguent donc des méthodes décrites dans la section précédente uniquement par le fait que la localisation du nouveau locuteur est réalisée à partir de vecteurs de coordonnées non plus extraits des systèmes de référence mais ayant subis une transformation afin de les rendre plus discriminants. L'objectif de cette opération de réduction de dimension de

l'espace des locuteurs, à l'aide d'une technique comme l'analyse en composantes principales (ACP) par exemple, est de réduire le nombre K de poids à estimer pour localiser un nouveau locuteur. Ces techniques permettent ainsi d'utiliser moins de données d'adaptation pour estimer les poids du vecteur de coordonnées du nouveau locuteur.
Les premiers travaux sur les *EigenVoices*, réalisés par Kuhn, Nguyen et Junqua [66; 89; 67], ont porté sur leur emploi pour une adaptation rapide destinée à une reconnaissance de lettres isolées. Par exemple, en utilisant une seule lettre comme données d'adaptation, les *EigenVoices* ont permis d'obtenir une réduction relative du taux d'erreur de 16% par rapport à celui du système indépendant du locuteur [65]. Depuis la mise en évidence de leur efficacité dans le cas d'une adaptation rapide pour des systèmes à grand vocabulaire [8], les techniques basées sur les *EigenVoices* font actuellement l'objet d'un intérêt accrue de la part des chercheurs.
Par exemple, Tsao *et al.* [104] proposent d'améliorer la qualité de l'espace propre des locuteurs en le segmentant en N sous-espaces propres. La localisation du nouveau locuteur est ainsi supposée être plus précise. Plusieurs méthodes de segmentation de l'espace sont proposées. La première consiste à classer les gaussiennes des modèles acoustiques des systèmes dépendant du locuteur en N clusters. Un sous-espace est alors construit en utilisant les paramètres des gaussiennes regroupées dans l'un des N clusters. La deuxième approche consiste à segmenter les composants des vecteurs de moyennes des gaussiennes en N groupes de coefficients (énergie, MFCC, Δ MFCC, etc.). L'espace propre est alors segmenté en N sous-espace en utilisant pour chaque sous-espace les coefficients du groupe correspondant. Les résultats expérimentaux montrent que toutes ces méthodes de segmentation donnent de meilleurs résultats que l'approche classique des *EigenVoices*. Par ailleurs, leur combinaison permet d'obtenir encore de meilleurs résutats.
Une description plus détaillée de la technique classique des *EigenVoices*, agrémentée de résultats expérimentaux en utilisant le moteur *ESPERE*, est présentée dans le chapitre 6, page 95.

2.6.5 Techniques hybrides

Afin d'allier l'efficacité de *MLLR* lorsque la quantité de données d'adaptation est relativement faible et la capacité de convergence de *MAP* vers les modèles du *SDL* lorsque la quantité de données d'adaptation augmente, plusieurs travaux, dont [26; 89], ont porté sur la combinaison de *MLLR* et de *MAP*. La combinaison consiste à utiliser l'une des techniques après l'autre. Les modèles acoustiques du *SIL* sont adaptés à l'aide de l'une des techniques, puis l'autre technique est employée pour réadapter les modèles précédemment adaptés. Dans ces articles, *MAP* est appliquée aux modèles adaptés par *MLLR*. En dépit de sa simplicité, cette approche permet de réduire le taux d'erreur en mots de l'ordre de 20% par rapport aux performances obtenues par *MLLR* ou *MAP*, pour une tâche de reconnaissance de lettres isolées.
Récemment dans [99], Siohan *et al.* ont développé une méthode qui permet d'obtenir des performances significativement meilleures que *MLLR* et que *MAP* quelle que soit la quantité de données d'adaptation disponible. Pour cela, leur technique estime conjointement la transformation linéaire de *MLLR* et les vecteurs de moyenne des *HMMs* de *MAP*, en employant le critère du maximum *a posteriori*. Les vecteurs de moyenne Θ des *HMMs* et

les paramètres de la transformation W sont considérés comme des vecteurs aléatoires. Ils sont alors estimés selon la formule :

$$(\hat{\Theta}, \hat{W}) = argmax_{\Theta,W}\ p(O|\Theta, W)\ p(\Theta, W)$$

où $p(O|\Theta, W)$ représente la vraisemblance des données d'adaptation O étant donné les paramètres Θ et W et $p(\Theta, W)$ représente la distribution *a priori* jointe de Θ et de W.

2.7 Conclusions

Nous avons présenté dans ce chapitre les principes sous-jacents à l'adaptation dans un système de reconnaissance automatique de la parole, et plus particulièrement ceux employés pour l'adaptation au locuteur des modèles acoustiques. Nous avons proposé une représentation mathématique générale permettant de caractériser une technique d'adaptation des modèles acoustiques, ainsi qu'une taxinomie de ces techniques. Nous avons également présenté un état de l'art pour chacune des familles de techniques.

L'objectif de toute technique d'adaptation est d'améliorer les performances d'un système de reconnaissance automatique de la parole. Idéalement, peu de phrases d'adaptation devraient être utilisées afin d'apprendre les paramètres des modèles acoustiques.
Afin de mieux prendre conscience de la difficulté de concevoir une telle technique d'adaptation, nous donnons dans la deuxième partie de ce manuscrit une illustration des principes théoriques mis en lumière dans ce chapitre. A cet égard, nous proposons une présentation détaillée et une étude expérimentale des techniques d'adaptation au locuteur des modèles acoustiques les plus communément utilisées actuellement, à savoir *Structural Maximum Likelihood Linear Regression*, *Structural Maximum A Posteriori* et *EigenVoices*.

Deuxième partie

Présentation et résultats expérimentaux des méthodes classiques d'adaptation au locuteur des modèles acoustiques

Chapitre 3

Conditions expérimentales

Toutes les méthodes d'adaptation présentées dans les chapitres suivants ont été évaluées expérimentalement en utilisant le moteur de reconnaissance *ESPERE* et le corpus de parole *Resource Management* (*RM)*. Ce chapitre est destiné à décrire précisément les conditions employées respectivement pour la phase d'apprentissage et la phase d'adaptation des systèmes de reconnaissance automatique de la parole, ainsi que pour la phase de validation des résultats d'adaptation.

La première section fournit une description des caractéristiques du corpus de parole que nous avons utilisé pour l'apprentissage des systèmes de *RAP*, pour l'adaptation du système indépendant du locuteur ainsi que pour les tests d'adaptation. Nous décrivons dans la deuxième section les caractéristiques du moteur de reconnaissance *ESPERE*. Dans la troisième section sont enfin exposés le paramétrage des systèmes de reconnaissance utilisés ainsi que la manière dont le corpus *RM* a été employé.

3.1 Corpus de parole *Resource Management*

Le corpus *Resource Management* contient de la parole numérisée et transcrite destinée à la conception et à l'évaluation de systèmes de reconnaissance en parole continue.
Toutes les données de *RM* proviennent de phrases lues lors d'une tâche de gestion des ressources d'une compagnie navale, dans un environnement relativement calme. Le corpus total contient plus de 25000 phrases prononcées en anglais par plus de 160 locuteurs natifs américains, et numérisées à 16KHz, avec un codage de 16 bits. La durée de chaque phrase varie de trois à cinq secondes.

RM est composé de deux ensembles : l'ensemble *RM1* et l'ensemble d'extension *RM2*. *RM1* est lui-même divisé en deux ensembles de données : l'ensemble *RM1 SD* [15] contenant des données pour l'apprentissage, le développement et l'évaluation de systèmes dépendant du locuteur (*RM1 SD App.*, *RM1 SD Dév.*, *RM1 SD Eval.*) et l'ensemble de données *RM1 SI* [16] pour l'apprentissage, le développement et l'évaluation d'un système indépendant

15. *SD* pour *Speaker Dependent*.
16. *SI* pour *Speaker Independent*.

du locuteur (*RM1 SI App.*, *RM1 SI Dév.*, *RM1 SI Eval.*). L'extension *RM2* contient des données supplémentaires pour l'apprentissage de systèmes dépendant du locuteur, ainsi que pour l'évaluation de ces systèmes.

Les principales caractéristiques des ensembles de données de *RM* sont résumées dans le tableau 3.1 [17].

		Phrases par locuteur			
Ensemble	Nombre de locuteurs	Phrases standards	Phrases de dialecte	Phrases d'adaptation rapide	Phrases de tests
RM1 SI App.	80	40	2	-	-
RM1 SI Dév.	40	30	2	10	-
RM1 SI Eval.	40	30	2	10	-
RM1 SD App.	12	600	2	10	-
RM1 SD Dév.	12	-	-	-	100
RM1 SD Eval.	12	-	-	-	100
RM2	4	600	2	10	120

TABLE 3.1 – Caractéristiques générales du corpus *Resource Management*

L'ensemble *RM1 SD App.* contient les données de 12 locuteurs, chacun d'eux ayant lu un ensemble de 600 phrases d'adaptation, 2 phrases "de dialecte [18]" et 10 phrases destinées à une adaptation rapide. Cet ensemble est ainsi constitué de 7344 phrases. Les ensembles *RM1 SD Dév.* et *RM1 SD Eval.* sont constitués d'un total de 2400 phrases : les 12 locuteurs de l'ensemble *RM1 SD* ont chacun prononcé 100 phrases pour des tests d'évaluation et 100 autres phrases pour des tests de développement.

L'ensemble *RM1 SI* contient les données issues des phrases prononcées par 160 locuteurs. Les données de ces locuteurs sont disponibles dans trois ensembles : l'ensemble d'apprentissage *RM1 SI App.*, l'ensemble de développement *RM1 SI Dév.* et l'ensemble d'évaluation *RM1 SI Eval.*. L'ensemble d'apprentissage contient 80 locuteurs, l'ensemble de développement et l'ensemble d'évaluation contiennent 40 locuteurs chacun. Parce que certains locuteurs de *RM1 SD* sont également présents dans l'ensemble *RM1 SI*, les locuteurs communs à ces deux ensembles sont traditionnellement exclus de *RM1 SI* afin que l'apprentissage du système indépendant du locuteur n'influence pas l'adaptation pour certains locuteurs. En excluant ces locuteurs, l'ensemble d'apprentissage et l'ensemble de développement ne contiennent respectivement plus que 72 locuteurs et 37 locuteurs disponibles pour un apprentissage cohérent du système indépendant du locuteur.
Dans l'ensemble d'apprentissage, chaque locuteur prononça 2 phrases de dialecte et

17. Les nombres de phrases indiqués dans ce tableau correspondent aux phrases disponibles *par locuteur*.

18. Les phrases de dialecte peuvent être utilisées pour une adaptation dans le cas où le nouveau locuteur n'a pas le même dialecte que les locuteurs utilisés pour l'apprentissage.

40 phrases d'apprentissage, pour un total de 3360 phrases enregistrées. Dans l'ensemble de développement et dans l'ensemble d'évaluation, chaque locuteur prononça 2 phrases de dialecte, 10 phrases d'adaptation rapide et 30 phrases d'apprentissage.

Dans ces trois ensembles, chacun des 1600 énoncés fût successivement prononcé par deux locuteurs différents. En outre, aucune phrase ne fût prononcée deux fois par le même locuteur.

L'ensemble *RM2* est composé d'un total de 2928 phrases. Deux hommes et deux femmes ont chacun prononcé 600 phrases destinées à l'adaptation et à l'apprentissage de systèmes dépendant du locuteur, 2 phrases de dialecte, 10 phrases pour l'adaptation rapide et 120 phrases pour les tests de reconnaissance.

3.2 Moteur de reconnaissance *ESPERE*

ESPERE (Engine for SPEech REcognition) est un ensemble de modules logiciels développés au LORIA pour la reconnaissance automatique de la parole à base de *HMMs* du premier ordre. Il est composé de trois modules : le module de paramétrisation du signal vocal, le module d'apprentissage et le module de reconnaissance.

3.2.1 Module de paramétrisation du signal vocal

Ce module permet de transformer le signal vocal en une suite de vecteurs d'observation dont les coefficients sont des *MFCC* (*Mel Frequency Cepstral Coefficients*).

L'utilisateur peut agir sur la paramétrisation du signal en spécifiant la taille de la fenêtre d'analyse du signal, son décalage, le nombre de filtres triangulaires, ainsi que les valeurs de la fréquence minimale et de la fréquence maximale de ces filtres. Il peut définir le nombre des coefficients *MFCC* composant un vecteur d'observation, ainsi que leur type. Pour cela, il doit spécifier le nombre de coefficients statiques, auquel il peut y ajouter les coefficients des dérivées premières (*delta*) et secondes (*delta-delta*) des cepstres. Il peut en outre supprimer le coefficient $C0$ (l'énergie) ou le remplacer par le logarithme de l'énergie du signal.

3.2.2 Module d'apprentissage

L'utilisateur peut définir la topologie des *HMMs*, c'est-à-dire le nombre d'états, les transitions permises entre états et le nombre de gaussiennes des mélanges de gaussiennes modélisant les densités de probabilité d'observation associées aux états. L'unité acoustique qui est modélisée par un *HMM* peut être un mot, un phone ou un triphone.

L'apprentissage de ces unités peut être réalisée soit de manière isolée, soit de manière continue [19]. Dans le premier cas, chaque signal acoustique du corpus d'apprentissage nécessite un étiquetage manuel en terme d'unités acoustiques. L'identité, le début et la fin de

19. *embedded*

ces unités doivent être précisés pour chaque signal acoustique afin de déterminer à quels vecteurs d'observations elles sont associées pendant l'apprentissage. Dans le deuxième cas, la transcription de la phrase en terme d'unités acoustiques doit être indiquée.

Le nombre N_D de gaussiennes associées aux mélanges de gaussiennes des états est déterminé par l'utilisateur avant le processus d'apprentissage. L'apprentissage des modèles acoustiques est réalisé de manière itérative. A chaque itération, les modèles acoustiques sont appris en utilisant le même corpus d'apprentissage. L'apprentissage est initié avec des *HMMs* ne comportant qu'une seule gaussienne par état. Le nombre de gaussiennes de tous les états des modèles acoustiques est ensuite doublé à chaque itération, selon un mécanisme de division, jusqu'à obtenir la valeur N_D spécifiée.

3.2.3 Module de reconnaissance

Le module de reconnaissance d'*ESPERE* nécessite des *HMMs* acoustiques, appris selon la méthode décrite dans la section précédente, un lexique de prononciation et une grammaire. La structure du lexique permet à l'utilisateur de spécifier plusieurs prononciations pour un mot. La grammaire peut être représentée par des modèles bigrammes ou des modèles *word-pair*.

L'algorithme de reconnaissance est un algorithme en une passe (voir paragraphe 1.4, page 21) basé sur l'algorithme de Viterbi [33]. Il permet de déterminer la meilleure séquence d'états ayant permis de générer une séquence d'observations O. Afin d'améliorer la vitesse du processus de reconnaissance, un seuil peut être utilisé pour élaguer les chemins les moins probables, c'est-à-dire les suites de mots qui sont les moins susceptibles de correspondre à l'énoncé de la phrase qui a été prononcée.

3.3 Apprentissage, adaptation et évaluation des *SRAP*

3.3.1 Phase d'apprentissage

ESPERE a permis de construire le système indépendant du locuteur (*SIL*) et les systèmes dépendant du locuteur (*SDL*).
Pour chacun des 16 locuteurs d'adaptation (12 appartiennent à l'ensemble *RM1 SD App.* et 4 à l'ensemble *RM2*), un système dépendant du locuteur a été appris afin de pouvoir comparer les performances du système adapté à ce locuteur avec celles du système dépendant du locuteur.
Les techniques d'adaptation telles que *EigenVoices* utilisent un ensemble de systèmes dépendant du locuteur. Chaque système est construit à partir des phrases prononcées par un des locuteurs présents dans le corpus d'apprentissage. Un système dépendant du locuteur fut donc également généré pour chacun des 72 locuteurs d'apprentissage appartenant à *RM1* SI.

Les modèles acoustiques du système indépendant du locuteur furent appris en utilisant les 2880 phrases standards prononcées par 72 locuteurs de l'ensemble

RM1 SI. L'algorithme de Baum-Welch fut utilisé pour l'apprentissage de ces systèmes.

Les modèles acoustiques de chacun des 16 systèmes dépendant d'un locuteur d'adaptation furent appris à partir des modèles acoustiques du *SIL* en employant 10 itérations de l'algorithme de Baum-Welch et en utilisant les 600 phrases standards prononcées par l'un des 16 locuteurs d'adaptation.

Les modèles acoustiques de chacun des 72 systèmes dépendant d'un locuteur d'apprentissage furent appris par adaptation des modèles acoustiques du système indépendant du locuteur en utilisant les 40 phrases standards de chacun des locuteurs de *RM1* SI. Chaque système dépendant du locuteur a été appris à partir du système indépendant du locuteur en utilisant 10 itérations de l'algorithme d'adaptation *SMAP* (chapitre 5, page 79). Le nombre d'itérations et la méthode utilisée pour générer ces 72 systèmes dépendant du locuteur furent déterminés empiriquement et se révélèrent être les meilleurs parmi les tests réalisés.

Les résultats d'études antérieures réalisées sur le moteur *ESPERE* au sein de l'équipe PAROLE [32] ont montré expérimentalement qu'il était plus efficace d'utiliser des vecteurs d'observation constitués de 35 coefficients *MFCC* extraits du signal de parole. Tous nos systèmes utilisent donc des vecteurs d'observations de 35 coefficients cepstraux, à savoir les 11 cepstres $C1$ à $C11$, les 12 dérivées premières et les 12 dérivées secondes des cepstres $C0$ à $C11$.

Les modèles acoustiques de chaque système sont représentés par un ensemble de 46 *HMMs*. Chaque phonème anglais est représenté par un *HMM* gauche-droit sans saut à 3 états, tandis que le silence est caractérisé par un *HMM* à un état. La densité de probabilité d'observation de chaque état d'un *HMM* a été modélisée par un mélange de 32 gaussiennes. Ce choix fut déterminé empiriquement, en procédant à l'apprentissage de plusieurs systèmes indépendant du locuteur contenant un nombre variable de gaussiennes par état (8,16 ou 32) et en évaluant chacun de ces systèmes indépendant du locuteur pour chacun des 16 locuteurs d'adaptation. L'apprentissage des modèles acoustiques de chacun de ces systèmes indépendant du locuteur a été réalisé en utilisant 20 itérations de l'algorithme de Baum-Welch. Les résultats de ces expériences sont donnés dans le tableau 3.2 et représentent le taux de reconnaissance en mots moyenné sur les 16 locuteurs d'adaptation.

Nombre de gaussiennes par état	Taux de reconnaissance en mots
8	84.88%
16	86.58%
32	**87.30%**

TABLE 3.2 – Performances de systèmes indépendant du locuteur contenant 8, 16 ou 32 gaussiennes par état

L'ensemble des modèles acoustiques d'un *SRAP* comptabilise donc un total de 4352 gaussiennes ($45 \times 3 \times 32 + 32 = 4352$), ce qui représente 152320 moyennes de gaussiennes potentiellement sujets à une adaptation.

La moyenne du taux de reconnaissance en mot des 16 systèmes dépendant du locuteur (ceux appris en utilisant les phrases prononcées par les 16 locuteurs d'adaptation) est de 95.58%, en sachant que la taille du vocabulaire est de 991 mots.

3.3.2 Phase d'adaptation

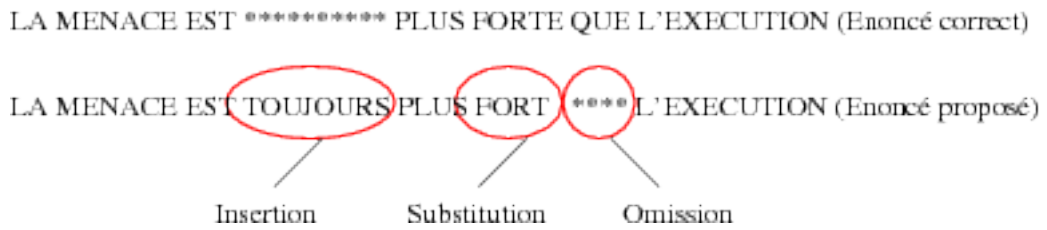
Toutes les techniques d'adaptation étudiées dans ce mémoire furent appliquées au système indépendant du locuteur pour générer un système adapté. Le système indépendant du locuteur fut adapté pour chacun des 12 locuteurs appartenant à *RM1* SD et pour chacun des 4 locuteurs de *RM2*, en utilisant une partie ou la totalité des 600 phrases standards prononcées par chacun de ces locuteurs.

3.3.3 Phase d'évaluation

Les tests de reconnaissance ont été réalisés en utilisant la grammaire *word-pair* standard fournie avec le corpus *RM*. Cette grammaire représente la liste des suites de deux mots qui sont valides pour la reconnaissance des phrases de *RM*.

Le taux de reconnaissance moyen en mot d'un *SRAP* est obtenu après l'évaluation de ce système sur l'ensemble des 1680 phrases de tests prononcées par les 12 locuteurs de *RM1 SD Dév.* et les 4 locuteurs de *RM2* ($12 \times 100 + 4 \times 120 = 1680$). Le taux de reconnaissance en mot suffit pour évaluer l'efficacité d'une technique d'adaptation sur les performances d'un système de *RAP*. Le taux de reconnaissance en phrase dépend en effet largement de la qualité du modèle de langage.

Pour chaque phrase du corpus de test, l'énoncé proposé par le système de *RAP* est aligné avec son énoncé correct. L'alignement ainsi obtenu permet de calculer le nombre de mots correctement reconnus et le nombre d'erreurs commises, en terme d'omission, de substitution ou d'insertion de mots. Par exemple, si le système propose l'énoncé "La menace est toujours plus fort l'exécution" alors que l'énoncé correct de la phrase prononcée est "La menace est plus forte que l'exécution", l'algorithme d'alignement dynamique pourra faire correspondre ces deux énoncés de la manière suivante :



La chaîne $***$ représente soit l'insertion d'un mot, dans le cas où il est présent dans l'énoncé correcte, soit l'omission d'un mot, dans le cas où il est présent dans l'énoncé proposé. Ici, l'alignement permet de mettre en évidence une substitution, une insertion, une omission et cinq mots correctement reconnus.

3.3.4 Calcul du taux de reconnaissance en mot

Le taux de reconnaissance en mot $WordAcc$ est calculé en comptabilisant le nombre d'omissions, d'insertions, de substitutions et le nombre de mots correctement reconnus sur l'ensemble des phrases prononcées dans le corpus de test. Il est obtenu selon la formule suivante :

$$WordAcc = \frac{NbrOk - NbrIns}{NbrOk + NbrOmi + NbrSub} \times 100 \tag{3.1}$$

où $NbrOk$, $NbrIns$, $NbrOmi$ et $NbrSub$ représentent respectivement le nombre total de mots correctement reconnus, le nombre total d'insertions, le nombre total d'omissions et le nombre total de substitutions. Cette mesure du taux de reconnaissance en mot permet ainsi de pénaliser les systèmes de reconnaissance qui insèrent un trop grand nombre de mots.

Chapitre 4

Structural Maximum Likelihood Linear Regression (SMLLR)

Nous débutons l'étude des méthodes classiques d'adaptation au locuteur des modèles acoustiques par la technique *SMLLR*. *SMLLR* est actuellement la technique d'adaptation des modèles acoustiques qui est la plus utilisée dans ce domaine et qui fait continuellement l'objet de nouvelles améliorations.
Initialement proposé par *C.J. Leggetter* et *P. C. Woodland* [78], *SMLLR*[20] permet de réestimer les paramètres des modèles acoustiques d'un *SRAP* indépendant du locuteur. *SMLLR* appartient à la famille des techniques d'adaptation qui sont capables de déterminer dynamiquement, en fonction de la quantité de données d'adaptation disponibles, le nombre de transformations qui peuvent être estimées de manière robuste et appliquées efficacement aux paramètres des modèles acoustiques du système indépendant du locuteur.

Dans ce chapitre, nous présenterons dans un premier temps les concepts adoptés dans la version classique de *SMLLR*, en précisant notamment comment est réalisée une adaptation globale des paramètres des modèles acoustiques, comment une adaptation fine de ces paramètres peut être obtenue et comment sont estimés les paramètres d'une transformation. Nous discuterons ensuite des méthodes de récolte des statistiques suffisantes qui sont utilisées dans le cadre de *SMLLR* pour estimer efficacement les paramètres des transformations en fonction du mode d'adaptation choisie (mode par lot ou mode incrémental). La troisième section de ce chapitre sera consacrée aux choix que nous avons fait concernant l'implantation de *SMLLR*. Nous présenterons en particulier l'algorithme retenu pour construire l'arbre de gaussiennes et la technique choisie pour constituer les classes de régression. Nous donnerons dans la quatrième section les résultats des expériences d'évaluation de la version implantée de *SMLLR*, ménées à l'aide du corpus *RM* et du moteur de reconnaissance *ESPERE*. Ces expériences fûrent réalisées en mode par lot supervisé et en mode incrémental non supervisé. Nous dresserons enfin nos conclusions sur l'efficacité d'adaptation de *SMLLR* et sur les choix de paramétrage qui nous ont permis

20. Le terme *MLLR* fut utilisé dans l'article mentionné alors qu'il s'agissait d'une nouvelle technique qui consistait à employer une structure hiérarchique dans le cadre de *MLLR*. Nous avons choisi d'employer ici le terme *SMLLR* pour signifier qu'il s'agit de la technique *MLLR* structurelle, c'est-à-dire utilisant une structure hiérarchique, dans le but de la distinguer de la technique *MLLR*.

de l'utiliser aussi efficacement que possible dans le cas d'une adaptation par lot supervisée et dans le cas d'une adaptation incrémentale non supervisée.

4.1 Approche classique

Dans le cas le plus général, *SMLLR* estime une transformation, représentée sous la forme d'une régression linéaire [21], qui modélise les différences supposées linéaires entre les conditions acoustiques d'apprentissage et les conditions acoustiques d'utilisation du *SRAP*. Les paramètres de cette régression linéaire sont estimés en maximisant la vraisemblance sur l'ensemble des données d'adaptation. Le système adapté est traditionnellement obtenu en appliquant cette transformation linéaire à chacune des moyennes des gaussiennes des modèles acoustiques.

4.1.1 Adaptation globale

La figure 4.1 illustre le cas d'une adaptation globale en utilisant *SMLLR* : toutes les gaussiennes des modèles acoustiques sont modifiées par l'application d'une seule régression linéaire.

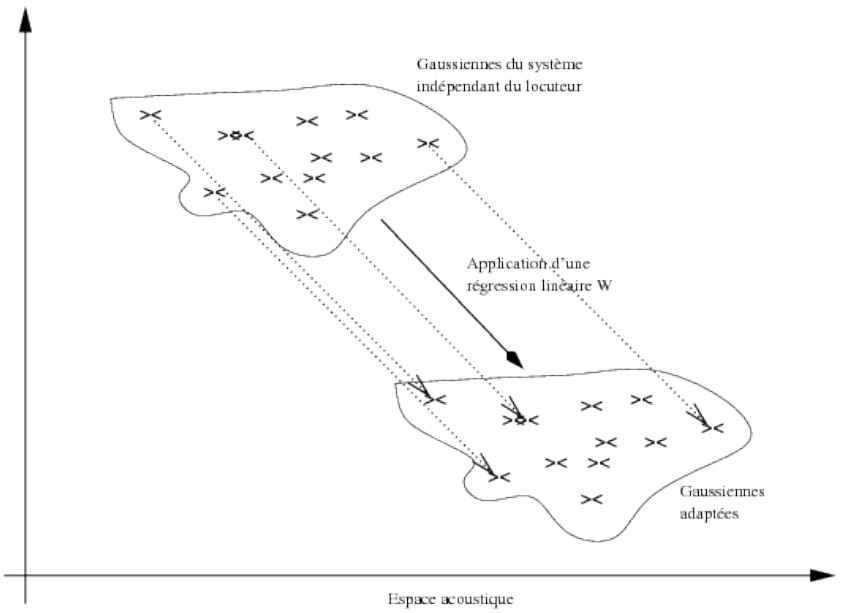


FIGURE 4.1 – Adaptation globale avec *SMLLR*

Lorsqu'une seule transformation est utilisée, l'ensemble des gaussiennes de tous les modèles acoustiques sont déplacées dans l'espace acoustique dans la même direction. Certaines gaussiennes peuvent donc se trouver à des emplacements de l'espace qui ne correspondent

21. Nous donnons dans le paragraphe 4.1.3, page 63, la définition d'une régression linéaire.

pas forcément à des positions optimales qui permettraient aux modèles acoustiques de mieux prendre en compte les caractéristiques acoustiques des données d'adaptation.

4.1.2 Classes de régression

L'utilisation de plusieurs transformations permet d'obtenir une adaptation plus fine des gaussiennes des modèles acoustiques. Dans ce cas, une transformation est appliquée à un ensemble de gaussiennes semblables et regroupées au sein d'une même classe selon un critère donné. L'association d'une transformation à un ensemble de gaussiennes constitue une *classe de régression*.
L'utilisation de classes de régression permet d'obtenir une adaptation plus précise des paramètres des modèles acoustiques. En effet, les gaussiennes réunies dans une même classe, et donc semblables au sens d'un certain critère, sont déplacées dans l'espace acoustique selon la même direction. Par extension, deux gaussiennes n'appartenant pas à la même classe de régression pourront être déplacées dans l'espace acoustique selon des directions différentes. De cette manière, chaque gaussienne est davantage susceptible d'atteindre sa propre position optimale, fournissant ainsi aux modèles acoustiques l'opportunité de délivrer de meilleures performances.

La figure 4.2 présente le processus d'adaptation de *SMLLR* dans le cas où le système indépendant du locuteur contient au total douze gaussiennes. L'adaptation utilise deux classes de régression C_1 et C_2. C_1 contient huit gaussiennes, tandis que C_2 en regroupe quatre.

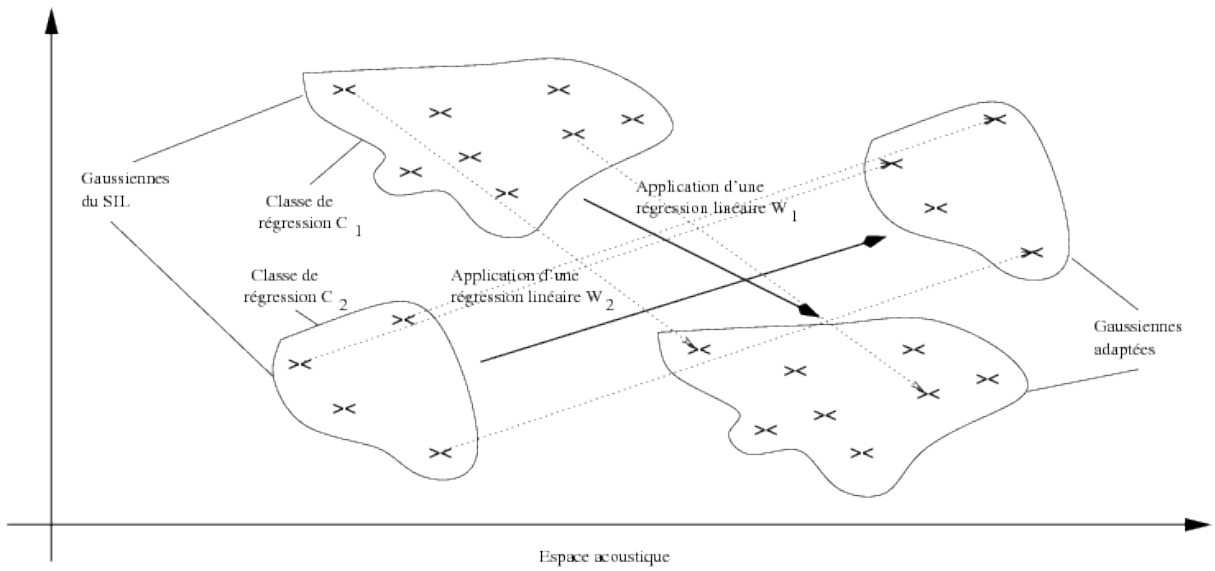


FIGURE 4.2 – Adaptation avec *SMLLR* en utilisant deux classes de régression

Les deux transformations W_1 et W_2 sont estimées en utilisant les données d'adaptation associées aux gaussiennes des classes C_1 et C_2, respectivement. W_1 est appliquée aux

gaussiennes de C_1 et W_2 à celles de C_2.
Nous pouvons contater dans cet exemple que les gaussiennes ont pu être déplacées dans des directions totalement différentes selon leur appartenance à l'une ou l'autre des classes de régression.

La définition des classes de régression peut être réalisée selon deux approches. La première consiste à définir un ensemble de classes de régression avant le processus d'adaptation, en ayant une connaissance *a priori* de la quantité disponible de données d'adaptation. Nous parlerons alors de *classes statiques*. La deuxième approche construit l'ensemble des classes de régression de manière dynamique, c'est-à-dire pendant le processus d'adaptation, en fonction de la quantité de données d'adaptation disponibles. Il s'agit dans ce cas de *classes dynamiques*.

4.1.2.1 Classes de régression statiques

Usuellement, des connaissances phonétiques sont utilisées pour définir les classes de régression statiques. Chaque classe contient des modèles phonétiquement proches. C'est l'approche utilisée notamment par Leggetter et Woodland [77]. Par exemple, on peut choisir de constituer trois classes phonétiques : dans la première sont regroupés les modèles acoustiques correspondant aux voyelles, la deuxième contient les modèles acoustiques correspondant aux consonnes et la troisième inclut le ou les modèles acoustiques du silence. Soulignons encore une fois que l'utilisation de classes de régression statiques ne s'applique que dans le cas où l'on connaît la quantité disponible de données d'adaptation. En effet, dans le cas contraire, l'adaptation peut conduire à deux éventualités. L'une des éventualités est qu'une ou plusieurs des transformations peuvent avoir été mal estimées. Ceci se produit lorsqu'une trop faible quantité de données d'adaptation est associée aux gaussiennes de certaines classes. Il en résulte alors le plus souvent une dégradation des performances du système adapté. L'autre éventualité est que le nombre de transformations qui ont été effectivement estimées soit inférieur au nombre de transformations qui auraient pu l'être en regard de la quantité importante de données d'adaptation disponibles. Dans cette situation, l'adaptation aurait pu être plus précise en utilisant un nombre plus important de classes de régression.

4.1.2.2 Classes de régression dynamiques

Pour pallier ce problème d'estimation robuste des transformations, *SMLLR* détermine le nombre de classes de régression de manière dynamique, en fonction de la quantité de données d'adaptation disponibles. Plus la quantité disponible de données d'adaptation sera grande, plus nombreuses sont les transformations qui pourront être estimées de manière robuste.
L'aspect structurel de *SMLLR* s'explique par le fait que cette technique utilise une structure arborescente pour déterminer l'ensemble des classes de régression, et donc le nombre de transformations à estimer. L'idée consiste à classer tout d'abord l'ensemble des gaussiennes des modèles acoustiques dans un arbre. Chaque nœud de l'arbre contient un ensemble de gaussiennes qui sont supposées être proches dans l'espace acoustique, au sens

d'une certaine mesure de distance. *SMLLR* émet alors l'hypothèse que les gaussiennes qui possèdent des valeurs voisines, et donc qui sont regroupées dans la même classe, seront modifiées à l'aide de la même transformation.
Plusieurs algorithmes ont été utilisés pour construire l'arbre de gaussiennes. Les techniques les plus simples ([76], [77]) utilisent une mesure de distance qui modélise des connaissances phonétiques ou acoustiques pour déterminer les gaussiennes qui doivent être regroupées au sein de la même classe. Nous donnons dans la section 4.3.1, page 72, ainsi que dans l'annexe B, la description d'un tel algorithme, étant donné qu'il s'agit de celui que nous avons choisi d'utiliser.
M.J.F. Gales a expérimenté dans [35] d'autres techniques qui permettent de générer de manière optimale l'arbre de gaussiennes, dans le sens où il maximise la vraisemblance des données d'adaptation. A l'issue de la phase d'adaptation, les arbres générés à l'aide de ces techniques ont permis de constater une amélioration plus significative de la vraisemblance des données d'adaptation, par rapport à celle observée par les techniques classiques. Toutefois, ils n'ont pas permis aux systèmes adaptés de délivrer de meilleures performances en terme de taux de reconnaissance.
Une fois l'arbre de gaussiennes construit, *SMLLR* l'utilise pour définir l'ensemble des classes de régression. Dans [76], la sélection des classes de régression est réalisée en employant une technique basée sur le concept de quantité de données suffisante pour une estimation robuste d'une transformation. Ce concept suppose qu'un nombre α_{SMLLR} minimum de vecteurs d'observations (ou trames), issues des données d'adaptation, doit être associées aux gaussiennes d'une classe pour que la transformation correspondante soit estimée de manière robuste. Lorsque la phase d'alignement des données d'adaptation est réalisée, à l'aide de la procédure *forward-backward* de l'algorithme *EM*, le nombre total de trames associées à chaque gaussienne est connu[22]. Un parcours de l'arbre est alors initié à partir des feuilles pour déterminer l'ensemble des classes de régression. Seuls les nœuds disposant d'un nombre suffisant de trames (supérieur à α_{SMLLR}) sont ainsi retenus comme classe de régression. Cette méthode de définition de l'ensemble des classes de régression permet ainsi de s'assurer que chaque transformation sera estimée de manière robuste et que chaque gaussienne sera adaptée avec la transformation la plus spécifique.
Dans [35], deux autres techniques ont été proposées. Plus coûteuses en terme de temps de calcul, mais automatiques dans le sens où le nombre de trames minimum α_{SMLLR} n'a plus à être spécifié, ces techniques se sont révélées aussi performantes que la technique classique.

4.1.3 Estimation d'une régression linéaire

Nous abordons maintenant le problème de l'estimation des paramètres d'une régression linéaire dans le cas où elle est appliquée aux moyennes des gaussiennes des *HMMs*.

22. Il s'agit de la quantité $\Gamma_r = \sum_{t=1}^{T} \gamma_r(t)$ qui représente la probabilité d'occupation de la gaussienne g_r. Elle est calculée à partir de la probabilité *a posteriori* $\gamma_r(t)$ de se trouver dans la gaussienne g_r au temps t en sachant que la séquence d'observation O a été générée.

4.1.3.1 Qu'est-ce qu'une régression linéaire ?

Le terme de régression exprime l'hypothèse qu'il existe une relation liant une variable inconnue dite dépendante avec une ou plusieurs variables connues dites indépendantes. L'utilisation d'une régression linéaire, dans notre cas, nous conduit ainsi à émettre l'hypothèse que chaque coefficient $\hat{\mu}_i$ du vecteur de moyenne $\hat{\mu} = (\hat{\mu}_1\, \hat{\mu}_2\, \ldots\, \hat{\mu}_{N_D})$ d'une gaussienne du système adapté s'exprime par une combinaison linéaire de tous les coefficients du vecteur de moyenne μ de cette même gaussienne appartenant au système indépendant du locuteur. Cette relation s'exprime par la formule :

$$\hat{\mu}_i = w_{i,1} \times \mu_1 + w_{i,2} \times \mu_2 + \ldots + w_{i,N_D} \times \mu_{N_D} + w_{i,N_D+1} \quad \forall i = 1, 2, \ldots, N_D \tag{4.1}$$

où N_D est la dimension d'un vecteur de moyenne μ et $w_{i,1}, w_{i,2}, \ldots, w_{i,N_D+1}$ sont les poids de la combinaison linéaire utilisée pour estimer le coefficient $\hat{\mu}_i$.
Par convention, nous noterons $w_i = (w_{i,1}\, w_{i,2}\, \ldots\, w_{i,N_D+1})$ le vecteur de poids affecté à l'estimation du coefficient $\hat{\mu}_i$. La concaténation de chaque vecteur de poids w_i, pour $i = 1, 2, \ldots, N_D$, forme ainsi une matrice W de dimension $N_D \times (N_D + 1)$, appelée matrice de régression linéaire et telle que :

$$W = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N_D} \end{pmatrix} = \begin{pmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,N_D+1} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,N_D+1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N_D,1} & w_{N_D,2} & \cdots & w_{N_D,N_D+1} \end{pmatrix}$$

L'estimation des coefficients du vecteur de moyenne $\hat{\mu}$ d'une gaussienne du système adapté est alors donnée par la formule suivante :

$$\hat{\mu} = W\, \xi \tag{4.2}$$

où $\xi = (\mu'\ 1)'$ est le vecteur étendu du vecteur de moyenne μ d'une gaussienne du système indépendant du locuteur.

Par la suite, nous supposerons dans un premier temps qu'une seule matrice de régression W est utilisée pour modifier l'ensemble des moyennes des gaussiennes des modèles acoutiques du *SRAP*. Nous distinguerons alors les cas où cette transformation globale est pleine ou diagonale, selon que l'on considère respectivement qu'aucun des coefficients cepstraux du vecteur de moyenne n'est indépendant, ou que tous les coefficients cepstraux sont indépendants entre eux. Nous indiquerons ensuite comment estimer les paramètres d'une transformation linéaire lorsqu'elle est associée à un ensemble de gaussiennes au sein d'une classe de régression.

4.1.3.2 Estimation d'une transformation globale

Rappelons que chaque gaussienne $g_r = (\mu_r, \sigma_r)$ des modèles acoustiques est caractérisée par une fonction de densité de probabilité d'observation $b_r(o)$, pour $r = 1, 2, \ldots, N_G$, telle que :

$$\begin{aligned} b_r(o) &= \mathcal{N}(o, \mu_r, \sigma_r) \\ &= \frac{1}{\sqrt{(2\pi)^{N_D}\ |\sigma_r|}}\ exp\ \left[-\frac{1}{2}\ (o-\mu_r)'\ {\sigma_r}^{-1}\ (o-\mu_r)\right] \end{aligned} \tag{4.3}$$

où

- μ_r est le vecteur de moyenne de dimension N_D de la gaussienne g_r,
- σ_r est la matrice de variances-covariances de dimension $N_D \times N_D$ de la gaussienne g_r,
- o est un vecteur d'observation de dimension N_D.

Dans *SMLLR*, la moyenne $\hat{\mu}_r$ de chaque gaussienne g_r du système adapté est obtenue par transformation linéaire de la moyenne μ_r de chaque gaussienne des modèles initiaux selon l'équation :

$$\mu_r = W\ \xi_r \text{ pour } r = 1, 2, \ldots, N_G$$

où W est la matrice de transformation linéaire, de dimension $N_D \times (N_D+1)$. Cette matrice peut être pleine :

$$W = \begin{pmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1_{N_D}+1} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,N_D+1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N_D,1} & w_{N_D,2} & \cdots & w_{N_D,N_D+1} \end{pmatrix}$$

ou diagonale :

$$W = \begin{pmatrix} w_{1,1} & 0 & \cdots & 0 & w_{1,N_D+1} \\ 0 & w_{2,2} & \ddots & \vdots & w_{2,N_D+1} \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & w_{N_D,N_D} & w_{N_D,N_D+1} \end{pmatrix}.$$

$\xi_r = (\mu_r^{'}\ \ f_b)'$ est un vecteur de dimension $(N_D + 1)$ qui représente le vecteur étendu de la moyenne de la gaussienne g_r. f_b représente le terme indiquant la prise en compte ($f_b = 1$) ou non ($f_b = 0$) du vecteur de biais $\begin{pmatrix} w_{1,N_D+1} & w_{2,N_D+1} & \ldots & w_{N_D,N_D+1} \end{pmatrix}'$ dans la transformation linéaire.
Soit Θ l'ensemble des paramètres des modèles acoustiques représentés par des *HMMs*. Dans le système adapté, chaque moyenne μ_r est obtenue par transformation linéaire de la moyenne initiale μ_r. La fonction de densité de probabilité d'observation $\hat{b}_r(o)$ de la gaussienne g_r adaptée devient donc :

$$\hat{b}_r(o) = \frac{1}{\sqrt{(2\pi)^{N_D}\ |\sigma_r|}}\ exp\ \left[-\frac{1}{2}\ (o - W\ \xi_r)'\ {\sigma_r}^{-1}\ (o - W\ \xi_r)\right]$$

L'ensemble des paramètres Θ des modèles acoustiques devient alors $\Theta = \Theta \cup W$. W est estimée au maximum de vraisemblance des données d'adaptation $O = (o_1, o_2, ..., o_T)$, ce qui revient à déterminer la matrice $\hat{W}$ telle que :

$$\hat{W} = \underset{W}{\text{argmax}} \; p(O/\Theta) \tag{4.4}$$

Or

$$\begin{aligned} p(O/\Theta) &= \sum_{S \in \Phi} p(O, S/\Theta) \\ &= \pi_{s_0} \prod_{t=1}^{T} a_{s_{t-1}s_t} b_{s_t}(o_t) \\ &= \pi_{s_0} \prod_{t=1}^{T} a_{s_{t-1}s_t} \left[\sum_{m=1}^{N_P} c_{s_t,m} b_{s_t,m}(o_t) \right] \end{aligned} \tag{4.5}$$

avec $p(O, S/\Theta)$ la probabilité de générer la séquence d'observations O en passant par le chemin $S = (s_1, s_2, ..., s_T)$ et Φ l'ensemble des séquences d'états S de longueur T.
Soit $\Psi = \{1, 2, ..., M\}^T$ l'ensemble des séquences de gaussiennes $K = (k_1, k_2, ..., k_T)$ de longueur T et $p(O, S, K/\Theta)$ la probabilité jointe de O, S et K. Alors, $p(O, S/\Theta)$ peut s'écrire comme la probabilité marginale de $p(O, S, K)$ sur Ψ telle que :

$$\begin{aligned} p(O, S/\Theta) &= \sum_{K \in \Psi} p(O, S, K/\Theta) \\ &= \sum_{K \in \Psi} \pi_{s_0} \prod_{t=1}^{T} a_{s_{t-1}s_t} c_{s_t,k_t} b_{s_t,k_t}(o_t) \end{aligned} \tag{4.6}$$

L'équation (4.4) se réécrit alors comme :

$$\hat{W} = \underset{W}{\text{argmax}} \sum_{S \in \Phi} \sum_{K \in \Psi} p(O, S, K/\Theta) \tag{4.7}$$

Il n'existe pas de solution directe pour résoudre cette équation. L'algorithme *Expectation-Maximisation* (*EM*) est alors employé pour trouver la matrice de transformation $\hat{W}$ qui maximise $p(O/\Theta)$. *EM* est un algorithme itératif qui, à partir d'un modèle Θ_t, va construire un modèle Θ_{t+1} tel que $p(O/\Theta_{t+1}) > p(O/\Theta_t)$. Les paramètres d'une matrice de transformation sont donc estimés avec *EM* en utilisant β_{SMLLR} itérations. Plus le nombre d'itérations est grand, plus les paramètres pourront être estimés avec davantage de précision.
Au lieu de maximiser la fonction $p(O/\Theta)$, *EM* utilise une fonction auxiliaire Q telle que :

$$Q(\Theta, \hat{\Theta}) = \frac{1}{p(O/\Theta)} \sum_{S \in \Phi} \sum_{K \in \Psi} p(O, S, K/\Theta) \; log \; p(O, S, K/\hat{\Theta}) \tag{4.8}$$

où $\hat{\Theta}$ est l'ensemble des paramètres à estimer et Θ l'ensemble des paramètres actuels. En maximisant la fonction Q par rapport à $\hat{\Theta}$, en remplaçant Θ par $\hat{\Theta}$ et en itérant ce

processus jusqu'à ce que la différence $Q(\Theta, \hat{\Theta}) - Q(\Theta, \Theta)$ converge, nous sommes assurés que le paramètre $\hat{\Theta}$ est celui qui maximise localement $p(O/\Theta)$.
Dans notre cas, seule W est estimée. Ainsi, seule la densité de probabilité $b_{s_t,k_t}(o_t)$ est affectée, d'où :

$$Q(\Theta, \hat{\Theta}) \;=\; cst + \frac{1}{p(O/\Theta)} \sum_{S\in\Phi} \sum_{K\in\Psi} \sum_{t=1}^{T} p(O, S, K/\Theta)\, log\, \hat{b}_{s_t,k_t}(o_t) \tag{4.9}$$

Notons :

$$\gamma_r(t) = p(s_t = f_s(r), k_t = f_g(r)/O, \Theta)$$

où $\gamma_r(t)$ est la probabilité *a posteriori* de se trouver dans la gaussienne r à l'instant t en sachant que la séquence d'observations O a été générée ; $f_s(r)$ et $f_g(r)$ sont les fonctions définies par les équations 2.4 et 2.5 au paragraphe 2.3.1, page 32. $f_s(r)$ et $f_g(r)$ retournent respectivement le numéro de l'état et le numéro de la gaussienne dans cet état où se situe physiquement la gaussienne g_r. Chaque $\gamma_r(t)$, pour $1 \leq r \leq N_G$ et $1 \leq t \leq T$ est obtenue à partir du corpus d'adaptation à l'aide de l'algorithme *forward-backward*.
Comme :

$$\sum_{S\in\Phi} \sum_{K\in\Psi} p(S, K/O, \Theta) = \frac{1}{p(O/\Theta)} \sum_{S\in\Phi} \sum_{K\in\Psi} p(O, S, K/\Theta)$$

Alors :

$$\begin{aligned} \sum_{S\in\Phi} \sum_{K\in\Psi} p(S, K/O, \Theta) \;&=\; \sum_{r=1}^{N_G} \sum_{t=1}^{T} p(s_t = f_s(r), k_t = f_g(r)/O, \Theta) \\ &=\; \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t) \end{aligned} \tag{4.10}$$

L'équation (4.9) peut donc se réécrire comme :

$$Q(\Theta, \hat{\Theta}) = cst + \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, log\, \hat{b}_r(o_t) \tag{4.11}$$

$$Q(\Theta, \hat{\Theta}) = cst \; + \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t) \left[-\frac{1}{2} \left[N_D\, log\, (2\pi) + log\, |\sigma_r| + h(o_t, r) \right] \right] \tag{4.12}$$

avec

$$h(o_t, r) = (o_t - \hat{W}\xi_r)' \, \sigma_r^{-1} \, (o_t - \hat{W}\xi_r)$$

Trouver W qui maximise la fonction $Q(\Theta, \hat{\Theta})$ revient à dériver Q par rapport à W puis à mettre la dérivée à zéro, soit :

$$\frac{\partial}{\partial \hat{W}} Q(\Theta/\hat{\Theta}) = \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t) \sigma_r^{-1} (o_t \; - \; \hat{W} \; \xi_r) \xi_r'$$

$$\frac{\partial}{\partial \hat{W}} Q(\Theta/\hat{\Theta}) \equiv 0$$

ce qui donne :

$$\sum_{r=1}^{N_G}\sum_{t=1}^{T} \gamma_r(t)\,\sigma_r^{-1}\,o_t\,\xi_r^{'} \;=\; \sum_{r=1}^{N_G}\sum_{t=1}^{T} \gamma_r(t)\,\sigma_r^{-1}\,\hat{W}\,\xi_r\,\xi_r^{'} \tag{4.13}$$

Nous allons maintenant distinguer l'estimation d'une matrice de transformation pleine de l'estimation d'une matrice de transformation diagonale.

4.1.3.2.1 Estimation d'une matrice de transformation pleine

Dans le cas où W est une matrice pleine, considérons que :

$$Z \;=\; \sum_{r=1}^{N_G}\sum_{t=1}^{T} \gamma_r(t)\,\sigma_r^{-1}\,o_t\,\xi_r^{'} \;\text{ est la partie gauche de l'équation (4.13)}$$

et que

$$Y \;=\; \sum_{r=1}^{N_G}\sum_{t=1}^{T} \gamma_r(t)\,\sigma_r^{-1}\,\hat{W}\,\xi_r\,\xi_r^{'} \;\text{ est la partie droite de (4.13).}$$

Y peut se réecrire sous la forme :

$$Y \;=\; \sum_{r=1}^{N_G} V^{(r)}\,\hat{W}\,U^{(r)}$$

avec

$$V^{(r)} \;=\; \sum_{t=1}^{T} \gamma_r(t)\,\sigma_r^{-1}$$

et

$$U^{(r)} \;=\; \xi_r\,\xi_r^{'}$$

Soient $y_{i,j}$, $v_{i,j}^{(r)}$, $u_{i,j}^{(r)}$ et $w_{i,j}$ les éléments des matrices respectives Y, $V^{(r)}$, $U^{(r)}$ et W. La valeur de l'élément $y_{i,j}$, situé à la i-ème ligne de la j-ème colonne de la matrice Y, est alors obtenue par :

$$y_{i,j} \;=\; \sum_{p=1}^{N_D}\sum_{q=1}^{N_D+1} w_{p,q}\left[\sum_{r=1}^{N_G} v_{i,p}^{(r)}\,d_{q,j}^{(r)}\right]$$

En supposant que toutes les matrices de variances-covariances σ_r sont diagonales, V devient diagonale, si bien que :

$$\sum_{r=1}^{N_G} v_{i,p}^{(r)}\,u_{q,j}^{(r)} \;=\; \sum_{r=1}^{N_G} v_{i,i}^{(r)}\,u_{j,q}^{(r)}$$

étant donné que U est symétrique.

On peut alors considérer que :

$$y_{i,j} \;=\; \sum_{q=1}^{N_D+1} w_{i,q}\,a_{j,q}^{(i)}$$

où $a_{j,q}^{(i)}$ sont les éléments de la matrice $A^{(i)}$ de dimension $(N_D+1)\times(N_D+1)$ définie par :

$$a_{j,q}^{(i)} = \sum_{r=1}^{N_G} v_{i,i}^{(r)}\, u_{j,q}^{(r)}$$

Soient $z_{i,j}$ les éléments de la matrice Z de dimension $N_D \times (N_D+1)$ définie précédemment. Alors :

$$z_{i,j} = y_{i,j} \;=\; \sum_{q=1}^{N_D+1} w_{i,q}\, a_{j,q}^{(i)}$$

$z_{i,j}$ et $a_{j,q}^{(i)}$ ne dépendent pas de $\hat{W}$ et peuvent donc être calculé à partir des statistiques suffisantes obtenues à l'aide de la procédure *forward-backward*. Ainsi, $\hat{W}$ peut être estimée à partir du système d'équations suivant :

$$w_i' \;=\; A^{(i)-1}\, z_i' \tag{4.14}$$

où w_i et z_i sont les i-èmes colonnes de $\hat{W}$ et de Z, respectivement.

Le calcul de la matrice pleine $\hat{W}$ s'effectue ligne par ligne en résolvant les N_D systèmes de N_D+1 équations linéaires. Ces équations peuvent être résolues en utilisant la méthode de décomposition LU.

4.1.3.2.2 Estimation d'une matrice de transformation diagonale Dans le cas où une matrice diagonale doit être estimée, la matrice de transformation $\hat{W}$ peut se réécrire en un vecteur $\hat{w}$ tel que :

$$\hat{w} = \begin{pmatrix} w_{1,1} \\ \vdots \\ w_{N_D,1} \\ \\ w_{1,2} \\ \vdots \\ w_{N_D,N_D+1} \end{pmatrix}$$

Soit une matrice $A^{(r)}$ de dimension $N_D \times 2N_D$, constituée des éléments du vecteur ξ_r telle que :

$$A^{(r)} = \begin{pmatrix} f_b & 0 & \cdots & \cdots & 0 & \mu_r^1 & 0 & \cdots & \cdots & 0 \\ 0 & f_b & 0 & \cdots & 0 & 0 & \mu_r^2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & f_b & 0 & 0 & \cdots & 0 & \mu_r^{N_D-1} & 0 \\ 0 & \cdots & \cdots & 0 & f_b & 0 & \cdots & \cdots & 0 & \mu_{,r}^{N_D} \end{pmatrix}$$

où μ_r^k est le k-ème élément du vecteur de moyenne μ_r de la gaussienne g_r. Alors, la réécriture de l'équation 4.13 conduit à la formule suivante d'estimation des paramètres de $\hat{w}$:

$$\hat{w} = \left[\sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, A^{(r)\prime}\, \sigma_r^{-1}\, A^{(r)} \right]^{-1} \left[\sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, A^{(r)\prime}\, \sigma_r^{-1}\, o_t \right] \tag{4.15}$$

Ainsi, seule une inversion de matrice est requise pour estimer les paramètres d'une matrice de transformation diagonale. Dans le cas où le terme d'indication de biais $f_b = 0$, toutes les matrices deviennent diagonales, ce qui simplifie encore davantage la résolution de cette équation.

4.1.3.3 Estimation d'une transformation liée à une classe de régression

Lorsqu'une matrice de transformation est appliquée à un sous-ensemble de gaussiennes, seules les statistiques suffisantes associées à ces gaussiennes sont utilisées pour l'estimation de la matrice. Soit $G_c = \{g_{s_1^{(c)}}, g_{s_2^{(c)}}, \ldots, g_{s_{M_c}^{(c)}}\}$ une classe de régression regroupant M_c gaussiennes. $\{s_1^{(c)}, s_2^{(c)}, \ldots, s_{M_c}^{(c)}\}$ sont les indices des M_c gaussiennes regroupées dans la classe G_c. L'estimation de la matrice de transformation $\hat{W}_c$ associée à la classe de régression G_c s'obtient alors en modifiant l'équation (4.13) comme suit :

$$\sum_{i=1}^{M_c} \sum_{t=1}^{T} \gamma_{s_i^{(c)}}(t)\, \sigma_{s_i^{(c)}}^{-1}\, o_t\, \xi_{s_i^{(c)}}^{\prime} \;=\; \sum_{i=1}^{M_c} \sum_{t=1}^{T} \gamma_{s_i^{(c)}}(t)\, \sigma_{s_i^{(c)}}^{-1}\, \hat{W}_c\, \xi_{s_i^{(c)}}\, \xi_{s_i^{(c)}}^{\prime} \tag{4.16}$$

4.2 Récolte des statistiques suffisantes

Comme toutes les méthodes basées sur l'algorithme *EM* pour estimer les moyennes des gaussiennes du système adapté, *SMLLR* nécessite une phase de récolte des statistiques suffisantes qui est antérieure à la phase d'estimation. Cette phase correspond à l'étape *Expectation* de l'algorithme *EM*. Elle consiste à cumuler pour chaque gaussienne g_r du système indépendant du locuteur les statistiques suffisantes $\Omega_r = \sum_{t=1}^{T} \gamma_t(r)\, o_t$ et $\Gamma_r = \sum_{t=1}^{T} \gamma_t(r)$ où $\gamma_t(r)$ est la probabilité *a posteriori* de se trouver dans la gaussienne g_r au temps t en sachant que la séquence d'observations O a été générée. Les probabilités $\gamma_t(r)$ associées aux gaussiennes g_r, pour $r = 1, 2, \ldots, N_G$, sont fournies par la procédure *forward-backward* à partir des données d'adaptation. Elles permettent de calculer les statistiques suffisantes Ω_r et Γ_r, qui sont ensuite utilisées pour estimer les variables d'adaptation. Dans le cas de *SMLLR*, elles sont employées pour estimer les paramètres d'une matrice de régression linéaire, selon la formule 4.13 qui peut maintenant être réécrite plus simplement comme :

$$\sum_{r=1}^{N_G} \sigma_r^{-1}\, \Omega_r\, \xi_r^{\prime} \;=\; \sum_{r=1}^{N_G} \Gamma_r\, \sigma_r^{-1}\, \hat{W}\, \xi_r\, \xi_r^{\prime} \tag{4.17}$$

4.2.1 Méthode simple de cumul des statistiques pour l'adaptation incrémentale

Les méthodes *MLLR* exposées dans [79; 78] furent initialement conçues pour réaliser une adaptation par lot. Aucune modification n'est donc requise pour qu'elle puisse opérer en mode incrémental, comme le suggère d'ailleurs les auteurs dans [78]. Il suffit pour cela de constituer des lots ne contenant qu'une seule phrase, de cumuler les statistiques suffisantes (comme indiqué précédemment) pour chaque gaussienne à partir de cette unique phrase, puis d'estimer les paramètres des matrices de régression linéaire.
Cette approche n'est cependant pas très efficace. En effet, l'adaptation est réalisée à chaque fois avec une seule phrase d'adaptation. En raison de la quantité faible de données d'adaptation généralement disponible dans une phrase d'adaptation, les variables d'adaptation, même d'une seule régression linéaire, peuvent donc avoir été mal estimées. Ceci peut entraîner une dégradation des performances du système. Dans le meilleur des cas, où *SMLLR* a été calibré pour estimer les paramètres d'une matrice de régression linéaire uniquement si un certain nombre de trames est associée à la classe correspondante, les performances du système stagneront. Si les phrases d'adaptation utilisées individuellement ne sont pas assez longues, cette approche sera donc inopérante.

4.2.2 Méthode efficace de cumul des statistiques pour l'adaptation incrémentale

Digalakis propose dans [28] une méthode performante de cumul des statistiques suffisantes pour les techniques de la famille de *MLLR* destinées à une adaptation incrémentale. Cette approche est basée sur la version incrémentale de l'algorithme *EM* : *IEM* [45]. Il a été montré que *IEM* permet d'accélérer la convergence de l'apprentissage au maximum de vraisemblance de *HMMs* [87], tout en réduisant de manière significative le temps de calcul [88].
Cette méthode consiste à conserver l'ensemble des modèles acoustiques d'un *SRAP* et un ensemble de statistiques suffisantes pour chaque locuteur qui est susceptible d'utiliser le système de reconnaissance automatique de la parole. Avant qu'un locuteur n'est prononcé une phrase, les modèles acoustiques Θ qui lui sont associés sont ceux du système indépendant du locuteur et ses statistiques suffisantes Γ_r et Ω_r associées à chaque gaussienne g_r, $\forall r = 1, 2, \ldots, N_G$, sont initialisées à 0, c'est-à-dire que $\Gamma_r = 0$ et $\Omega_r = \vec{0}$. Lorsqu'un locuteur prononce une phrase, des nouvelles statistiques $\bar{\Gamma}_r$ et $\bar{\Omega}_r$ sont calculées à l'aide de l'algorithme Baum-Welch pour chaque gaussienne g_r en utilisant les modèles acoustiques Θ et les données acoustiques de cette phrase. Pour chaque gaussienne g_r, les statistiques $\bar{\Gamma}_r$ et $\bar{\Omega}_r$ sont respectivement cumulées aux anciennes statistiques Γ_r et Ω_r telles que :

$$\Gamma_r = \Gamma_r + \bar{\Gamma}_r \tag{4.18}$$

$$\Omega_r = \Omega_r + \bar{\Omega}_r \tag{4.19}$$

Les statistiques Γ_r et Ω_r sont ensuite utilisées pour estimer les variables d'adaptation, qui

permettent d'adapter les modèles acoustiques Θ. Ce processus de calcul et d'accumulation des nouvelles statistiques est ainsi répété à chaque fois qu'une nouvelle phrase est prononcé par un locuteur.

4.3 Choix d'implantation

4.3.1 Construction de l'arbre des gaussiennes

Pour des raisons de rapidité et de simplicité d'utilisation, et étant donné qu'il a été montré dans [35] que le type d'arbre de gaussiennes utilisé par *SMLLR* n'a que peu d'influence sur les performances effectives du système adapté généré, nous avons choisi d'utiliser l'algorithme *LBG* pour construire l'arbre des gaussiennes utilisé par *SMLLR*. *LBG* est un algorithme rapide qui permet de générer un arbre binaire de manière descendante. Le nœud racine est tout d'abord créé. Il contient l'ensemble des gaussiennes de tous les modèles. Le centre de gravité du nœud racine est ensuite calculé, puis est perturbé pour générer deux nouveaux centres de gravité associés respectivement à deux nouveaux nœuds fils. Chaque centre de gravité permet de regrouper dans le nœud correspondant les gaussiennes similairement proche au sens d'une certaine mesure de distance. Une fois que chaque gaussienne du nœud racine est affectée dans l'un ou l'autre des nœuds fils, la procédure *K-Means* est utilisée pour affiner cette affectation. Cette procédure consiste à recalculer le centre de gravité de chaque nœud en utilisant les gaussiennes qui y sont regroupées, à réaffecter les gaussiennes du nœud père dans l'un des nœud fils et à répéter ces opérations jusqu'à ce que cette affectation soit stable pour chaque nœud fils. Ce processus de perturbation et d'affectation des gaussiennes est itéré pour chaque nœud généré jusqu'à obtenir les nœuds feuilles à la profondeur souhaitée. Nous donnons dans l'annexe B une description plus détaillée de cet algorithme.

4.3.2 Définition des classes de régression

M.J.F. Gales a également montré dans [35] que des performances similiaires sont obtenues quelque soit le critère choisi pour constituer l'ensemble des classes de régression à partir de l'arbre de gaussiennes. Pour les mêmes raisons que précédemment, nous avons choisi d'implanter une technique rapide et simple d'utilisation pour sélectionner l'ensemble des classes de régression. Il s'agit de la technique basée sur le nombre minimum de trames, introduite dans [76] et décrite dans le paragraphe 4.1.2.2, page 62.

4.4 Evaluations expérimentales

Nous présentons dans cette section les résultats obtenus avec *SMLLR*, en utilisant les conditions expérimentales décrites dans le chapitre 3.

La qualité des estimations dans *SMLLR*, et donc les performances effectives du système adapté généré, sont influencées par trois paramètres :

- le nombre d'itérations β_{SMLLR}, qui détermine le nombre de fois où l'algorithme *EM* est utilisé,
- le nombre minimum de trames α_{SMLLR}, qui définit la quantité minimale de vecteurs d'observations extraits des données d'adaptation et requis pour estimer de manière robuste une matrice de transformation,
- la structure d'une matrice de transformation σ_{SMLLR}, qui peut être pleine avec biais ($\sigma_{SMLLR} = p+$), pleine sans biais ($\sigma_{SMLLR} = p-$), diagonale avec biais ($\sigma_{SMLLR} = d+$) ou diagonale sans biais ($\sigma_{SMLLR} = d-$).

Nous avons réalisé plusieurs expériences en mode par lot supervisée en utilisant *SMLLR* avec des paramétrages différents de β_{SMLLR}, α_{SMLLR} et σ_{SMLLR}, afin de déterminer l'importance relative de chacun d'eux en fonction de la quantité disponible de données d'adaptation. Le meilleur paramétrage fût ensuite utilisé pour une adaptation incrémentale non supervisée du système indépendant du locuteur.

La figure 4.3 montre les résultats des expériences réalisées avec *SMLLR* en mode par lot supervisé, en faisant varier le nombre minimum de trames α_{SMLLR} et en employant une seule itération de *EM* et des matrices pleines avec biais.
Cette figure met en évidence l'existence d'un nombre minimum idéal de trames $\alpha_{\hat{SMLLR}}$, qui permet d'estimer de manière robuste les paramètres des matrices de régression linéaire quelle que soit la quantité de données d'adaptation disponibles. Les meilleures performances sont ainsi obtenues pour une valeur de α_{SMLLR} comprise entre 800 et 1500. Or nous avons utilisé des vecteurs d'observations de 35 coefficients. Les matrices de régressions linéaires pleines avec biais employées comportent ainsi chacune $35 \times 36 = 1260$ paramètres à estimer.
Il semblerait donc que α_{SMLLR} puisse être déterminé en fonction du nombre de paramètres d'une matrice de régression linéaire.

Lorsque le nombre minimum de trames α_{SMLLR} est inférieur à 800 et que la quantité disponible de données d'adaptation est faible, les paramètres des transformations sont mal estimés, ce qui cause une dégradation des performances du système adapté par rapport à celles du *SIL*. Ce phénomène est visible sur la figure lorsqu'une phrase d'adaptation est utilisée alors que α_{SMLLR} est petit (pour $\alpha_{SMLLR} = 0$ ou 500). Dans le cas où $\alpha_{SMLLR} = 0$, toutes les classes de régression du niveau des feuilles de l'arbre de gaussiennes sont créées. Certaines des matrices de régression sont donc très mal estimées puisque très peu de données d'adaptation y sont associées, ce qui engendre une forte dégradation des performances. Cette dégradation persiste jusqu'à ce que chaque classe de régression soit dotée d'une quantité suffisante de données d'adaptation (plus de 50 phrases) pour pouvoir estimer de manière robuste la matrice de régression associée. D'autre part, lorsque la quantité de données d'adaptation est faible et que la valeur de α_{SMLLR} est trop grande (cas où $\alpha_{SMLLR} = 1500$ avec moins de 3 phrases d'adaptation), aucune matrice de régression n'est estimée : le *SIL* n'est donc pas adapté, alors qu'une amélioration des performances est pourtant possible. Nous pouvons observer enfin que dans le cas où la quantité disponible de données d'adaptation est très importante, c'est-à-dire lorsque plus de 100 phrases sont disponibles, la valeur donnée à α_{SMLLR} n'affecte alors que très marginalement les

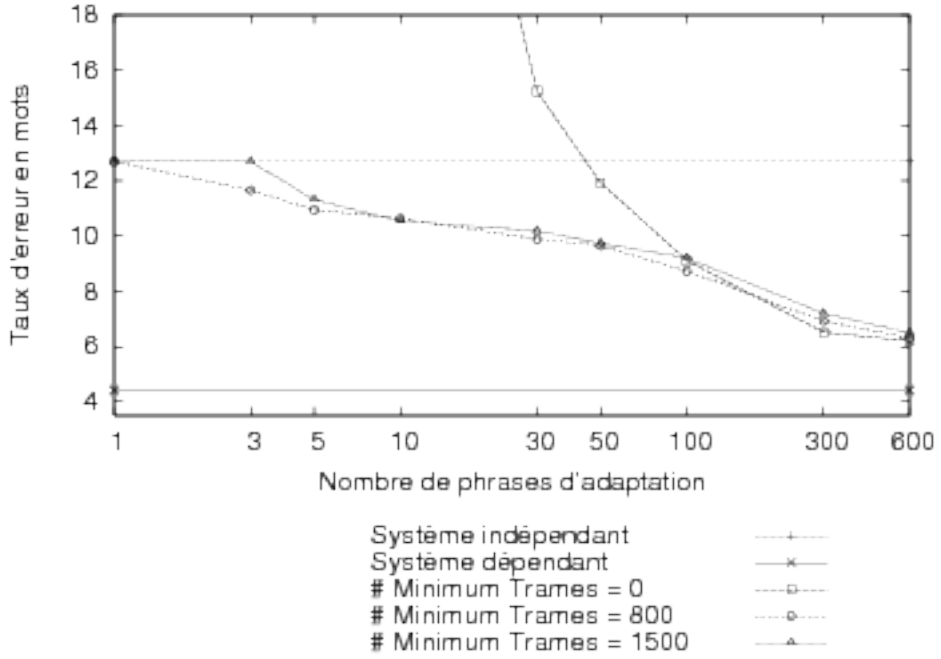


FIGURE 4.3 – Résultats *SMLLR* - Adaptation par lot supervisée - Variation du nombre minimum α_{SMLLR} de trames requises pour estimer de manière robuste une régression linéaire (matrice pleine avec biais)

performances du système adapté. Par la suite, nous avons décidé d'utiliser *SMLLR* avec $\alpha_{SMLLR} = 800$. Cette valeur s'est en effet révélée être la meilleure quelle que soit le nombre de phrases d'adaptation utilisées.

La figure 4.4 montre l'influence sur les performances du système adapté du nombre d'itérations de *EM* employées dans *SMLLR* en fonction du nombre de phrases d'adaptation utilisées. Des matrices pleines avec biais fûrent estimées pour ces expériences.

Lorsque la quantité de données d'adaptation est faible, une seule itération suffit pour obtenir les meilleures estimations des paramètres des matrices de régression linéaire. Dans le cas où trop d'itérations sont utilisées, les paramètres des matrices sont estimés trop précisément : ils sont trop bien appris, si bien que les modèles acoustiques du système adapté sont désormais capables de reconnaître parfaitement les phrases d'adaptation mais moins efficacement les phrases prononcées ultérieurement par le locuteur (cas où 10 itérations sont employées avec 3 phrases d'adaptation). En revanche, lorsque la quantité de données d'adaptation disponibles est grande, l'importante quantité d'informations qui y est contenue ne peut plus être pris en compte par l'ensemble des régressions linéaires lorsque leurs paramètres sont estimés grossièrement avec une seule itération. Dans ce cas, une grande précision dans l'estimation des paramètres des matrices de régression est alors requise, ce qui ne peut être obtenue qu'en employant plusieurs itérations de *EM*. Ce phénomène est observable sur la figure lorsque 600 phrases d'adaptation sont utilisées : les

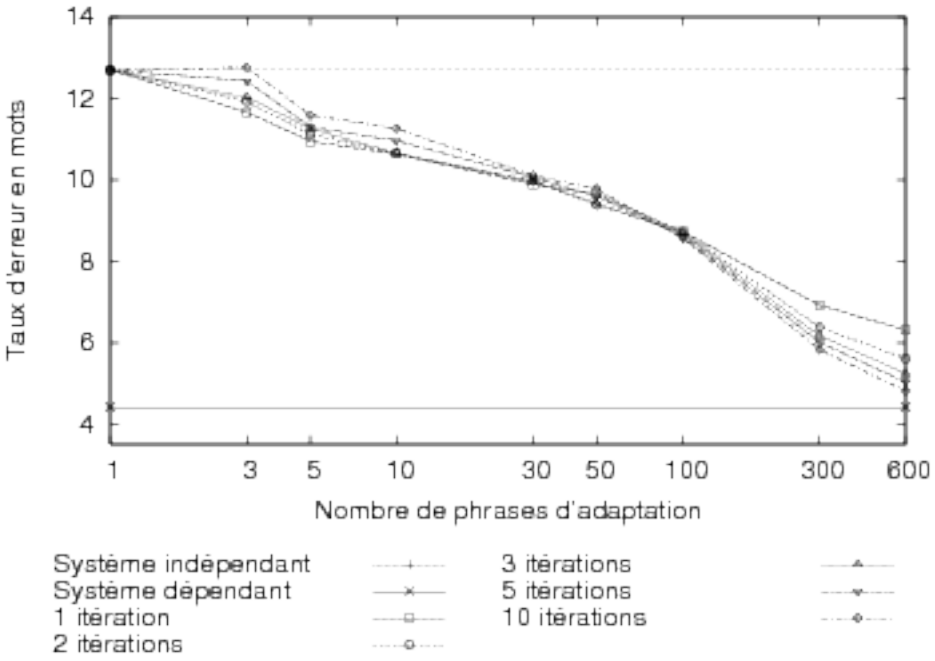


FIGURE 4.4 – Résultats *SMLLR* - Adaptation par lot supervisée - Variation du nombre d'itérations β_{SMLLR}

performances du système adapté sont dans ce cas proportionnelles au nombre d'itérations de *EM* utilisées.

Nous montrons dans la figure 4.5 comment les performances du système adapté sont affectées en fonction de la structure matricielle des régressions linéaires choisie, lorsque peu de données d'adaptation sont employées.

L'utilisation de matrices de régression linéaire diagonales à la place de matrices pleines n'améliore que très marginalement les performances du système adapté lorsque peu de données d'adaptation sont disponibles. Par contre, lorsque la quantité de données d'adaptation commence à être importante (plus de 5 phrases), l'utilisation de matrices pleines doit absolument être privilégiée. Ce phénomène peut s'expliquer par le fait que seule une quantité très réduite d'informations contenues dans les données d'adaptation peut être prise en compte par des matrices diagonales.

La figure 4.6 révèle l'efficacité de *SMLLR* lorsqu'elle est employée pour une adaptation par lot supervisée ou pour une adaptation incrémentale non supervisée. La méthode performante de récolte des statistiques suffisantes proposée par Digalakis [28] et présentée dans le paragraphe 4.2.2 fût utilisée pour cette expérience en mode incrémental.
Les performances obtenues en mode incrémental non supervisée sont similaires à celles obtenues avec une adaptation par lot supervisée. En mode incrémental, *SMLLR* améliore toutefois le taux de reconnaissance du système indépendant du locuteur de manière conti-

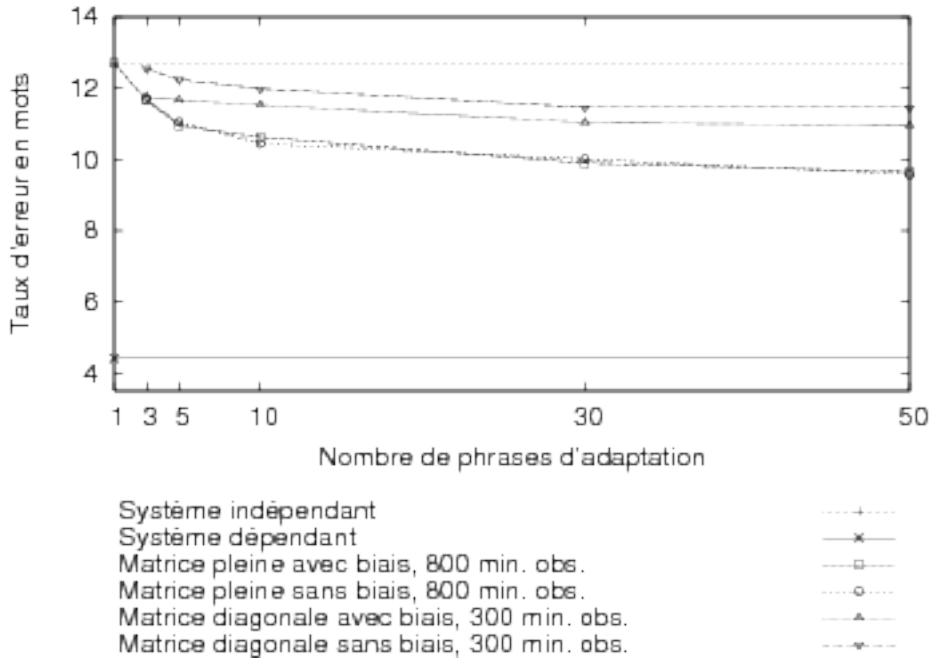


FIGURE 4.5 – Résultats *SMLLR* - Adaptation par lot supervisée - Sélection de la structure matricielle σ_{SMLLR} des régressions linéaires

nue, au fur et à mesure que le locuteur prononce de nouvelles phrases. Dans l'un ou l'autre de ces modes, *SMLLR* permet d'améliorer le taux de reconnaissance du système indépendant du locuteur de l'ordre de 8% lorsque trois phrases d'adaptation ont été utilisées, et de l'ordre de 24% lorsque 50 phrases ont été utilisées.

4.5 Conclusions

Nous avons présenté dans ce chapitre les principes théoriques fondamentaux de *SMLLR*, ainsi que les résultats obtenus avec cette méthode en utilisant le moteur de reconnaissance *ESPERE*, dans le cas d'une adaptation par lot supervisée et dans le cas d'une adaptation incrémentale non supervisée.

SMLLR s'est révélé capable d'améliorer efficacement les performances d'un système indépendant du locuteur lorsque plus de trois phrases d'adaptation sont disponibles, aussi bien en mode d'adaptation par lot qu'en mode d'adaptation incrémental. L'amélioration relative du taux de reconnaissance en mots du système adapté par rapport à celui du *SIL* est de l'ordre de 8% en utilisant 3 phrases d'adaptation, de l'ordre de 16% en utilisant 10 phrases et peut atteindre 50% en utilisant 600 phrases d'adaptation. Cependant, cette technique s'est montrée inefficace dans le cas d'une adaptation rapide, avec une seule phrase d'adaptation, ce qui représente moins de trois secondes de parole. Dans ce cas, les paramètres d'une matrice de régression linéaire ne peuvent effectivement pas être estimés

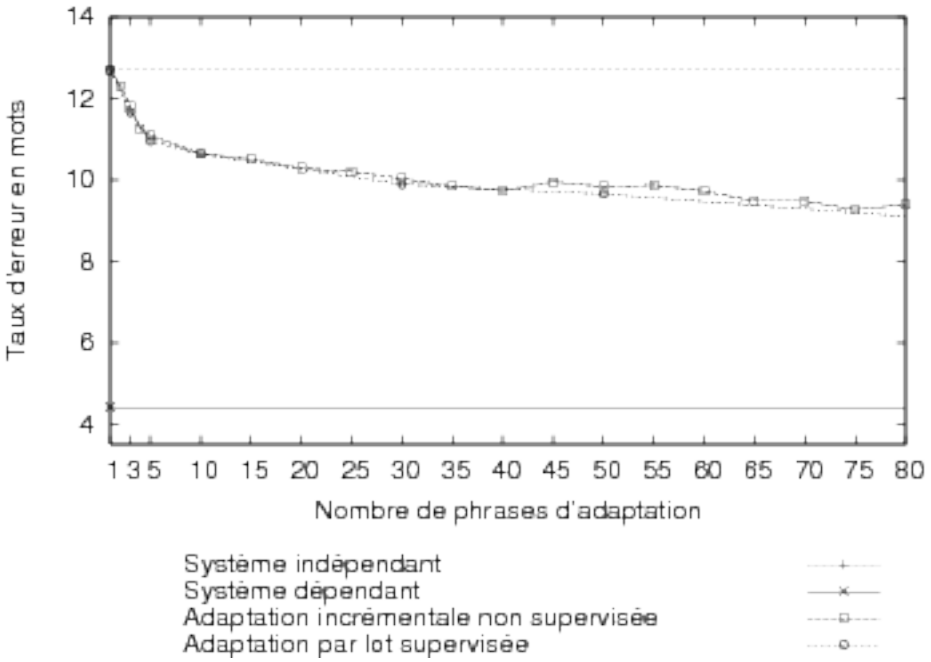


FIGURE 4.6 – Résultats *SMLLR* - Adaptation incrémentale non supervisée et adaptation par lot supervisée

de manière robuste, si l'on considère la faible quantité de données d'adaptation pouvant être extraite d'une seule phrase.

Ce chapitre nous a également permis de mieux comprendre l'influence des paramètres de *SMLLR* sur les performances effectives du système adapté généré. Ces paramètres sont α_{SMLLR}, β_{SMLLR} et σ_{SMLLR}. Ils représentent respectivement le nombre minimum de trames requis pour estimer de manière robuste les paramètres d'une régression linéaire, le nombre de fois où *EM* est utilisé pour estimer les variables d'adaptation et la structure matricielle des régressions linéaires. En ce qui concerne α_{SMLLR}, sa valeur peut être déterminée proportionnellement en fonction du nombre de paramètres d'une matrice de régression linéaire. Enfin, pour obtenir une amélioration significative des performances du système indépendant du locuteur en utilisant *SMLLR*, une seule itération de *EM* ainsi que des matrices de régressions linéaires pleines avec biais seront préférentiellement utilisées.

Chapitre 5

Structural Maximum A Posteriori (SMAP)

Nous continuons l'étude initiée dans le chapitre précédent qui porte sur les techniques d'adaptation des modèles acoustiques les plus populaires, en examinant ici une technique représentative de la famille des techniques bayésiennes : la technique d'adaptation *SMAP* [96; 97; 98].
SMAP est fondée sur une formulation bayésienne qui incorpore dans le processus d'adaptation des informations *a priori* sur les moyennes des gaussiennes. Ces informations, employées pour une estimation des moyennes au maximum *a posteriori*, sont particulièrement utiles dans le cas où la quantité de données d'adaptation est faible, dans le sens où elles permettent de fournir une estimation plus fiable qu'avec le critère du maximum de vraisemblance.
Dans cette méthode, les moyennes des gaussiennes des modèles acoustiques sont des vecteurs dont les coefficients sont considérés comme étant des variables aléatoires. A chaque vecteur de moyenne est ainsi associée une densité de probabilité *a priori*, qui est employée dans l'estimation au *MAP* des moyennes des gaussiennes.
SMAP améliore les performances obtenues par la technique *MAP* [42] lorsque la quantité disponible de données d'adaptation est faible. Etant donné qu'une estimation au maximum *a posteriori* est asymptotiquement équivalente à une estimation au maximum de vraisemblance, l'approche traditionnelle *MAP* fournit des performances rivalisant avec celles obtenues par un système dépendant du locuteur lorsque la quantité disponible de données d'adaptation devient importante. Cependant, cette convergence est (très) lente, car seules les moyennes des gaussiennes pour lesquelles des données ont été observées sont adaptées. *SMAP* pallie ce défaut en utilisant une structure hiérarchique qui rend possible l'estimation des moyennes des gaussiennes pour lesquelles aucune observation n'est associée.

Nous aborderons dans la première partie de ce chapitre les idées sous-jacentes à *SMAP*. Nous préciserons notamment comment la structure hiérarchique est construite puis utilisée pour estimer les paramètres des vecteurs de biais qui sont appliquées aux moyennes des gaussiennes des modèles acoustiques du système indépendant du locuteur. La deuxième section de ce chapitre sera consacrée aux choix que nous avons fait concernant l'implanta-

tion de *SMAP*. Nous présenterons en particulier l'algorithme retenu pour construire l'arbre de gaussiennes. Nous donnerons ensuite dans la troisième section les résultats des expériences d'évaluation de la version implantée de *SMAP*, ménées à l'aide du corpus *RM* et du moteur de reconnaissance *ESPERE*. Suivront dans la dernière section nos conclusions sur les choix de paramétrage qui nous ont semblé pertinents pour utiliser efficacement *SMAP* dans le cas d'une adaptation par lot supervisée d'une part, et dans le cas d'une adaptation incrémentale non supervisée d'autre part.

5.1 Approche classique

La technique d'adaptation *Structural Maximum A Posteriori*, proposée par K. Shinoda et C.-H. Lee dans [96; 97; 98], suppose qu'il existe une hiérarchie de densités *a priori*. Cette hiérarchie sert de support à l'estimation au *MAP* des paramètres de densités normalisées, qui représentent une approximation d'un ensemble de distributions de gaussiennes. Les paramètres de ces densités sont utilisés finalement pour adapter les vecteurs de moyenne des gaussiennes du *SIL*.

5.1.1 Approximation des distributions de gaussiennes

Un moyen simple d'adapter les moyennes μ_r des gaussiennes g_r du *SIL*, pour $r = 1, 2, \cdots, N_G$, à partir des vecteurs d'observations $O = (o_1, o_2, \cdots, o_T)$, est de considérer le processus où chaque vecteur d'observation o_t est transformé en un vecteur $y_{r,t}$ pour chaque gaussienne g_r, tel que :

$$y_{r,t} \;=\; \sigma_r^{-1/2}\,(o_t \;-\; \mu_r) \;\text{ pour } t = 1, \ldots, T \;\text{ et }\; r = 1, \ldots, N_G. \tag{5.1}$$

où T est le nombre total d'observations et N_G le nombre total de gaussiennes.
Lorsqu'il n'existe pas de différences acoustiques entre les données d'apprentissage et les données d'adaptation, la fonction de densité de probabilité pour $Y_r = (y_{r,1}, y_{r,2}, \cdots, y_{r,T})$ est la distribution normale $\mathcal{N}(Y_r|\,\vec{0}, I)$ où $\vec{0}$ est un vecteur dont tous les coefficients sont nuls et I est la matrice identité. Dans le cas contraire, la fonction de densité de probabilité pour Y_r a la forme $\mathcal{N}(Y_r|\nu_r, \eta_r)$, où $\nu_r \neq \vec{0}$ et $\eta_r \neq I$ représentent respectivement le vecteur de biais et la matrice de rotation nécessaires pour compenser la distortion acoustique entre les données d'apprentissage et les données d'adaptation. Les auteurs supposent que $h_r = \mathcal{N}(Y_r|\nu_r, \eta_r)$, qu'ils nomment fonction de densité de probabilité *normalisée* de la gaussienne g_r, modélise fidèlement les différences acoustiques entre les données d'apprentissage et les données d'adaptation. Elle permet ainsi de faire une approximation de la gaussienne g_r. La relation existant entre la fonction de densité de probabilité initiale et la fonction de densité de probabilité normalisée de la gaussienne g_r s'exprime alors par :

$$\mathcal{N}(o_t|\mu_r, \sigma_r) \;=\; \frac{\mathcal{N}(y_{r,t}|\nu_r, \eta_r)}{\sigma_r^{1/2}} \tag{5.2}$$

Dans notre cas, seuls les vecteurs de moyenne des gaussiennes sont adaptés. Ainsi, seul le paramètre ν_r de la fonction de densité de probabilité nécessite d'être estimé. Une fois estimé au maximum de vraisemblance, ν_r est utilisé pour modifier la moyenne μ_r de la gaussienne r selon l'équation :

$$\tilde{\mu}_r \;=\; \mu_r \;+\; \sigma_r^{1/2}\,\nu_r \tag{5.3}$$

L'estimation au maximum de vraisemblance du paramètre ν_r de la fonction de densité de probabilité normalisée de la gaussienne r est réalisée à l'aide de l'algorithme *EM*. Puisque les probabilités de transition et les poids associés aux gaussiennes restent inchangés, la fonction auxiliaire $Q(.|.)$ est donnée par :

$$\begin{aligned} Q(\hat{\Theta}|\Theta) &= \sum_{t=1}^{T}\sum_{r=1}^{N_G} \gamma_r(t)\; log\, \mathcal{N}(o_t|\mu_r,\sigma_r) \\ &= \sum_{t=1}^{T}\sum_{r=1}^{N_G} \gamma_r(t)\; log\, \frac{\mathcal{N}(y_{r,t}|\nu_r,\eta_r)}{|\sigma_r^{1/2}|} \end{aligned} \tag{5.4}$$

où Θ est l'ensemble des paramètres des modèles acoustiques du système indépendant du locuteur, et $\hat{\Theta}$ est l'ensemble des paramètres des modèles acoustiques du système adapté. $\gamma_r(t)$ représente la probabilité *a posteriori* de se trouver dans la gaussienne g_r au temps t en sachant que la séquence d'observation O a été générée.
L'estimée au maximum de vraisemblance du paramètre $\nu_r = \tilde{\nu}_r$ est obtenue après maximisation de l'équation 5.4 et est donnée par :

$$\tilde{\nu}_r = \frac{\sum_{t=1}^{T}\; \gamma_r(t)\; y_{r,t}}{\sum_{t=1}^{T}\; \gamma_r(t)} \tag{5.5}$$

En utilisant ces fonctions de densité de probabilité normalisées, l'adaptation des moyennes des gaussiennes du *SIL* est réalisée en estimant pour chaque gaussienne g_r le paramètre ν_r de la fonction de densité de probabilité normalisée qui y est associée, puis en réestimant la moyenne μ_r de chaque gaussienne g_r selon l'équation 5.3.

Avec cette méthode, toutefois, on constate rapidement que le nombre de paramètres qui peuvent être estimés de manière robuste, au sens du maximum de vraisemblance, nécessite une quantité importante de données d'adaptation.
Pour réduire le nombre de paramètres à estimer, *SMAP* suppose que les différences acoustiques peuvent être modélisées par un nombre N de densités normalisées largement inférieur au nombre total N_G de gaussiennes du *SIL*. Pour réaliser cela, l'ensemble G des gaussiennes du *SIL* est divisé en N sous-ensemble $G_1, G_2, \cdots, G_N$. Chaque sous-ensemble $G_i \quad \forall i = 1, 2, \cdots, N$ regroupe une partie des gaussiennes de G et caractérise ainsi une

partie de l'espace acoustique représenté par les moyennes des gaussiennes du *SIL*. Afin d'obtenir N fonctions de densité de probabilité normalisées, il suffit donc d'associer à chaque sous-ensemble G_i une fonction de densité de probabilité normalisée h_i. Chaque fonction de densité de probabilité normalisée représente alors une approximation d'un ensemble de distributions de gaussienne. La moyenne μ_r de chaque gaussienne d'un nœud G_i peut alors être mis à jour selon la formule 5.3 en utilisant le paramètre ν_i de la densité normalisée associée à ce nœud.
Selon les auteurs, une meilleure estimation des paramètres de cette densité peut cependant être obtenue selon le critère du maximum *a posteriori*. L'utilisation de ce critère nécessite la définition d'une fonction de densité de probabilité *a priori* utilisée pour l'estimation des paramètres de la fonction de densité de probabilité normalisée. Pour faciliter la mise en œuvre de cette approche, les auteurs ont considéré l'utilisation d'une structure hiérarchique (sous la forme d'un arbre de gaussiennes), dont chaque nœud regroupe un ensemble de gaussiennes auquel est associée une fonction de densité de probabilité normalisée. Deux raisons justifient le choix d'un arbre de gaussiennes. D'une part, un arbre de gaussiennes modélise la topologie de l'espace acoustique représenté par les moyennes des gaussiennes du *SIL*, en regroupant dans chaque nœud les gaussiennes qui sont relativement semblables au sens d'une certaine mesure de distance. A cet égard, cela permet ainsi de s'assurer qu'une fonction de densité de probabilité normalisée constitue une approximation d'un ensemble de gaussiennes relativement proches selon la mesure de distance choisie. D'autre part, un arbre de gaussiennes offre une exploitation naturelle et aisée des informations *a priori* nécessaires à l'estimation des paramètres d'une fonction de densité de probabilité associée à un noeud. En effet, une hiérarchie de fonctions de densité de probabilité peut facilement être établie en utilisant la structure intrinsèque de l'arbre : les paramètres d'une densité associée à un nœud i situé au niveau k dans l'arbre peuvent alors être utilisées pour estimer les paramètres de la densité liée à l'un des noeuds fils de i situé au niveau $k+1$. En outre, en considérant que tous les noeuds feuilles de l'arbre ne contiennent qu'une seule gaussienne, les paramètres de la densité normalisée associée à un de ces nœuds feuilles permet alors la mise à jour du vecteur de moyenne de la gaussienne correspondante selon la formule 5.3.

5.1.2 Hiérarchie de fonctions de densité de probabilité *a priori*

Les fonctions de densité de probabilité *a priori* dans *SMAP* sont réparties dans une structure hiérarchique arborescente. Le nœud racine de cet arbre contient l'ensemble des gaussiennes de tous les modèles acoustiques et chaque nœud feuille contient une seule gaussienne. Cet arbre peut être défini par le couple (G, P). G représente l'ensemble des nœuds tel que $G = \{ G^{(1,1)}, G^{(2,1)}, G^{(2,2)}, \ldots, G^{(2,M_2)}, \ldots, G^{(K,1)}, \ldots, G^{(K,M_K)} \}$ où K est la profondeur de l'arbre, M_k, pour $k = 1, 2, \ldots, K$, représente le nombre de nœuds au niveau k et $G^{(1,1)}$ est le nœud racine. P représente l'ensemble des parents des nœuds à partir du niveau 3, tel que $P = \{(p^{(3,1)}, p^{(3,2)}, \ldots, p^{(3,M_3)}), \ldots, (p^{(k,1)}, p^{(k,2)}, \ldots, p^{(k,M_k)}), \ldots, (p^{(K,1)}, p^{(K,2)}, \ldots, p^{(K,M_K)})\}$ où $p^{(k,i)}$ indique le numéro du nœud situé au niveau $k-1$ qui est père du nœud $G^{(k,i)}$.
Chaque nœud $G^{(k,i)}$ du niveau k, pour $i = 1, 2, \ldots, M_k$, regroupe un ensemble de gaussiennes qui est approximé par une fonction de densité de probabilité normalisée $h^{(k,i)}$.

Soient $\lambda_i^{(k)} = (\nu_i^{(k)}, \eta_i^{(k)})$ les paramètres de $h^{(k,i)}$. Ces paramètres sont estimés au maximum *a posteriori* (*MAP*) des données d'adaptation, c'est-à-dire :

$$\hat{\lambda}_i^{(k)} = \underset{\lambda_i^{(k)}}{\text{argmax}} \; p(\lambda_i^{(k)}|Y) \tag{5.6}$$

où Y est la séquence des vecteurs d'observations issus des données d'adaptation.
Afin d'obtenir une estimation plus précise des paramètres $\lambda_i^{(k)}$, les auteurs introduisent la contrainte supplémentaire que les paramètres de $\lambda_i^{(k)}$ sont estimés sur la base des connaissances des paramètres de la densité normalisée du nœud père $G^{(k-1,p^{(k,i)})}$, c'est-à-dire de $\lambda_{p^{(k,i)}}^{(k-1)}$. Une manière simple de satisfaire cette contrainte est de considérer une densité *a priori* de la forme $p(\lambda_i^{(k)}|\lambda_{p^{(k,i)}}^{(k-1)})$. Les hyperparamètres de ces densités *a priori* $p(\lambda_i^{(k)}|\lambda_{p^{(k,i)}}^{(k-1)})$ sont dérivés à partir des estimations au *MAP* $\hat{\lambda}_{p^{(k,i)}}^{(k-1)}$ de $\lambda_{p^{(k,i)}}^{(k-1)}$. L'équation (5.6) se réécrit alors comme :

$$\begin{aligned} \hat{\lambda}_i^{(k)} &= \underset{\lambda_i^{(k)}}{\text{argmax}} \; p(\lambda_i^{(k)}|\hat{\lambda}_{p^{(k,i)}}^{(k-1)}, Y) \\ &= \underset{\lambda_i^{(k)}}{\text{argmax}} \; \frac{p(Y|\lambda_i^{(k)}, \hat{\lambda}_{p^{(k,i)}}^{(k-1)})\, p(\lambda_i^{(k)}|\hat{\lambda}_{p^{(k,i)}}^{(k-1)})}{p(Y)} \end{aligned} \tag{5.7}$$

en considérant que $\hat{\lambda}_0^{(0)} = \lambda_0^{(0)} = (\vec{0}, I)$ sont les paramètres de la densité normalisée au nœud racine.
En supposant que $p(Y|\lambda_i^{(k)}, \hat{\lambda}_{p^{(k,i)}}^{(k-1)})$ ne dépend pas de $\hat{\lambda}_{p^{(k,i)}}^{(k-1)}$ et puisque $p(Y)$ n'est pas fonction de $\lambda_i^{(k)}$, l'équation 5.7 peut se réecrire ainsi :

$$\hat{\lambda}_i^{(k)} = \underset{\lambda_i^{(k)}}{\text{argmax}} \; p(Y|\lambda_i^{(k)})\, p(\lambda_i^{(k)}|\hat{\lambda}_{p^{(k,i)}}^{(k-1)})$$

Nous donnons dans la section suivante les équations qui permettent d'estimer les paramètres des densités normalisées et de réestimer les moyennes des gaussiennes des modèles acoustiques.

5.1.3 Estimation des moyennes des gaussiennes

Considérons le cas de l'estimation du vecteur de moyenne μ d'une gaussienne g présente dans le nœud feuille G_K. Pour atteindre cette gaussienne particulière en parcourant l'arbre, K nœuds seront visités. Nommons $G_1, G_2, \ldots, G_K$ ces nœuds, auxquels sont respectivement associées les fonctions de densité de probabilité $h_1, h_2, \ldots, h_K$. $\lambda_i = (\nu_i, \eta_i)$ sont les paramètres de la fonction de densité de probabilité h_i. $\nu_1, \nu_2, \ldots, \nu_K$ sont respectivement les paramètres des fonctions de densité de probabilité $h_1, h_2, \ldots, h_K$ qui sont utilisés pour mettre à jour la moyenne de la gaussienne g.
Le processus d'estimation selon *SMAP* de ce vecteur de moyenne μ peut être illustré par la figure 5.1.

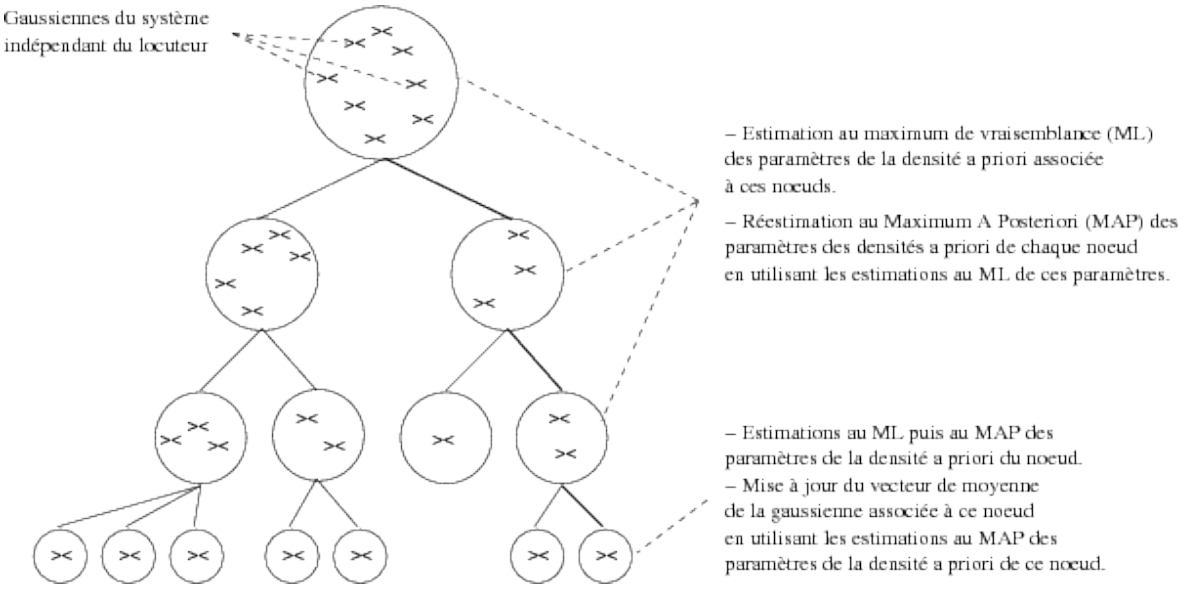


FIGURE 5.1 – Adaptation du vecteur de moyenne d'une gaussienne selon *SMAP*

A chaque nœud G_k situé au niveau k, de la racine ($k = 0$) jusqu'au niveau des feuilles ($k = K$), les opérations suivantes sont successivement exécutées :

1. Estimation au maximum de vraisemblance (ML) du paramètre ν_k de la densité normalisée associée à ce nœud : $\tilde{\nu}_k$.
2. Réestimation au maximum *a posteriori* de $\nu_k = \hat{\nu}_k$, en utilisant le paramètre $\tilde{\nu}_k$ estimé au maximum de vraisemblance et le paramètre $\hat{\nu}_{k-1}$ du nœud père G_{k-1}.

La moyenne μ de la gaussienne g du nœud N_K est mise à jour en utilisant l'estimée au MAP du paramètre $\nu_K = \hat{\nu}_K$.
L'estimation du paramètre ν_k au MAP dans chaque nœud G_k est réalisée à l'aide de l'équation suivante :

$$\hat{\lambda}_k \quad = \quad \underset{\lambda_k}{\text{argmax}} \;\; p(Y|\lambda_k)\, p(\lambda_k|\hat{\lambda}_{k-1}) \tag{5.8}$$

Pour pouvoir résoudre cette équation, les auteurs supposent que $p(\lambda_k|\hat{\lambda}_{k-1})$ est une densité normale de Wishart [24; 42] de la forme :

$$\begin{aligned} Wishart(\nu_k, \eta_k|\hat{\nu}_{k-1}, \hat{\eta}_{k-1}, \xi_k, \tau_k) \quad \propto \quad & |\eta_k|^{-\frac{\xi_k}{2}}\; exp\left[-\frac{\tau_k}{2}\left(\nu_k - \hat{\nu}_{k-1}\right)' {\eta_k}^{-1}\left(\nu_k - \hat{\nu}_{k-1}\right)\right] \\ & exp\left[-\frac{1}{2}\, tr(\hat{\eta}_{k-1}\, {\eta_k}^{-1})\right] \end{aligned} \tag{5.9}$$

où ν_k et $\hat{\nu}_{k-1}$ sont des vecteurs de dimension N_D, η_k et $\hat{\eta}_{k-1}$ des matrices de dimension $N_D \times N_D$, τ_k et ξ_k des scalaires représentant les hyperparamètres de la densité *a priori*, tels que $\tau_k > 0$ et $\xi_k > N_D - 1$.
L'équation (5.8) se résout alors en utilisant l'algorithme *EM*. Pour cela, la fonction auxiliaire $R(\hat{\lambda}_k, \lambda_k)$ à maximiser est définie par :

$$R(\hat{\lambda}_k, \lambda_k) = Q(\hat{\lambda}_k, \lambda_k) \; + \; log \;\; p(\lambda_k | \lambda_{k-1}) \tag{5.10}$$

avec

$$Q(\hat{\lambda}, \lambda) \;\; = \;\; \sum_{t=1}^{T} \sum_{m \in G_k} \gamma_m(t) \; log \;\; \frac{\mathcal{N}(y_{m,t} | \nu_k, \eta_k)}{|\sigma_m^{1/2}|} \tag{5.11}$$

où $\gamma_m(t)$ est la probabilité de se trouver dans la m-ième gaussienne du nœud G_k au temps t en sachant que la séquence d'observation O a été générée ; $\mathcal{N}(y_{m,t} | \nu_k, \eta_k)$ est la fonction de densité de probabilité d'observation h_k appliquée à l'observation normalisée $y_{m,t}$ et σ_m est la matrice de variances-covariances de la m-ième gaussienne du nœud G_k.
En dérivant l'équation 5.10 et en mettant la dérivée à zéro, l'estimée au *MAP* $\hat{\nu}_k$ de ν_k est obtenue par :

$$\hat{\nu}_k \;\; = \;\; \frac{\Gamma_k \; \tilde{\nu}_k \; + \; \tau_k \; \hat{\nu}_{k-1}}{\Gamma_k \; + \; \tau_k} \tag{5.12}$$

où $\Gamma_k = \sum_{t=1}^{T} \; \sum_{m \in G_k} \; \gamma_m(t)$. En outre, $\nu_1 = \vec{0}$ à la racine (niveau 1) et $\tilde{\nu}_k$ est l'estimée au *ML* de ν_k, obtenue avec l'équation :

$$\tilde{\nu}_k \;\; = \;\; \frac{\displaystyle\sum_{t=1}^{T} \sum_{m \in G_k} \gamma_m(t) \; y_{m,t}}{\displaystyle\sum_{t=1}^{T} \sum_{m \in G_k} \gamma_m(t)} \tag{5.13}$$

L'estimation $\hat{\mu}$ du vecteur de moyenne μ de la gaussienne g au nœud G_K est enfin calculée selon l'équation :

$$\hat{\mu} \;\; = \;\; \mu + (\sigma)^{1/2} \hat{\nu}_K \tag{5.14}$$

σ étant la matrice de variances-covariances de la gaussienne g.

5.2 Détermination de l'hyperparamètre τ_k

Le cadre théorique de *SMAP* décrit précédemment ne fournit pas de moyens de déterminer le paramètre τ_k associé à la fonction de densité de probabilité *a priori* de chaque nœud N_k de l'arbre.
Les auteurs préconisent de déterminer la valeur de cet hyperparamètre τ_k pour un nœud N_k en employant la même valeur τ pour tous les nœuds. La valeur optimale de τ peut alors être définie en réalisant une série d'expériences préliminaires d'adaptation. A l'issue de cette étape, la valeur τ qui a permis d'obtenir le système adapté ayant exhibé les meilleures performances est alors celle qui est ultérieurement utilisé dans *SMAP* pour adapter le *SIL* à un nouveau locuteur.
Les hyperparamètres τ_k peuvent également être estimés pendant le processus d'adaptation, selon le critère du maximum de vraisemblance [52].

5.3 Choix d'implantation

5.3.1 Construction de l'arbre des gaussiennes

La particularité de l'arbre des gaussiennes utilisé par *SMAP* réside dans le fait que chaque nœud feuille n'est constitué que d'une seule gaussienne. La méthode présentée dans l'annexe B permet de construire un arbre binaire de gaussiennes d'une certaine profondeur, ce qui implique que les noeuds feuilles peuvent regrouper plusieurs gaussiennes. Elle peut cependant être utilisée dans le cadre de *SMAP* en considérant que chaque gaussienne d'un nœud feuille i représente elle-même un nœud fils de i (figure 5.2).

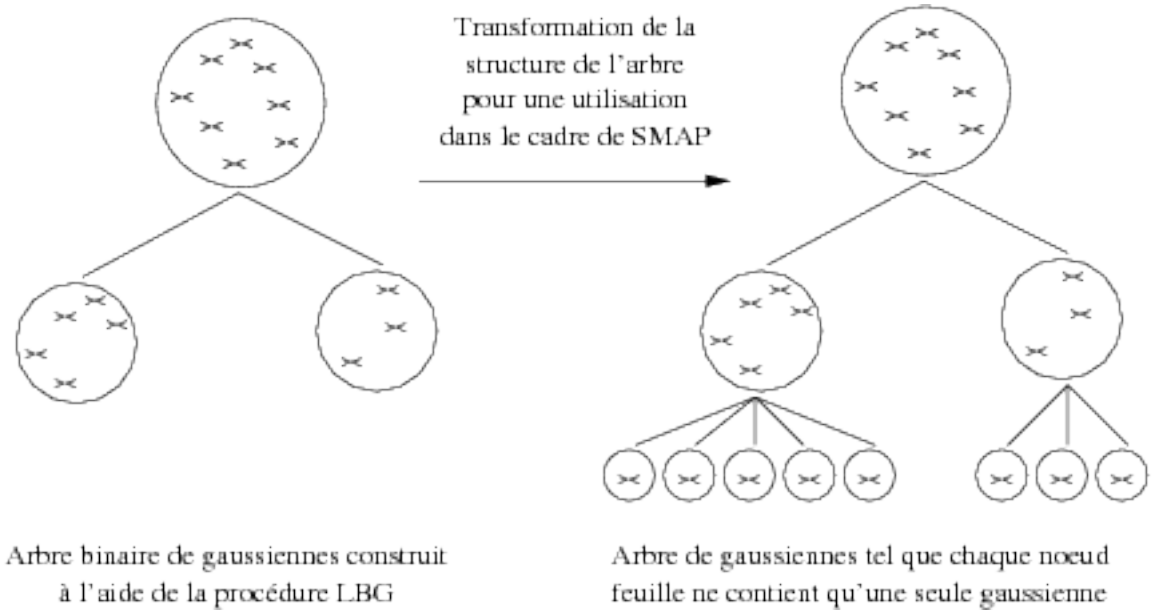


FIGURE 5.2 – Utilisation d'un arbre binaire de gaussiennes dans le cadre de *SMAP*

5.3.2 Choix de l'hyperparamètre τ_k

Nous avons choisi d'utiliser la même valeur τ pour tous les hyperparamètres τ_k associés aux nœuds de l'arbre de gaussiennes. Une série d'expériences d'adaptation fût réalisée (voir la section suivante) pour déterminer quelle est la valeur optimale τ qui permet d'obtenir un système adapté délivrant les meilleures performances. Nous avons constaté que τ dépend de la quantité de données d'adaptation disponible. Lorsque le nombre de trames d'adaptation est petit, τ doit être grand afin de prendre plus en compte les connaissances *a priori* par rapport aux estimations obtenues à l'aide des données d'adaptation. A l'inverse, lorsque la quantité disponible de données d'adaptation devient plus importante, les estimations obtenues au maximum de vraisemblance doivent prédominer par rapport à celles obtenues à l'aide des connaissances *a priori*. Ceci est possible si τ est relativement petit.
Afin d'obtenir un tel comportement d'estimation, nous avons formulé deux relations pour fixer la valeur τ de manière automatique, selon la quantité de trames d'adaptation disponibles. Soit T le nombre total de trames présentes dans le corpus d'adaptation actuellement disponible.

Relation 1 : La première relation considère que la valeur τ est très grande lorsqu'une seule trame est disponible, qu'elle est proche de 1 lorsqu'un certain nombre de trames est disponible et qu'elle est proche de zéro lorsque le nombre de trames est très important. Cette relation suggère que plus la quantité disponible de données d'adaptation est petite, plus la valeur de τ est grande. Elle est définie par :

$$\tau = \left(\frac{\alpha}{T}\right)^{\rho} \tag{5.15}$$

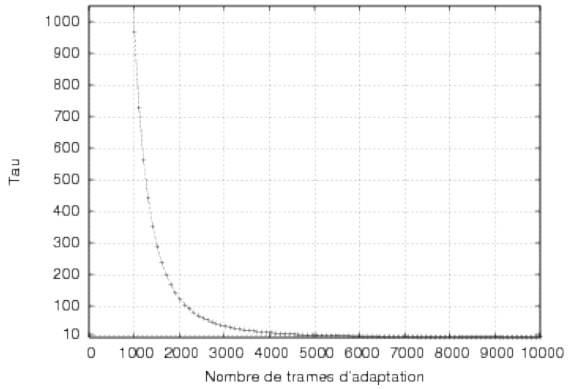
α est le nombre de trames à partir duquel τ est inférieur ou égal à 1. ρ est le facteur d'exponentiation qui détermine la valeur maximale que peut prendre τ lorsqu'une seule trame d'adaptation est disponible.

Relation 2 : La seconde relation considère que τ est égale à une certaine valeur tant que le nombre T de trames disponibles ne dépasse pas un certain seuil ω, et que le comportement de τ est celui de la relation 1 lorsque T devient supérieur à ω. A l'inverse de la relation 1, il est supposé ici que la valeur de τ est fixée à un pallier lorsque de moins en moins de données d'adaptation sont disponibles. Cette relation peut alors être représentée par :

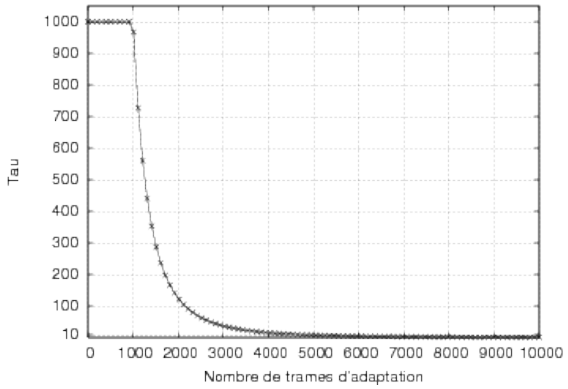
$$\tau = \begin{cases} \varphi & \text{si } T < \omega \\ \left(\frac{\alpha}{T}\right)^{\rho} & \text{si } T \geq \omega \end{cases} \tag{5.16}$$

φ est le facteur qui détermine la valeur que prend τ lorsque moins de ω trames d'adaptation sont disponibles. α est le nombre de trames à partir duquel τ est inférieur ou égal à 1. ρ est le facteur d'exponentiation qui détermine la valeur maximale que peut prendre τ lorsque ω trames d'adaptation sont disponibles.

Les relations 1 (équation 5.15) et 2 (équation 7.1) ainsi définies entre le nombre de trames



Relation 1 avec
$\alpha = 10000$, $\rho = 3$
(hypothèse 1)



Relation 2 avec
$\varphi = 1000$, $\omega = 1000$, $\alpha = 10000$, $\rho = 3$
(hypothèse 2)

FIGURE 5.3 – Relations entre le nombre de trames d'adaptation disponibles et la valeur τ affectée aux hyperparamètres des densités *a priori*

d'adaptation et la valeur τ à appliquer à chaque hyperparamètre τ_k peuvent être représentées graphiquement comme dans la figure 5.3.

Nous montrons dans la section suivante les résultats des expériences d'adaptation avec *SMAP* en fixant *a priori* la valeur τ d'une part, et en déterminant τ à l'aide de l'une ou l'autre des hypothèses 1 (équation 5.15 avec $\alpha = 10000$, $\rho = 3$) et 2 (équation 7.1 avec $\varphi = 1000$, $\omega = 1000$, $\alpha = 10000$, $\rho = 3$) d'autre part. Nous donnerons alors nos conclusions sur l'équation à utiliser pour obtenir les meilleures performances.

5.4 Evaluations expérimentales

Nous présentons dans cette section les résultats obtenus avec *SMAP* sous les conditions expérimentales exposées dans le chapitre 3. Pour toutes les expériences suivantes, nous avons utilisé le même arbre de gaussiennes, de profondeur 10, utilisé lors des expériences avec *SMLLR*.

La qualité du système adapté généré en utilisant *SMAP* est influencé par deux paramètres :
- le nombre d'itérations β_{SMAP}, qui détermine le nombre de fois où l'algorithme *EM* est utilisé,
- le choix de l'hyperparamètre τ, qui détermine l'influence de l'estimation *a priori* sur l'estimation au maximum de vraisemblance.

Nous avons réalisé plusieurs expériences en mode par lot supervisée en utilisant *SMAP* avec des paramétrages différents de β_{SMAP} et de τ afin de déterminer l'importance relative de chacun d'eux en fonction de la quantité disponible de données d'adaptation. Le meilleur

paramétrage fût ensuite utilisé pour une adaptation incrémentale non supervisée du système indépendant du locuteur.

La figure 5.4 montre les résultats des expériences réalisées avec *SMAP* en mode par lot supervisé, en faisant varier τ, déterminée *a priori*, et en employant une seule itération de *EM*.

Cette figure révèle que le choix de l'hyperparamètre τ dépend de la quantité de données d'adaptation disponible. Lorsque beaucoup de phrases d'adaptation sont disponibles (c'est-à-dire pour plus de 100 phrases), les meilleures performances sont obtenues en utilisant une valeur faible de τ ($\tau = 1$ ou $\tau = 0.1$). Lorsque peu de phrases d'adaptation sont disponibles (moins de 10 phrases), les meilleurs performances sont obtenues en employant une grande valeur pour τ ($\tau = 1000$). Entre 10 et 100 phrases d'adaptation, une valeur de τ comprise entre 0.1 et 10 permet d'obtenir les meilleures performances, par rapport aux autres valeurs de τ qui furent testées.

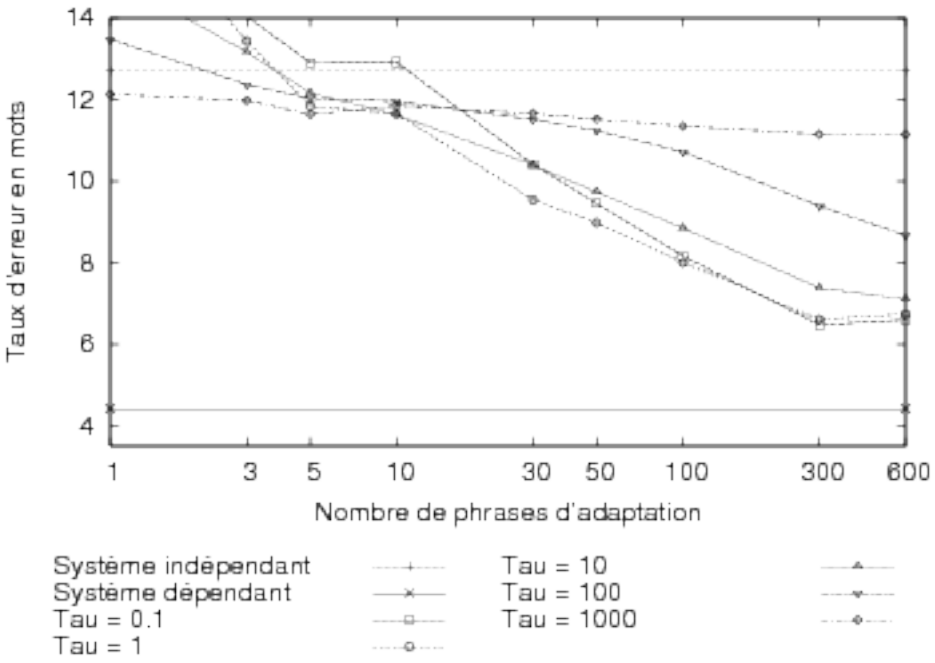


FIGURE 5.4 – Résultats *SMAP* - Adaptation par lot supervisée - Variation de τ fixée *a priori*

Afin de déterminer de manière automatique la valeur de τ en fonction de la quantité de données d'adaptation disponible, nous avons formulé deux hypothèses (voir paragraphe 5.3.2). Ces deux hypothèses permettent de reproduire approximativement le comportement idéal de τ. Nous donnons dans la figure 5.5 les résultats de *SMAP* en déterminant τ d'une part selon l'hypothèse 1, d'autre part selon l'hypothèse 2.

Les performances obtenues en utilisant l'une ou l'autre hypothèse pour déterminer la valeur de τ sont assez similaires. La relation 7.1 définie sous l'hypothèse 2 permet toutefois de fournir de meilleures performances lorsqu'une seule phrase est disponible. Ceci indique que la valeur de τ ne doit pas être fixée trop grande lorsque très peu de données d'adaptation sont disponibles. En effet, une valeur trop grande de τ rend négligeable l'estimation au maximum de vraisemblance par rapport à l'estimation *a priori*, si bien que les données d'adaptation ne sont pas du tout employées dans l'estimation. L'hypothèse 2 met en évidence l'importance de prendre en compte, même faiblement, l'estimation au maximum de vraisemblance lorsque très peu de données d'adaptation sont disponibles.

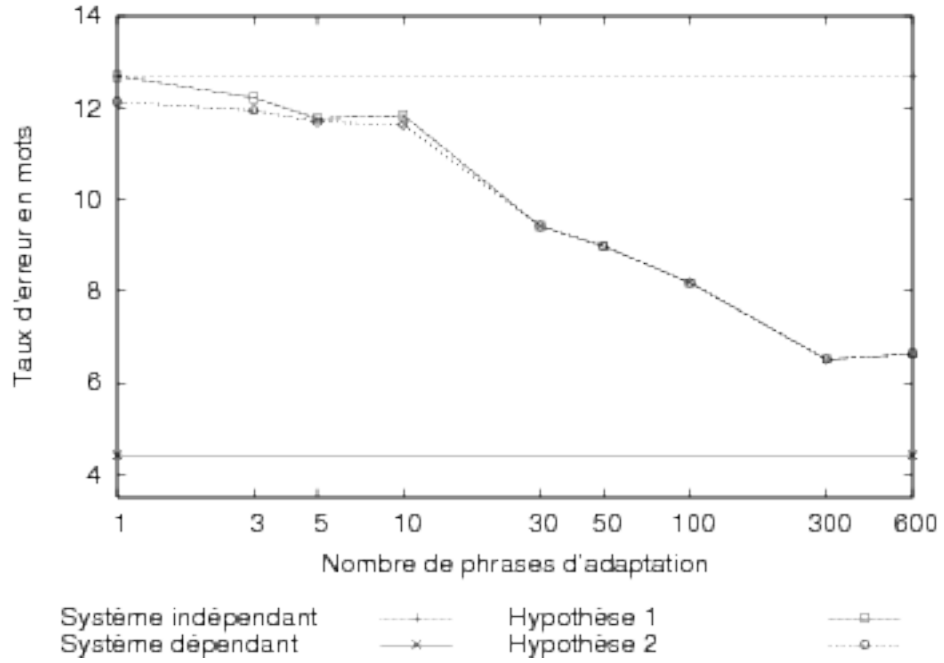


FIGURE 5.5 – Résultats *SMAP* - Adaptation par lot supervisée - τ déterminée selon les hypothèses 1 et 2

La figure 5.6 montre l'influence du nombre d'itérations de *EM* employées dans *SMAP* sur les performances du système adapté, en fonction du nombre de phrases d'adaptation utilisées. τ fut déterminée automatiquement selon la relation 7.1.

Lorsque la quantité de données d'adaptation est faible, une seule itération suffit pour obtenir les meilleures estimations des variables d'adaptation (les coefficients des vecteurs de biais). Dans le cas où trop d'itérations sont utilisées, les moyennes des gaussiennes sont "surapprises", si bien que les modèles acoustiques du système adapté sont désormais capables de reconnaître parfaitement les phrases d'adaptation mais moins efficacement les phrases prononcées ultérieurement par le locuteur. C'est le cas, par exemple, lorsque 10 itérations sont employées avec une phrase d'adaptation. En revanche, lorsque la quantité de données d'adaptation disponibles est grande, l'importante quantité d'informations qui

y est contenue peut être plus efficacement exploitée lorsque plusieurs itérations sont utilisées, plutôt qu'une seule. Dans ce cas, une grande précision dans l'estimation des moyennes des gaussiennes est alors requise, ce qui ne peut être obtenue qu'en employant plusieurs itérations de *EM*. Ce phénomène est remarquable sur la figure lorsque 600 phrases d'adaptation sont utilisées : les performances du système adapté sont dans ce cas proportionnelles au nombre d'itérations de *EM* utilisées.

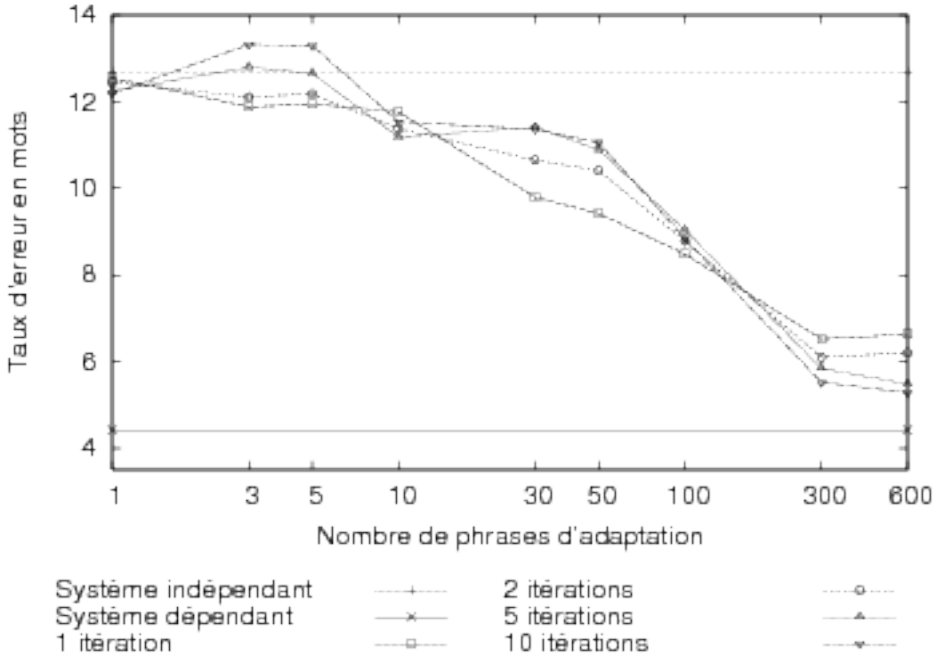


FIGURE 5.6 – Résultats *SMAP* - Adaptation par lot supervisée - Variation du nombre d'itérations β_{SMAP}

La figure 5.7 montre les performances du système adapté en utilisant *SMAP*, dans le cas d'une adaptation par lot supervisée et dans le cas d'une adaptation incrémentale non supervisée. La méthode utilisée pour récolter les statistiques suffisantes en mode incrémentale est celle proposée par Digalakis [28] et présentée dans le paragraphe 4.2.2. Dans les deux modes d'adaptation, τ fut déterminée automatiquement, en fonction du nombre de trames d'adaptation utilisées, selon la relation 7.1.

Les performances obtenues en employant une adaptation incrémentale non supervisée sont inférieures à celles obtenues avec une adaptation par lot supervisée. Nous expliquons ce phénomène par le fait que le processus d'estimation des moyennes des gaussiennes dans *SMAP* est sensible aux erreurs de transcriptions d'une part, et à la valeur de l'hyperparamètre τ d'autre part. Il semblerait toutefois que l'hyperparamètre τ a plus d'influence sur le processus d'estimation que ne l'ont les erreurs de transcriptions, si l'on compare les performances obtenues en utilisant τ défini selon l'hypothèse 1 de celles obtenues en employant τ défini selon l'hypothèse 2. Nous pensons donc que la définition d'une relation

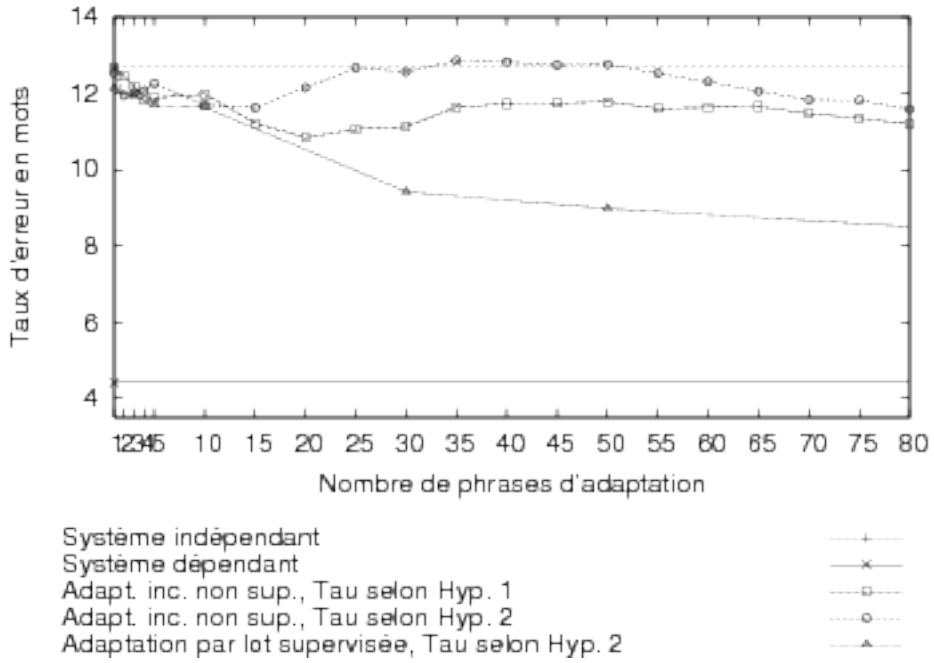


FIGURE 5.7 – Résultats *SMAP* - Adaptation incrémentale non supervisée et adaptation par lot supervisée

modélisant l'évolution de la valeur de τ en fonction de la quantité de données d'adaptation utilisée, dans le cas d'une adaptation incrémentale non supervisée, permettrait d'améliorer les performances de *SMAP* dans ce mode d'adaptation.

En mode par lot supervisé, *SMAP* permet d'améliorer le taux de reconnaissance du système indépendant du locuteur de l'ordre de 1% lorsqu'une seule phrase d'adaptation est utilisée, et de l'ordre de 25% lorsque 50 phrases ont été utilisées. En mode incrémental non supervisé, l'amélioration relative des performances chute à moins de 1% lorsqu'une seule phrase d'adaptation est employée, et à 7% dans le cas où 50 phrases ont été utilisées.

5.5 Conclusions

Nous avons abordé dans ce chapitre les principes théoriques fondamentaux de *SMAP*, ainsi que les résultats obtenus avec cette méthode en utilisant le moteur de reconnaissance *ESPERE*, dans le cas d'une adaptation par lot supervisée et dans le cas d'une adaptation incrémentale non supervisée.

SMAP permet d'améliorer le taux de reconnaissance en mots du système indépendant du locuteur, dès l'instant où une seule phrase d'adaptation est disponible. L'amélioration relative du taux de reconnaissance en mots du système adapté par rapport à celui du *SIL* n'est cependant que de l'ordre de 1% en utilisant une seule phrase d'adaptation. Elle

est proche de 7% en utilisant 10 phrases et peut atteindre 47% en utilisant 600 phrases d'adaptation.

Nous avons également montré dans ce chapitre l'influence des paramètres de *SMAP* sur les performances effectives du système adapté. Les expériences d'évaluation de *SMAP* ont révélé qu'une seule itération de *EM* est le plus souvent suffisant pour obtenir des estimations fiables des moyennes des gaussiennes du système adapté. En outre, la relation 7.1 que nous avons proposé pour déterminer automatiquement τ en fonction du nombre de trames d'adaptation disponibles s'est révélée être capable de générer des systèmes adaptés meilleurs qu'en utilisant une valeur τ fixée *a priori*, et ceci quelle que soit le nombre de phrases d'adaptation disponibles.

Chapitre 6

EigenVoices (*EV*)

Nous avons présenté et évalué dans les deux chapitres précédents les deux techniques d'adaptation des modèles acoustiques qui sont actuellement les plus utilisées, à savoir *SMLLR* et *SMAP*. Nous allons clore cette étude en vous invitant dans ce chapitre à examiner la technique des *EigenVoices*, qui fût proposée par Kuhn, Nguyen *et al.* dans [66] et [89].
A l'inverse des techniques *SMLLR* et *SMAP*, qui peuvent être utilisée aussi bien pour une adaptation au locuteur que pour une adaptation à l'environnement, la technique *EigenVoices* a été spécialement conçue pour une adaptation au locuteur. Par rapport aux techniques *SMLLR* et *SMAP*, *EV* est capable d'adapter le plus rapidement [23] les paramètres des modèles acoustiques d'un système indépendant du locuteur. *EV* permet en effet d'adapter efficacement un *SRAP* pour un nouveau locuteur après que celui-ci n'est prononcé qu'une seule phrase. Pour obtenir de telles performances, cette méthode exploite les informations contenues dans un ensemble de systèmes dépendant du locuteur. Cet ensemble caractérise la variabilité possible inter-locuteurs. A l'aide de ces systèmes, la phase d'adaptation ne consiste plus à déterminer le système adapté au nouveau locuteur à partir du système indépendant du locuteur, en estimant des paramètres associés à des transformations qui sont ensuite appliquées aux moyennes des gaussiennes du système indépendant du locuteur. Dans *EV*, l'adaptation s'apparente plutôt à une tâche de localisation du nouveau locuteur dans un espace de locuteurs de références, construit à partir de l'ensemble de systèmes dépendant du locuteur. La localisation du nouveau locuteur dans l'espace des locuteurs est obtenu après l'estimation d'un vecteur de coordonnées. Le système adapté au nouveau locuteur est obtenu en combinant linéairement les vecteurs de base de l'espace des locuteurs selon les coordonnées estimées du nouveau locuteur. Le caractère relatif de l'adaptation selon *EV* (localisation dans un espace) permet ainsi d'estimer un nombre beaucoup plus réduit de paramètres (quelques dizaines) par rapport à des techniques comme *SMLLR* ou *SMAP* (qui en estiment plusieurs milliers), si bien qu'une plus faible quantité de données d'adaptation peut être utilisée pour les estimer de manière robuste.

Nous consacrerons la première partie de ce chapitre aux concepts et aux hypothèses adop-

23. c'est-à-dire avec peu de données d'adaptation

tés dans le cadre d'une adaptation selon *EV*. Nous présenterons les différentes étapes mises en oeuvre dans le processus d'adaptation par *EV*. Nous donnerons ensuite dans la deuxième section les résultats des expériences d'évaluation de la version des *EigenVoices* que nous avons implantée. Ces expériences fûrent réalisées en utilisant le système de reconnaissance *ESPERE* et le corpus *RM*. Nous conclurons enfin sur l'efficacité d'adaptation de cette technique et sur les choix de paramétrage que nous avons retenus pour utiliser aussi efficacement que possible *EV* dans le cas d'une adaptation par lot supervisée et dans le cas d'une adaptation incrémentale non supervisée.

6.1 Approche classique

EV se distingue des méthodes classiques comme *SMLLR* ou *SMAP* par l'utilisation d'informations sur la variabilité inter-locuteurs. Grâce à elles, *EV* est capable de contraindre les modèles acoustiques adaptés à un nouveau locuteur pour qu'ils soient localisés dans une certaine portion d'un espace : l'espace des locuteurs de référence. L'adaptation selon *EV* consiste ainsi, dans un premier temps, à construire un espace des locuteurs de référence, et dans un second temps à localiser le nouveau locuteur dans cet espace.

6.1.1 Construction de l'espace des locuteurs

La construction de l'espace représentatif des locuteurs, illustrée par la figure 6.1, est réalisée à partir d'un ensemble de N systèmes dépendant du locuteur (*SDL*).

Toute cette première étape de construction de l'espace propre des locuteurs est réalisée avant le processus d'adaptation des modèles acoustiques.
Dans un premier temps, à partir de chacun de ces N *SDL* est extrait un supervecteur, de dimension D. Chaque supervecteur contient l'ensemble des paramètres d'un *SDL* qui doivent faire l'objet d'une adaptation. Dans notre cas, puisque seules les moyennes des gaussiennes des modèles acoustiques sont sujets à une adaptation, le supervecteur s_n, extrait du n-ème *SDL*, est constitué de la concaténation de tous les vecteurs de moyenne de toutes les gaussiennes de tous les modèles de ce *SDL*, c'est-à-dire :

$$s_n = \begin{bmatrix} \mu_1(n) \\ \mu_2(n) \\ \cdots \\ \mu_r(n) \\ \cdots \\ \mu_{N_G}(n) \end{bmatrix}$$

où $\mu_r(n)$ est le vecteur de moyenne de la r-ième gaussienne du n-ème *SDL* et N_G est le nombre total de gaussiennes des modèles acoustiques d'un *SDL*. La taille d'un supervecteur est donc de $D = N_G \times N_D$.
Dans la terminologie de *EV*, un supervecteur représente les caractéristiques acoustiques d'un locuteur.

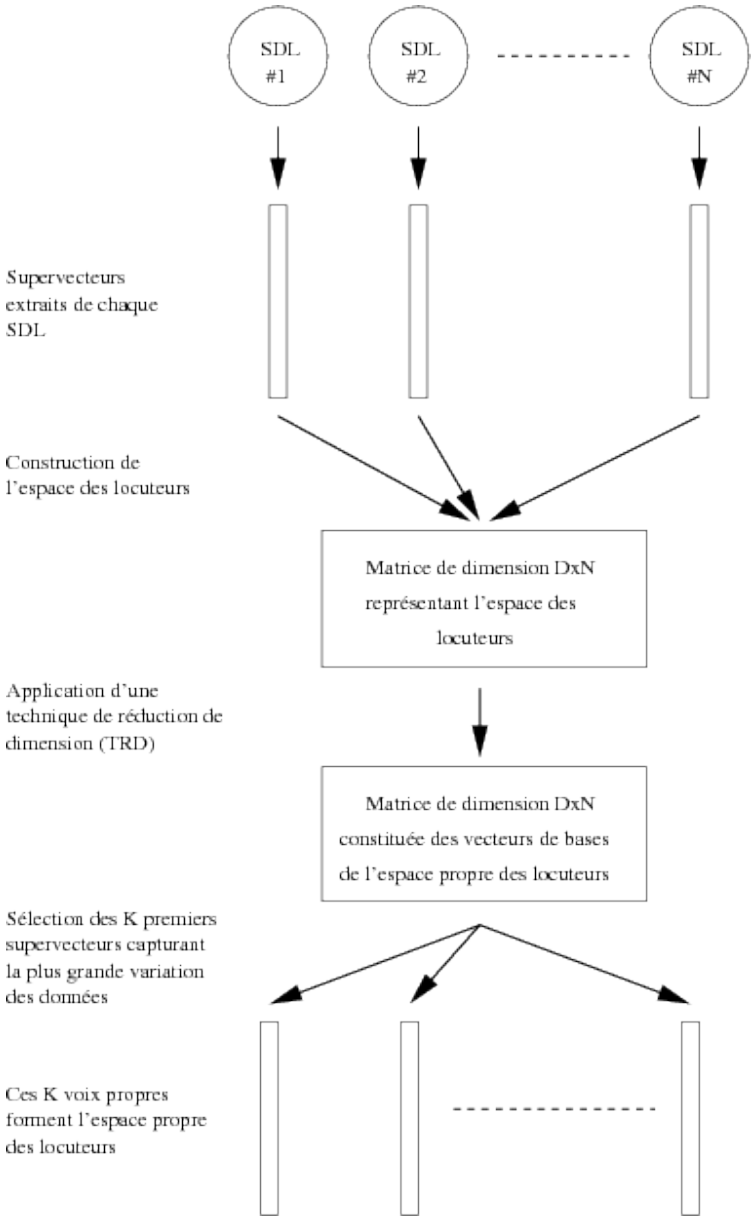


FIGURE 6.1 – Construction de l'espace des locuteurs

L'espace représentatif des locuteurs peut être représenté par une matrice M, de dimension $D \times N$, telle que :

$$M = \left[\begin{array}{cccc} s_1 & s_2 & \cdots & s_N \end{array} \right]$$

L'utilisation d'un espace des locuteurs, de dimension N, tel que représenté par la matrice M, peut conduire à estimer un nombre de paramètres N assez grand, lorsque le nombre N de locuteurs dépasse quelques centaines. La question d'employer *EV* plutôt que *SMLLR* pourrait alors se poser. Pour réduire le nombre de paramètres à estimer, une technique de réduction de dimension (*TRD*) est appliquée à la matrice M pour obtenir un espace des locuteurs qui soit réduit en dimension. Cet espace, appelé espace propre des locuteurs (ou *eigenspace*), de dimension K avec $K << N$, constitue alors une approximation de l'espace des locuteurs. De cette manière, *EV* considère que le nouveau locuteur ne peut être localisé que dans l'espace propre des locuteurs, plutôt que situé n'importe où dans l'espace des locuteurs. Les coordonnées du nouveau locuteur ne sont donc plus exprimées en N dimensions.
L'analyse en composantes principales (ACP), l'analyse en composantes principales Probabiliste [24], l'analyse en composantes indépendantes [25] ou l'analyse discriminante linéaire [26] font parties des techniques qui peuvent être utilisées comme *TRD*. Considérons que l'espace des locuteurs est caractérisé par un nuage de points, chaque point représentant un *SDL*. Les techniques de réduction de dimension permettent de rechercher à partir de ce nuage de points les axes principaux qui sont les plus discriminants, c'est-à-dire ceux qui permettent de conserver la plus grande variabilité acoustique.

La figure 6.2 illustre le concept d'utilisation d'un espace propre pour localiser un nouveau locuteur. Dans cette figure, nous faisons l'hypothèse que l'espace des locuteurs est bidimensionnel. Chaque système, indiqué par une croix, est donc représenté par un supervecteur à 2 dimensions. L'espace propre, unidimensionnel, est représenté par une droite en pointillé. La région blanche à l'intérieur de la patatoïde représente l'ensemble des systèmes possibles adaptés au nouveau locuteur, uniquement sur la base des données d'adaptation. Nous y avons représenté le système indépendant du locuteur (SIL) ainsi que le système adapté au nouveau locuteur (SA). Ce dernier est situé à la fois sur la droite représentant l'espace propre et dans la région blanche. Comme nous pouvons le constater, une estimation plus précise des modèles acoustiques du système adapté est obtenue grâce à l'utilisation de l'espace propre, et donc d'informations sur la variabilité acoustique inter-locuteurs.

6.1.2 Utilisation de l'ACP comme *TRD*

Dans le cas où l'ACP est utilisée comme *TRD*, N vecteurs propres $e(1), e(2), \cdots, e(N)$, tous de dimension D également et orthogonaux entre eux, sont obtenus à partir des N supervecteurs $s_1, s_2, \cdots, s_N$. Un vecteur propre représente une direction caractéristique

24. Probabilistic *PCA*
25. Independant Component Analysis (*ICA*)
26. Linear Discrimination Analysis (LDA)

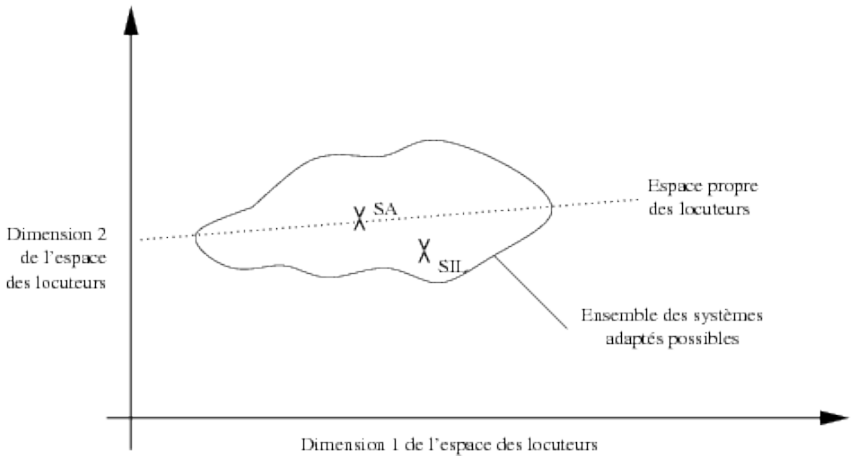


FIGURE 6.2 – Localisation du nouveau locuteur dans un espace propre

dans l'espace acoustique des locuteurs. Les K vecteurs propres qui ont la valeur propre la plus grande sont ceux qui retiennent la plus grande variation des données : ce sont les voix propres (ou *eigenvoices*). L'espace propre des locuteurs est alors engendré par ces K voix propres.
A ces K voix propres est joint un supervecteur $e(0)$ qui représente l'origine, dans l'espace des locuteurs, de l'espace propre engendré par les K voix propres. L'espace propre ainsi engendré représente alors une approximation *linéaire* de l'espace des locuteurs. Le supervecteur origine $e(0)$ peut être choisi comme étant soit le supervecteur moyen des N supervecteurs $s_1, s_2, \cdots, s_N$, soit le supervecteur extrait du système indépendant du locuteur.

6.1.3 Localisation du nouveau locuteur

EV suppose que le supervecteur $\hat{\mu}$ du système adapté est une combinaison linéaire des $K+1$ supervecteurs $e(0), e(1), e(2), \cdots, e(K)$ (figure 6.3).

Le supervecteur du système adapté est calculé selon l'équation suivante :

$$\hat{\mu} = \sum_{k=0}^{K} w_k \, e(k)$$

Les autres paramètres du système adapté qui n'ont pas subi d'adaptation sont extraits du *SIL*.
Les $K+1$ coordonnées [27] $w_i \ \ \forall i = 0, 1, 2, \cdots, K$ sont estimées au maximum de vraisemblance des données d'adaptation, à l'aide de la procédure *Maximum Likelihood Eigen-Decomposition (MLED)* [89]. Elle consiste à maximiser la vraisemblance des données

27. ou poids, dans le sens où chaque w_i pondère le vecteur $e(i)$ auquel il est associé dans la combinaison linéaire. Dans la littérature consacrée à *EigenVoices*, le terme de poids est plus communément employé.

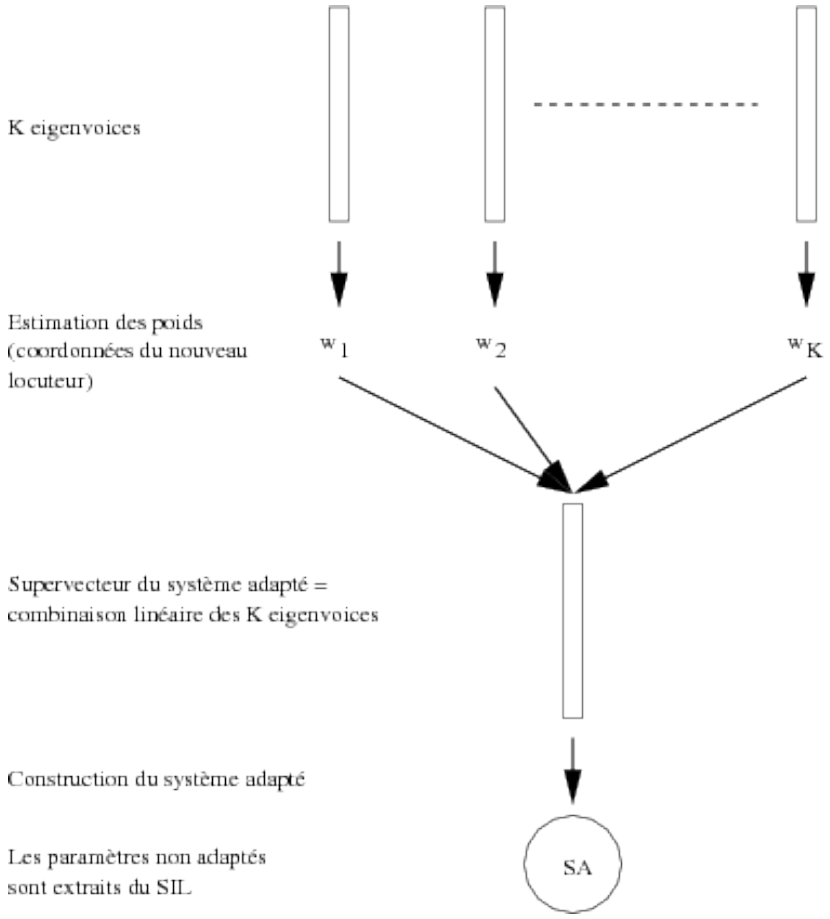


FIGURE 6.3 – Localisation du nouveau locuteur et construction du système adapté

d'adaptation $p(O/\Theta)$, où O est l'ensemble des observations issues des données d'adaptation et Θ est l'ensemble des paramètres des modèles acoustiques du système indépendant du locuteur.
Comme toute procédure d'estimation basée sur le critère du maximum de vraisemblance, l'algorithme *EM* est utilisé pour maximiser $p(O/\Theta)$. Il a été montré que maximiser la fonction auxiliaire $Q(\Theta, \hat{\Theta})$ de manière itérative revient à maximiser la vraisemblance $p(O/\Theta)$.
Soit :

$$Q(\Theta, \hat{\Theta}) = -\frac{1}{2} \sum_{r=1}^{N_G} \sum_{t=1}^{T} \left[\gamma_r(t)\ N_D\ log\ (2\pi)\ +\ log\ |\sigma_r|\ +\ h(o_t, r)\right] \tag{6.1}$$

où

$$h(o_t, r) = (o_t\ -\ \hat{\mu}_r)'\ \sigma_r^{-1}\ (o_t\ -\ \hat{\mu}_r)$$

$\hat{\Theta}$ sont les paramètres des modèles acoustiques du système adapté, $\hat{\mu}_r$ est le vecteur de moyenne de la r-ième gaussienne du système adapté, tel que :

$$\hat{\mu}_r = \sum_{k=0}^{K} w_k\ e_r(k)$$

où $e_r(k)$ représente le vecteur de N_D paramètres situé à la position de la r-ième gaussienne dans le supervecteur $e(k)$:

$$e(k) = \begin{bmatrix} e_1(k) \\ e_2(k) \\ \vdots \\ e_r(k) \\ \vdots \\ e_{N_G}(k) \end{bmatrix}$$

Maximiser Q revient alors à la dériver et à mettre la dérivée à zéro pour chaque poids w_i :

$$\frac{\delta Q}{\delta w_i} = 0 = \sum_{r=1}^{N_G} \sum_{t=1}^{T} \left[\frac{\delta}{\delta w_i}\ \gamma_r(t)\ h(o_t, r)\right]\ \text{pour } i = 0, 1, 2, \cdots, K$$

car $\frac{\delta w_i}{\delta w_j} = 0$ pour $i \neq j$ puisque les voix propres sont orthogonales entre elles.
Nous obtenons alors le système suivant de $K+1$ équations linéaires à $K+1$ inconnues :

$$\sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\ e_r(i)' \sigma_r^{-1} o_t\ \ =\ \ \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\ \sum_{k=0}^{K} w_k\ e_r(k)'\ \sigma_r^{-1}\ e_r(i) \tag{6.2}$$

pour $i = 0, 1, 2, \cdots, K$.
Ce système peut se réécrire sous forme matricielle comme :

$$v = M\ w \tag{6.3}$$

où $v = \begin{pmatrix} \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, e_r(0)'\, \sigma_r^{-1}\, o_t \\ \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, e_r(1)'\, \sigma_r^{-1}\, o_t \\ \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, e_r(2)'\, \sigma_r^{-1}\, o_t \\ \vdots \\ \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, e_r(K)'\, \sigma_r^{-1}\, o_t \end{pmatrix}$, $M = (m_{(i,j)})$ et $w = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_K \end{pmatrix}$ avec

$$m_{(i,j)} = \sum_{r=1}^{N_G} \sum_{t=1}^{T} \gamma_r(t)\, e_r(j)'\, \sigma_r^{-1}\, e_r(i)$$

Le vecteur de poids w peut alors être obtenu selon la formule :

$$w \quad = \quad M^{-1}v \tag{6.4}$$

Une seule inversion de matrice est donc nécessaire pour obtenir les coordonnées du nouveau locuteur.

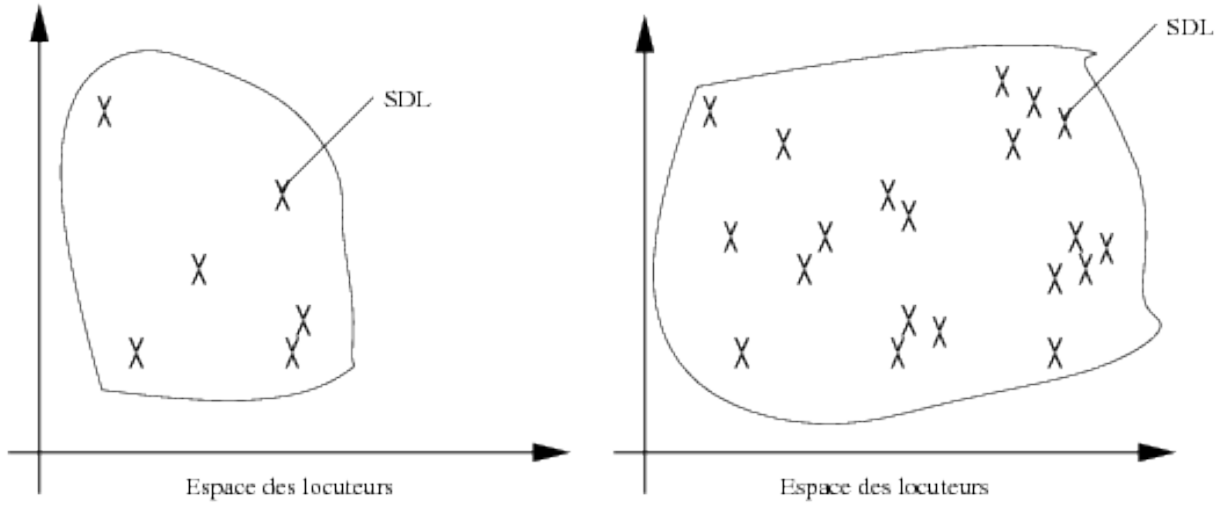
6.2 Qualité de l'espace propre des locuteurs

L'approche des *EigenVoices* émet l'hypothèse forte que le nouveau locuteur est situé dans une certaine portion de l'espace des locuteurs de référence : cette portion constitue l'espace propre des locuteurs. Il apparait ainsi que l'efficacité de *EV* est étroitement liée à la qualité de cet espace propre.
L'espace propre est une approximation de l'espace des locuteurs. Il est construit à l'intérieur de ce dernier. Un mauvais espace des locuteurs implique ainsi que l'estimation de la position d'un nouveau locuteur peut être incorrecte.
Pour disposer d'un espace des locuteurs aussi fiable que possible, il doit être construit à partir d'un assez grand nombre de *SDL*, et ceux-ci doivent avoir été suffisamment entrainés. En effet, plus on dispose de *SDL*, qui plus est appris à l'aide de données provenant de locuteurs qui possèdent des caractéristiques différentes (en genre, dialecte, vitesse d'élocution, etc.), plus cet espace pourra capturer aussi largement que possible la variabilité acoustique inter-locuteurs. La figure 6.4 illustre clairement ce principe.

En outre, si chaque *SDL* a été suffisamment entrainé, cette variabilité pourra être exprimée de manière encore plus précise.
Enfin, c'est le choix de la technique de réduction de dimension qui conditionne la topologie et donc la qualité de l'espace propre engendré.

6.3 Apprentissage des systèmes dépendant du locuteur

Comme nous l'avons évoqué précédemment, l'efficacité de la technique des *EigenVoices* dépend en grande partie de la phase d'apprentissage des systèmes dépendant du locuteur.



(a) Prise en compte d'une faible variabilité inter–locuteurs (région blanche) due à un nombre restreint de SDL

(b) Prise en compte d'une grande variabilité inter–locuteurs (région blanche) due à un nombre important de SDL

FIGURE 6.4 – Qualité d'un espace des locuteurs

Plusieurs approches sont possibles pour apprendre un ensemble de systèmes dépendant du locuteur, selon que l'on dispose de plus ou moins de données d'apprentissage par locuteur. La plus simple des méthodes, lorsque l'on dispose d'une quantité très importante de données d'apprentissage pour chaque locuteur de référence (ce qui est rarement le cas), consiste à poursuivre l'apprentissage du *SIL* en utilisant les phrases prononcées par un locuteur de référence r, pour obtenir le r-ème *SDL*. Le plus souvent, cette approche génère des *SDL* médiocres, en raison des estimations imprécises fournies par une quantité faible de données d'apprentissage disponibles. Une méthode alternative consiste à adapter le *SIL* avec une technique d'adaptation conventionnelle (*SMLLR* ou *SMAP*) en utilisant les phrases prononcées par un locuteur de référence r afin de générer le r-ème *SDL*.

6.4 Evaluations expérimentales

Nous présentons dans cette section les résultats obtenus avec *EV* sous les conditions expérimentales exposées dans le chapitre 3.

La qualité du système adapté généré en utilisant *EV* dépend de cinq paramètres, qui sont :

- la méthode utilisée pour apprendre chaque *SDL*,
- la technique de réduction de dimension choisie pour générer les vecteurs propres,
- l'origine de l'espace propre des locuteurs dans l'espace des locuteurs,
- le nombre d'itérations β_{EV} définissant le nombre de fois où l'algorithme *EM* est utilisé,
- le nombre de poids K à estimer.

Etant donné que nous disposons de peu de données d'adaptation par locuteur (40 phrases), un réapprentissage à partir du *SIL* de tous les paramètres des modèles acoustiques ne peut raisonnablement être réalisé. Les paramètres des modèles acoustiques des *SDL* seraient effectivement susceptibles d'être mal estimés. Chaque *SDL* fût généré à partir du *SIL* à l'aide de la technique d'adaptation *SMAP* en utilisant le corpus de 40 phrases disponible pour chaque locuteur. Les vecteurs propres fûrent obtenus en appliquant la technique *PCA* aux supervecteurs extraits des *SDL* générés précédemment.

L'influence du choix de l'origine e_0 de l'espace propre dans l'espace des locuteurs sur les performances du système adapté est montré dans le tableau 6.1. Nous pouvons constater que l'utilisation du supervecteur du *SIL* comme origine de l'espace propre donne de meilleurs résultats que l'utilisation du supervecteur obtenu en moyennant les supervecteurs des *SDL*.

Nombre de poids	Taux de reconnaissance en Mots
10	87.24%
20	87.54%
30	87.61%
40	87.72%
50	**87.86%**
60	87.71%
72	87.83%

Origine = Supervecteur du *SIL*

Nombre de poids	Taux de reconnaissance en Mots
10	-
20	60.73%
30	86.91%
40	87.60%
50	87.45%
60	87.64%
72	87.45%

Origine = Supervecteur moyenne des supervecteurs des *SDL*

TABLE 6.1 – Résultats *EV* - Adaptation avec 1 phrase - Variation du nombre de poids

Des expériences ont révélé que le nombre d'itérations n'a qu'une influence marginale sur les performances du système adapté avec *EV*.
Ceci peut s'expliquer par le fait que *EV* ne peut pas prendre en compte la totalité des informations contenues dans les données d'adaptation. Les variables d'adaptation à estimer (les $K+1$ poids de la combinaison linéaire), en nombre très réduit, peuvent être estimées précisément avec une quantité faible de données d'adaptation. A cet égard, plusieurs itérations de *EM* sont donc inutiles pour affiner les estimations obtenues après la première itération.

La figure 6.5 présente les résultats d'une adaptation par lot supervisée et d'une adaptation incrémentale non supervisée en utilisant *EV*. Nous avons utilisé le supervecteur du *SIL* comme origine de l'espace propre ainsi qu'une seule itération de *EM*. La méthode de récolte des statistiques suffisantes proposée par Digalakis [28] et présentée au paragraphe 4.2.2 fût utilisée pour l'adaptation incrémentale.

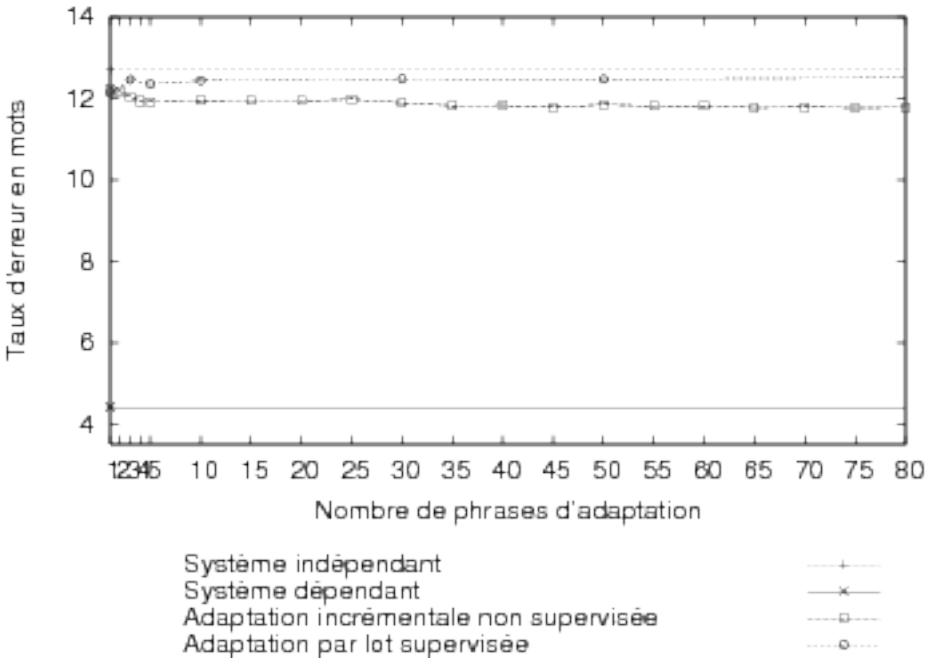


FIGURE 6.5 – Résultats *EV* - Adaptation par lot supervisée et adaptation incrémentale non supervisée

Cette figure met en évidence l'efficacité de *EV* lorsqu'il s'agit d'adapter rapidement un système indépendant du locuteur. L'emploi d'une seule phrase d'adaptation permet en effet d'améliorer les performances du système indépendant du locuteur de l'ordre de 4%. Il est intéressant de remarquer que l'emploi de la procédure de récolte des statistiques de Digalakis permet, dans le cadre de *EV* et pour une adaptation incrémentale, d'atteindre de meilleures performances par rapport à celles obtenues en mode par lot. Toutefois, dans les deux modes d'adaptation, les performances du système adapté généré à l'aide de *EV* stagnent rapidement lorsque le nombre de phrases d'adaptation disponibles augmente. Ce phénomène est essentiellement dû au fait que le nombre de variables d'adaptation dans *EV* reste constant quelle que soit la quantité de données d'adaptation disponibles.

6.5 Conclusions

Ce chapitre nous a permis de présenter les concepts théoriques fondamentaux de *EV*, ainsi que les résultats obtenus avec cette méthode en utilisant le moteur de reconnaissance *ESPERE*, dans le cas d'une adaptation par lot supervisée et dans le cas d'une adaptation incrémentale non supervisée.

EV permet d'améliorer de manière significative les performances d'un système indépendant du locuteur en utilisant très peu de données d'adaptation (moins de 3 secondes

de parole). Toutefois, cette technique souffre d'une saturation rapide des performances lorsque la quantité de données d'adaptation augmente, aussi bien en mode d'adaptation par lot qu'en mode d'adaptation incrémental. L'amélioration relative du taux de reconnaissance en mots du système adapté par rapport à celui du *SIL* reste en effet de l'ordre de 4% pour une adaptation par lot, quelque soit le nombre de phrases d'adaptation disponibles. Pour une adaptation incrémentale, l'amélioration des performances peut atteindre 7% en utilisant 5 phrases d'adaptation.

Nous avons également montré dans ce chapitre l'influence des paramètres de *EV* sur les performances effectives du système adapté. L'évaluation expérimentale de *EV* a révélé que de meilleures performances sont atteintes lorsque le supervecteur du *SIL* est utilisé comme origine de l'espace propre dans l'espace des locuteurs, plutôt que d'employer le supervecteur moyenne des supervecteurs des *SDL*. Le nombre de poids K à estimer reste difficile à déterminer. Cependant, si K est suffisamment grand (dans notre cas supérieur à 20), ce paramètre n'affecte alors que très marginalement les performances du système adapté. Enfin, une seule itération de *EM* est suffisante pour estimer les variables d'adaptation, c'est-à-dire les coordonnées du nouveau locuteur. Une grande précision dans l'estimation des paramètres est en effet inutile dans le cas où la quantité de données d'adaptation est importante puisque le nombre de variables d'adaptation n'évolue pas avec la quantité disponible de données d'adaptation.

Chapitre 7

Bilan comparatif des techniques *SMLLR*, *SMAP* et *EV*

Nous avons étudié dans les chapitres 4, 5 et 6 les techniques d'adaptation au locuteur *SMLLR*, *SMAP* et *EV*, respectivement. Elles constituent actuellement les techniques les plus communément employées pour adapter les moyennes des gaussiennes d'un système indépendant du locuteur. Nous nous sommes intéressés plus particulièrement dans ces chapitres à déterminer, pour chaque technique, le paramétrage qui lui permet de fournir les meilleures performances.

Afin de mieux cerner l'efficacité relative de ces techniques, nous dressons dans ce chapitre un comparatif des techniques *SMLLR*, *SMAP* et *EV*. En l'occurence, nous traiterons de la différence de ces trois techniques en terme d'amélioration du taux d'erreur en mots, en terme de complexité et en terme de place mémoire.

7.1 Comparatif en terme d'amélioration du taux d'erreur en mots

Tous les résultats qui suivent sont obtenus en utilisant les paramétrages donnés dans le tableau 7.1. Le meilleur paramétrage de chaque technique fût déterminé à l'issue d'une série d'expériences (voir paragraphes 4.4, 5.4 et 6.4).

7.1.1 Adaptation par lot supervisée

La figure 7.1 montre l'évolution du taux d'erreur en mots des systèmes obtenus après une adaptation par lot supervisée d'un système indépendant du locuteur en utilisant *SMLLR*, *SMAP* ou *EV*, en fonction du nombre de phrases d'adaptation disponibles.
Le tableau 7.2 indique l'évolution de l'amélioration des performances par rapport au système indépendant du locuteur, des systèmes adaptés en utilisant *SMLLR*, *SMAP* ou *EV* en mode par lot supervisé.

Lorsqu'une seule phrase d'adaptation est disponible, la technique *EV* donne les meilleurs

Techniques	Nombre d'itérations de *EM*	Autres paramètres
SMLLR	1	Matrices pleines, $\alpha_{SMLLR} = 800$
SMAP	1	τ déterminé selon l'équation $\tau = \begin{cases} 1000 & \text{si } T < 1000 \\ \left(\frac{10000}{T}\right)^3 & \text{si } T \geq 1000 \end{cases}$
EV	1	Technique de Réduction de Dimension=PCA Origine de l'espace = *SIL*, $K = 50$

TABLE 7.1 – Paramétrage des techniques *SMLLR*, *SMAP* et *EV*

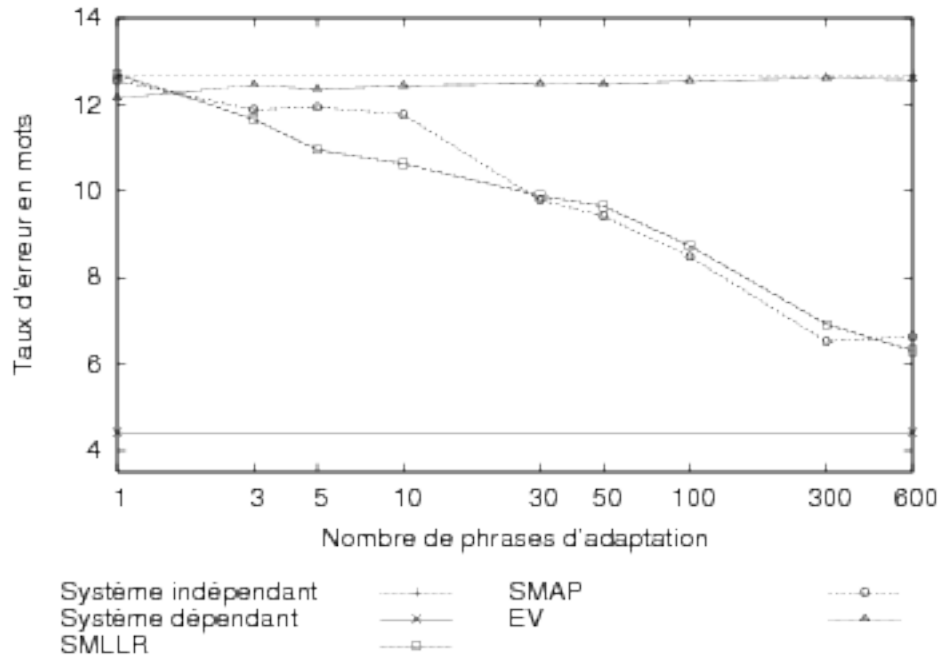


FIGURE 7.1 – Evolution du taux d'erreur en mots des systèmes adaptés en utilisant *SMLLR*, *SMAP* ou *EV* en mode d'adaptation par lot supervisée

	1	5	10	30	50	100	300	600
SMLLR	0 %	**13.8** %	**16.3** %	22.1 %	24.0 %	31.3 %	45.5 %	**50.3** %
SMAP	1.2 %	6.0 %	7.3 %	**22.9** %	**25.8** %	**33.2** %	**48.7** %	47.8 %
EV	**4.4** %	2.8 %	2.0 %	1.7 %	1.8 %	1.3 %	0.7 %	1.0 %

TABLE 7.2 – Amélioration du taux d'erreur en mots par rapport au *SIL* des systèmes adaptés en utilisant *SMLLR*, *SMAP* ou *EV* en mode d'adaptation par lot supervisée

résultats. En l'absence d'un nombre suffisant de données d'adaptation, les connaissances *a priori* sur la variabilité due au locuteur permettent à *EV* d'adapter de manière robuste et efficace les moyennes observées, ainsi que celles non observées. *SMAP* utilise également des connaissances *a priori*. Mais en constatant l'amélioration plus faible des performances obtenues par *SMAP* par rapport à celle de *EV*, les connaissances employées par *SMAP* semblent moins riches que celles utilisées par *EV*. A l'inverse, comme aucune connaissance *a priori* n'est introduite dans *SMLLR*, une certaine quantité de données d'adaptation est nécessaire pour estimer de manière robuste les paramètres des transformations linéaires. Dans le cas où une seule phrase d'adaptation est disponible, la quantité de données d'adaptation n'est pas suffisante pour les estimer de manière robuste : par précaution, les moyennes ne sont donc pas adaptées, si bien que le système généré par *SMLLR* fournit les mêmes performances que le *SIL*.

En utilisant plus de trois phrases d'adaptation, ce sont les techniques *SMLLR* et *SMAP* qui génèrent les systèmes de *RAP* les plus performants. Le nombre plus élevé de variables d'adaptation permet effectivement à ces deux techniques de prendre en compte une quantité plus importante des informations contenues dans les données d'adaptation disponibles, par rapport à *EV*. Lorsqu'entre trois et dix phrases d'adaptation sont disponibles, *SMLLR* est capable d'améliorer les performances du *SIL* d'au plus 16%. Avec plus de dix phrases d'adaptation, *SMAP* peut améliorer les performances du *SIL* de l'ordre d'au plus 48%. Les écarts de performances entre ces deux techniques sont dûs, à notre avis, aux paramétrages non optimaux de *SMLLR* et de *SMAP*.

7.1.2 Adaptation incrémentale non supervisée

La figure 7.2 montre l'évolution du taux d'erreur en mots des systèmes adaptés de manière incrémentale et non supervisée à partir d'un système indépendant du locuteur en utilisant *SMLLR*, *SMAP* ou *EV*.
L'évolution de l'amélioration des performances par rapport au système indépendant du locuteur des systèmes adaptés en utilisant *SMLLR*, *SMAP* ou *EV*, en mode incrémental non supervisé, est indiquée dans le tableau 7.3.

	1	3	5	10	20	30	50	80
SMLLR	0 %	**7.1** %	**12.7** %	**16.2** %	**19.0** %	**21.0** %	**22.6** %	**26.1** %
SMAP	0.1 %	4.0 %	6.5 %	5.7 %	14.6 %	12.4 %	7.2 %	11.9 %
EV	**3.5** %	5.2 %	6.1 %	5.9 %	6.0 %	6.4 %	6.7 %	7.2 %

TABLE 7.3 – Amélioration du taux d'erreur en mots par rapport au *SIL* des systèmes adaptés en utilisant *SMLLR*, *SMAP* ou *EV* en mode d'adaptation incrémentale non supervisée

Lorsqu'une seule phrase d'adaptation est disponible, *EV* est, dans ce mode d'adaptation également, la technique qui permet d'améliorer le plus significativement les performances du système indépendant du locuteur. Les connaissances *a priori* employées par *EV* lui

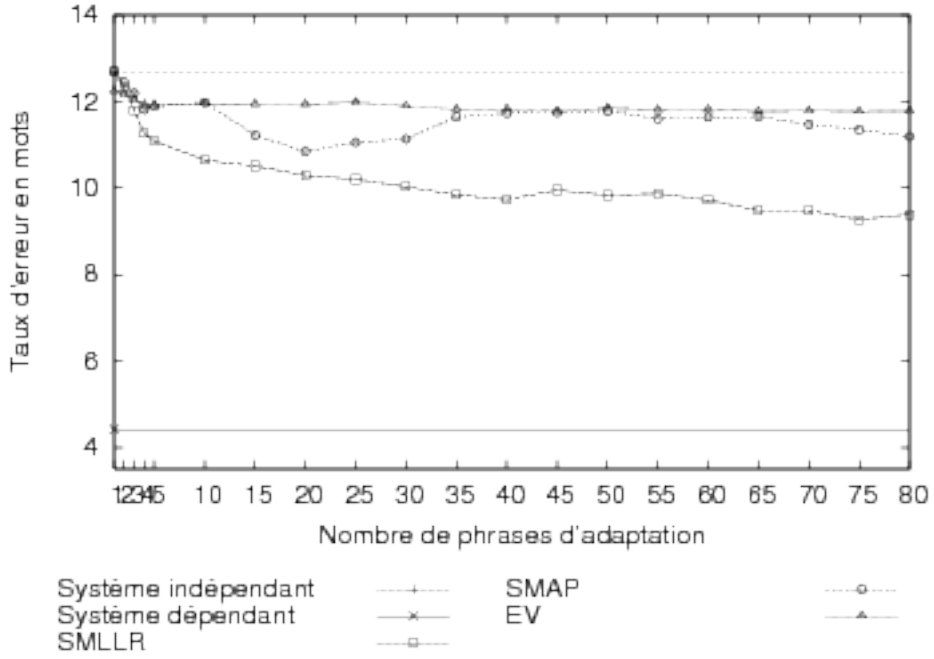


FIGURE 7.2 – Evolution du taux d'erreur en mots des systèmes adaptés avec *SMLLR*, *SMAP* ou *EV* en mode d'adaptation incrémentale non supervisée

permettent d'estimer un nombre plus restreint de variables d'adaptation, par rapport à *SMLLR* et à *SMAP*. Dans le cas où la transcription de la phrase d'adaptation n'est pas connue et lorsque peu de données d'adaptation sont disponibles, ces variables d'adaptation peuvent donc être estimées d'une manière d'autant plus fiable dans le cas de *EV*.

Lorsque plus de trois phrases d'adaptation sont utilisées, la meilleure amélioration des performances est obtenue en utilisant *SMLLR*. Sa flexibilité, ainsi que sa moins grande sensibilité au paramétrage (par rapport à *SMAP*, voir paragraphe 5.4) permet à *SMLLR* d'améliorer significativement les performances du *SIL* quelque soit le nombre de phrases d'adaptation utilisées.

7.2 Comparatif en terme de complexité

Nous donnons dans cette section une étude de complexité pour les techniques *SMLLR*, *SMAP* et *EV*, lorsqu'elles sont employées pour adapter uniquement les moyennes d'un système indépendant du locuteur.
Nous considérons dans cette étude que :

- toutes les matrices de variances-covariances sont diagonales,
- une seule itération de *EM* est utilisée,
- seule l'estimation des variables d'adaptation et l'application de ces variables aux vecteurs de moyenne des gaussiennes sont prises en compte,

- N_D représente la taille d'un vecteur d'observation et
- N_G est le nombre total de gaussiennes des modèles acoustiques du *SIL*.

7.2.1 Complexité de *SMLLR*

Dans le cas de *SMLLR*, l'estimation des paramètres d'une transformation implique l'inversion d'une matrice de dimension $(N_D+1)\times(N_D+1)$ pour chacune des N_D dimensions de la matrice de transformation (équation 4.14). Une inversion de matrice peut être réalisée en $\mathcal{O}({N_D}^3)$ opérations. Le coût total pour estimer les paramètres d'une transformation est donc de $\mathcal{O}({N_D}^4)$ opérations. L'estimation de M matrices de transformation nécessite alors $\mathcal{O}(M \times {N_D}^4)$ opérations. Une fois les transformations estimées, $\mathcal{O}(N_G \times {N_D}^2)$ opérations sont nécessaires pour adapter les vecteurs de moyenne des gaussiennes.
Le nombre d'opérations N^{SMLLR} nécessaires pour adapter les moyennes des gaussiennes d'un *SIL* en utilisant *SMLLR* est donc :

$$N^{SMLLR} = \mathcal{O}(M \times {N_D}^4 + N_G \times {N_D}^2) \tag{7.1}$$

7.2.2 Complexité de *SMAP*

Dans le cas de *SMAP*, un vecteur de biais est estimé à chaque nœud de l'arbre (équation 5.12). L'estimation de ce vecteur nécessite $\mathcal{O}(N_D)$ opérations. Si l'arbre comporte F nœuds, $\mathcal{O}(F\times N_D)$ opérations sont donc nécessaires pour estimer tous les vecteurs de biais. Une fois les N_G vecteurs de biais estimés au niveau des feuilles de l'arbre, $\mathcal{O}(N_G \times N_D)$ opérations sont nécessaires pour adapter les vecteurs de moyenne des gaussiennes du *SIL*.

Le nombre d'opérations N^{SMAP} nécessaires pour adapter les moyennes des gaussiennes d'un *SIL* en utilisant *SMAP* est ainsi :

$$N^{SMAP} = \mathcal{O}((F + N_G) \times N_D) \tag{7.2}$$

7.2.3 Complexité de *EV*

Dans le cas de *EV*, l'estimation des K variables d'adaptation nécessite l'inversion d'une matrice de dimension $(K + 1) \times (K + 1)$ (équation 6.4). Cette inversion requiert $\mathcal{O}(K^3)$ opérations. Une fois les variables d'adaptation estimées, l'adaptation des N_G vecteurs de moyenne nécessite $\mathcal{O}(N_G \times N_D \times K)$.
Le nombre total d'opérations N^{EV} nécessaires pour adapter les moyennes des gaussiennes d'un *SIL* en utilisant *EV* est donc :

$$N^{EV} = \mathcal{O}(K^3 + N_G \times N_D \times K) \tag{7.3}$$

7.2.4 Comparaison des temps de calcul de *SMLLR*, *SMAP* et *EV*

Afin de comparer les temps de calcul de ces trois techniques, considérons que les valeurs de N_D, N_G, M, F et K sont celles utilisées dans les expériences d'évaluation des chapitres 4, 5 et 6, et telles que :

- $N_D = 35$
- $N_G = 4352$
- $F = 32$
- $K = 50$

La valeur de M dépend de la quantité de données d'adaptation disponibles. Le tableau 7.4 précise le nombre de transformations effectivement estimées par *SMLLR* en fonction du nombre de phrases d'adaptation disponibles.

Nombre de phrases d'adaptation	1	3	5	10	30	50	100	300	600
Nombre de transformations estimées	0	1	2	5	15	28	68	167	266

TABLE 7.4 – Nombre de transformations estimées par *SMLLR* en fonction du nombre de phrases d'adaptation disponibles

Le nombre théorique d'opérations réalisées par *SMLLR*, *SMAP* et *EV*, en supposant les valeurs précédentes, est indiqué dans le le tableau 7.5 :

Nombre de phrases d'adaptation	*SMLLR*	*SMAP*	*EV*
1	5331200	153440	7741000
3	6831825	153440	7741000
5	8332450	153440	7741000
10	12834325	153440	7741000
30	27840575	153440	7741000
50	47348700	153440	7741000
100	107373700	153440	7741000
300	255935575	153440	7741000
600	404497450	153440	7741000

TABLE 7.5 – Nombre théorique d'opérations réalisées par *SMLLR*, *SMAP* et *EV* en fonction du nombre de phrases d'adaptation disponibles

Aucun des paramètres de *SMAP* et de *EV* ne varie en fonction de la quantité de données d'adaptation : leur temps de calcul est donc constant quelque soit le nombre de phrases d'adaptation utilisées. Ce n'est pas le cas pour *SMLLR*, dont le nombre d'opérations réalisées augmente considérablement en fonction de la quantité de données d'adaptation disponible.

A titre indicatif, le tableau 7.6 présente les temps d'exécution en secondes, sur un PC cadencé à 1,4 GHz et disposant de 512 Mo de RAM, des techniques *SMLLR*, *SMAP* et *EV*, en fonction du nombre de phrases d'adaptation disponibles. *SMLLR*, *SMAP* et *EV* furent employées pour une adaptation par lot supervisée.
Ces temps d'exécution concernent une seule itération de *EM* et ne prennent en compte que l'estimation des variables d'adaptation ainsi que l'application de ces variables aux moyennes des gaussiennes pour obtenir les moyennes adaptées.

	1	3	5	10	30	50	100	300	600
SMLLR	0	9	9	10	11	12	15	17	18
SMAP	< 1	< 1	< 1	< 1	< 1	< 1	< 1	< 1	< 1
EV	13	13	13	13	13	13	13	13	13

TABLE 7.6 – Temps d'exécution en secondes de *SMLLR*, *SMAP* et *EV* en fonction du nombre de phrases disponibles

Sachant qu'une phrase provenant du corpus *RM* dure approximativement trois secondes, nous pouvons constater que toutes les techniques sont temps-réel, à l'exception de *EV* lorsque moins de cinq phrases d'adaptation sont disponibles.

7.3 Comparatif en terme de place mémoire

Nous traitons dans cette section de la place mémoire nécessaire pour implanter *SMLLR*, *SMAP* ou *EV*, dans l'hypothèse que seules les moyennes des gaussiennes sont adaptées. Uniquement le stockage relatif aux moyennes sera pris en compte, que ce soient les moyennes des gaussiennes du système indépendant du locuteur ou des systèmes dépendant du locuteur utilisés dans le cas de *EV*. Nous considérons en effet que le volume de stockage des autres paramètres est négligeable par rapport à celui des moyennes.

Dans le cas de *SMLLR* et de *SMAP*, les $N_G \times N_D$ moyennes du système indépendant du locuteur doivent être stockées.

Dans le cas de *EV*, chaque voix propre employée dans l'adaptation des moyennes est composé de $N_G \times N_D$ paramètres. L'utilisation de K voix propres nécessite donc le stockage de $K \times N_G \times N_D$ paramètres.

En utilisant les valeurs de N_G, N_D et de K indiquées dans la section précédente, l'emploi

de *SMLLR* ou de *SMAP* nécessite alors le stockage de 152320 paramètres, tandis que celui de *EV* requiert le stockage de 7616000 paramètres.

EV constitue donc la technique la plus gourmande en place mémoire, alors que la place mémoire requise pour *SMLLR* est la même que celle nécessaire à *SMAP*.

7.4 Conclusions

Les expériences d'évaluation des techniques *SMLLR*, *SMAP* et *EV* qui ont été présentées respectivement dans les chapitres 4, 5 et 6, ainsi que le comparatif de ces techniques qui a été abordé dans ce chapitre, nous permettent de formuler les conclusions suivantes :

- Lorsque le nombre de phrases d'adaptation disponibles est réduit (moins d'une centaine de phrases), une seule itération de *EM* peut être utilisée par *SMLLR*, *SMAP* et *EV*.

- Dans le cas de *SMLLR*, une matrice pleine avec biais peut généralement être employée efficacement pour représenter une régression linéaire.

- Dans le cas de *SMAP*, des performances acceptables peuvent être obtenues, quelque soit le nombre de phrases d'adaptation utilisées, en déterminant automatiquement l'hyperparamètre τ en fonction du nombre de trames disponibles.

- Dans le cas de *EV*, le système indépendant du locuteur peut être utilisé en tant qu'origine de l'espace propre des locuteurs dans l'espace des locuteurs.

- Pour une adaptation rapide (moins de 5 secondes de parole), *EV* est la technique la plus performante.

- Pour une adaptation continue, à la fois en mode d'adaptation par lot et en mode incrémental, *SMLLR* est la technique la plus efficace, dès l'instant où au moins trois phrases d'adaptation ont été utilisées.

Troisième partie

Contributions à l'adaptation au locuteur des modèles acoustiques

Chapitre 8

Méthodes originales pour l'adaptation continue

Nous avons présenté dans le chapitre 6 la technique *EigenVoices*, ainsi que les résultats expérimentaux obtenus avec cette méthode en utilisant le moteur de reconnaissance *ESPERE* et le corpus de parole *RM*. Les résultats de ces expériences ont mis en lumière le principal inconvénient d'utiliser *EV* dans un système de reconnaissance automatique de la parole : aussi bien pour une adaptation par lot que pour une adaptation incrémentale, *EV* souffre d'une saturation précoce des performances lorsque la quantité de données d'adaptation devient importante. Ce phénomène s'explique par le fait que *EV* utilise très peu de variables d'adaptation et que le nombre de ces variables ne varie pas avec la quantité disponible de données d'adaptation. Lorsque la quantité de données d'adaptation est assez importante, les informations qui y sont contenues ne peuvent donc pas être totalement exploitée par cet ensemble restreint de variables d'adaptation.
A l'inverse, la technique *SMLLR* (présentée dans le chapitre 4 et comparée aux autres techniques dans le chapitre 7) s'est révélée particulièrement efficace lorsque le nombre de phrases d'adaptation utilisée est important. Elle n'améliore cependant pas les performances d'un *SIL* lorsque la quantité de données d'adaptation est trop faible.

Dans ce contexte, nous avons décidé d'orienter nos travaux sur la conception d'une technique d'adaptation capable de délivrer des performances acceptables quelle que soit le nombre de phrase d'adaptation utilisées. Ce chapitre est destiné à présenter l'ensemble de ces travaux.
La première section de ce chapitre sera consacrée à la technique *SEV*, qui permet d'améliorer l'efficacité de la technique *EV* lorsque le nombre de phrases d'adaptation devient important. Elle constitue la première étape dans l'élaboration d'une technique fiable en adaptation continue.
La deuxième section de ce chapitre portera sur les techniques que nous avons proposées afin d'adapter efficacement et de manière continue les modèles acoustiques d'un système indépendant du locuteur.

8.1 *Structural EigenVoices* (*SEV*)

Nous présentons dans cette section la version structurelle de *EV*, Structural *EigenVoices* (ou *SEV*), que nous avons développé et évalué au cours de cette thèse [71]. Cette approche est destinée à pallier le problème de saturation rapide des performances rencontrée par la version classique de *EV*.
A l'instar de la version classique de *EV*, *SEV* utilise un espace propre des locuteurs, dans lequel est localisé le nouveau locuteur. Cette localisation permet de générer le système adapté au nouveau locuteur. Mais à l'inverse de *EV*, la localisation du locuteur dans *SEV* peut être de plus en plus précise à mesure que la quantité de données d'adaptation s'accroît. Pour cela, nous avons empruntés les concepts d'arbre de gaussiennes et de classes utilisés dans *SMLLR* et nous les avons introduits dans la technique *EV* pour la dériver en la technique *Structural EigenVoices*.

Nous abordons ci-après la phase de localisation du nouveau locuteur telle qu'elle est effectuée dans *SEV*. Suivront les résultats des expériences menées dans le but d'évaluer l'efficacité de *SEV*, dans le cas d'une adaptation par lot supervisée d'une part, et dans le cas d'une adaptation incrémentale non supervisée d'autre part. Nous exposerons enfin nos conclusions sur l'utilisation de *SEV*.

8.1.1 Localisation du nouveau locuteur

Dans la version classique de *EV*, un locuteur est représenté par un supervecteur s. Ce supervecteur est constitué de l'ensemble des moyennes des gaussiennes des modèles acoustiques, tel que :

$$s = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_{N_G} \end{bmatrix}$$

N_G est le nombre total de gaussiennes du système adapté.
L'adaptation du système indépendant du locuteur à un nouveau locuteur consiste alors à localiser ce nouveau locuteur dans un espace des locuteurs, c'est-à-dire à déterminer les moyennes μ_i du supervecteur s. Plutôt que d'estimer l'ensemble des moyennes de s, *EV* construit un espace propre des locuteurs à partir de l'espace des locuteurs. Cet espace propre permet de réduire le nombre des paramètres à estimer. Il est engendré par $K+1$ voix propres $e(0), e(1), \ldots, e(K)$, où $e(0)$ représente l'origine de l'espace propre des locuteurs dans l'espace des locuteurs. La localisation du nouveau locuteur revient à estimer un vecteur de $K+1$ poids $w_0, w_1, \ldots, w_K$ à l'aide des données d'adaptation disponible pour ce locuteur. Le supervecteur du système adapté pour le nouveau locuteur est obtenu en combinant linéairement ces $K+1$ poids avec les $K+1$ voix propres $e(k)$. Chacun des vecteurs de moyenne μ_i du système adapté, de dimension N_D, pour $i = 1, 2, \ldots, N_G$, s'obtient alors comme suit :

$$\mu_i = \sum_{k=0}^{K} w_k \, e_i(k)$$

$e_i(k)$ correspond au i-ème vecteur (de dimension N_D) extrait à la position $(i \times N_D)$ à partir du supervecteur $e(k)$ (de dimension $N_G \times N_D$).
La phase de localisation dans *EV* consiste donc à déterminer un ensemble de variables d'adaptation, les poids, qui permettent d'exprimer chaque moyenne des gaussiennes du système adapté comme étant fonction des moyennes correspondantes des gaussiennes des modèles acoustiques de plusieurs systèmes dépendant du locuteur.

8.1.1.1 Constitution de "classes acoustiques"

Dans *SEV*, nous avons fait l'hypothèse que les paramètres à adapter, c'est-à-dire les moyennes des gaussiennes des modèles acoustiques du système indépendant du locuteur, ne sont plus regroupées dans un unique supervecteur s, mais dans N supervecteurs $s_1, s_2, \ldots, s_N$. Un locuteur n'est donc plus représenté par un seul supervecteur, mais par N supervecteurs. Sa localisation dans l'espace propre des locuteurs consistera donc à déterminer N vecteurs de poids. Dans le cas où la quantité de données d'adaptation disponibles pour un locuteur est grande, nous supposons que la localisation de ce locuteur sera plus précise lorsque N est grand. En effet, les moyennes de toutes les gaussiennes adaptées ne sont désormais plus exprimée en fonction du même ensemble de poids, mais en fonction d'un ensemble de poids spécifique, si bien qu'elles peuvent être obtenues avec plus de précision. Par exemple, si N est égal au nombre total de gaussiennes des modèles acoustiques du *SIL*, le vecteur de moyenne de chaque gaussienne sera adapté à l'aide d'un ensemble différent de poids.

Un supervecteur est construit à partir d'une "classe acoustique". Une classe acoustique regroupe les gaussiennes des modèles acoustiques qui sont similaires au sens d'une certaine mesure de distance. En ce sens, nous dirons qu'un supervecteur est constitué de caractéristiques acoustiques similaires.

Chaque supervecteur s_i est constitué de vecteurs de moyenne proches les uns des autres au sens de la mesure de distance choisie, et tel que :

$$s_i = \begin{bmatrix} \mu_{(i,1)} \\ \mu_{(i,2)} \\ \vdots \\ \mu_{(i,N_i)} \end{bmatrix} \tag{8.1}$$

N_i est le nombre de vecteurs de moyenne concaténés dans s_i. $\mu_{(i,j)}$ est le j-ème vecteur de moyenne concaténé dans le supervecteur s_i. Il correspond au vecteur de moyenne de la $r(i,j)$-ème gaussienne des modèles acoustiques. $r(i,j)$ est la fonction qui permet de retrouver, à l'issue du processus de constitution des classes acoustiques, le numéro de la gaussienne qui doit être concaténé en j-ème position dans le supervecteur s_i.

La constitution de ces classes acoustiques est réalisée de la même manière que dans *SMLLR*. La procédure *LBG* est tout d'abord utilisée pour construire un arbre de gaussiennes à partir des gaussiennes des modèles acoustiques du système indépendant du locuteur. L'ensemble des classes acoustiques, qui est exploité pour construire les supervecteurs $s_1, s_2, \ldots, s_N$ représentant un locuteur, est alors déterminé en utilisant la méthode utilisée dans *SMLLR* pour définir les classes de régression (4.3.2, page 72). Dans *SEV*, cette méthode est la suivante. Soit α_{SEV} le nombre de trames minimum requises pour estimer de manière robuste les variables d'adaptation (les poids) associées à une classe acoustique. L'arbre de gaussiennes est parcouru à partir des feuilles. Seuls les nœuds disposant d'un nombre suffisant de trames, supérieur à α_{SEV}, sont retenus pour faire partie de l'ensemble de classes acoustiques. A l'instar de *SMLLR*, le nombre de classes acoustiques dépend de la quantité de données d'adaptation disponibles. Plus la quantité de données est grande, plus le nombre de classes acoustiques sera important et inversement.
Une fois les N classes acoustiques définies, chacun des N vecteurs de poids peut alors être estimé et être utilisé afin d'adapter les moyennes des gaussiennes regroupées dans la classe acoustique correspondante.

Des expériences préliminaires sur *SEV* nous ont révélées qu'une dégradation des performances survient si l'estimation des N vecteurs de poids est réalisée lorsque très peu de données d'adaptation sont disponibles. Nous avons donc considéré que, lorsque moins de θ_{SEV} trames acoustiques sont disponibles, une seule classe acoustique est définie dans *SEV*. Cette classe acoustique regroupe alors l'ensemble des gaussiennes des modèles acoustiques du *SIL*. Dans le cas où le nombre de trames disponibles est inférieure à θ_{SEV}, *SEV* revient donc à utiliser l'algorithme *EV*.

8.1.1.2 Localisation des caractéristiques acoustiques d'un locuteur

Puisqu'un locuteur est représenté par N supervecteurs, la localisation dans l'espace propre des locuteurs de ses caractéristiques acoustiques (c'est-à-dire des moyennes des gaussiennes contenues dans les supervecteurs) consiste à déterminer N vecteurs de poids $w_{(1)}, w_{(2)}, \ldots, w_{(N)}$. Chaque vecteur de poids $w_{(i)}$, pour $i = 1, 2, \ldots, N$, est constitué des poids $w_{(i,0)}, w_{(i,1)}, w_{(i,2)}, \ldots, w_{(i,K)}$. Une fois ces N vecteurs de poids estimés, le vecteur de moyennes $\mu_{r(i,j)}$ de chacune des gaussiennes du système adapté est obtenu selon la formule :

$$\mu_{r(i,j)} = \sum_{k=0}^{K} w_{(i,k)}\, e_{r(i,j)}(k) \tag{8.2}$$

pour $i = 1, 2, \ldots, N$ et $j = 1, 2, \ldots, N_i$.

8.1.1.3 Estimation des vecteurs de poids

Les $N \times (K+1)$ poids $w_{(i,k)}$, pour $i = 1, 2, \ldots, N$ et $k = 0, 1, 2, \ldots, K$, sont estimées de telle manière à maximiser la vraisemblance des données d'adaptation $p(O/\Theta)$. O est

l'ensemble des observations issues des données d'adaptation et Θ est l'ensemble des paramètres des modèles acoustiques du système indépendant du locuteur.
L'algorithme *EM* est utilisé pour maximiser $p(O/\Theta)$. Maximiser la vraisemblance $p(O/\Theta)$ revient à maximiser la fonction auxiliaire $Q(\Theta, \hat{\Theta})$ de manière itérative. Soit :

$$Q(\Theta, \hat{\Theta}) = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N_i} \sum_{t=1}^{T} \left[\gamma_{r(i,j)}(t) \; N_D \; log \; (2\pi) \; + \; log \; |\sigma_{r(i,j)}| \; + \; h(o_t, i, j) \right] \tag{8.3}$$

cette fonction auxiliaire, où :

$$h(o_t, i, j) = (o_t \; - \; \hat{\mu}_{r(i,j)})' \; \sigma^{-1}_{r(i,j)} \; (o_t \; - \; \hat{\mu}_{r(i,j)})$$

$\hat{\Theta}$ sont les paramètres des modèles acoustiques du système adapté, $\hat{\mu}_{r(i,j)}$ est le vecteur de moyenne de la $r(i,j)$-ième gaussienne du système adapté, tel que :

$$\hat{\mu}_{r(i,j)} = \sum_{k=0}^{K} \; w_{(i,k)} \; e_{r(i,j)}(k)$$

$e_{r(i,j)}(k)$ représente le vecteur de N_D paramètres situé à la position de la $r(i,j)$-ième gaussienne dans le supervecteur $e(k)$ défini par :

$$e(k) = \begin{bmatrix} e_1(k) \\ e_2(k) \\ \vdots \\ e_{r(i,j)}(k) \\ \vdots \\ e_{N_G}(k) \end{bmatrix}$$

Maximiser Q revient à la dériver et à mettre la dérivée à zéro pour chaque poids $w_{(i,k)}$:

$$\frac{\delta Q}{\delta w_{(i,k)}} = 0 = \sum_{j=1}^{N_i} \sum_{t=1}^{T} \left[\frac{\delta}{\delta w_{(i,k)}} \; \gamma_{r(i,j)}(t) \; h(o_t, i, j) \right] \text{ pour } i = 1, 2, \ldots, N \text{ et } k = 0, 1, \ldots, K$$

$\frac{\delta w_{(i,k)}}{\delta w_{(i,l)}} = 0$ pour $k \neq l$ puisque les voix propres sont orthogonales entre elles.
Nous obtenons alors N systèmes de $K+1$ équations linéaires à $K+1$ inconnues :

$$\sum_{j=1}^{N_i} \sum_{t=1}^{T} \; \gamma_{r(i,j)}(t) \; e_{r(i,j)}(l)' \sigma^{-1}_{r(i,j)} o_t \quad =$$

$$\sum_{j=1}^{N_i} \sum_{t=1}^{T} \; \gamma_{r(i,j)}(t) \; \sum_{k=0}^{K} \; w_{(i,k)} \; e_{r(i,j)}(k)' \; \sigma^{-1}_{r(i,j)} \; e_{r(i,j)}(l) \text{ pour } i = 1, 2, \ldots, N \text{ et } l = 0, 1, 2, \ldots, K \tag{8.4}$$

Chaque système d'équations linéaires peut se réécrire sous forme matricielle comme :

$$v_i = M_i \, w_{(i)} \tag{8.5}$$

où $v_i = \begin{pmatrix} \sum_{j=1}^{N_i} \sum_{t=1}^{T} \gamma_{r(i,j)}(t) \, e_{r(i,j)}(0)' \, \sigma_{r(i,j)}^{-1} \, o_t \\ \sum_{j=1}^{N_i} \sum_{t=1}^{T} \gamma_{r(i,j)}(t) \, e_{r(i,j)}(1)' \, \sigma_{r(i,j)}^{-1} \, o_t \\ \sum_{j=1}^{N_i} \sum_{t=1}^{T} \gamma_{r(i,j)}(t) \, e_{r(i,j)}(2)' \, \sigma_{r(i,j)}^{-1} \, o_t \\ \vdots \\ \sum_{j=1}^{N_i} \sum_{t=1}^{T} \gamma_{r(i,j)}(t) \, e_{r(i,j)}(K)' \, \sigma_{r(i,j)}^{-1} \, o_t \end{pmatrix}$, $M_i = (m_{(i,x,y)})$ et $w_{(i)} = \begin{pmatrix} w_{(i,0)} \\ w_{(i,1)} \\ w_{(i,2)} \\ \vdots \\ w_{(i,K)} \end{pmatrix}$

avec

$$m_{(i,x,y)} = \sum_{j=1}^{N_i} \sum_{t=1}^{T} \gamma_{r(i,j)}(t) \, e_{r(i,j)}(y)' \, \sigma_{r(i,j)}^{-1} \, e_{r(i,j)}(x)$$

Le vecteur de poids $w_{(i)}$ peut alors être obtenu selon la formule :

$$w_{(i)} \quad = \quad M_i^{-1} v_i \tag{8.6}$$

Dans *SEV*, N inversions matricielles sont donc nécessaires pour obtenir les $N \times K$ poids représentant les coordonnées du nouveau locuteur.

8.1.2 Evaluations expérimentales

Afin de rendre compte de l'amélioration de *SEV* par rapport à *EV*, la technique *SEV* fut également évaluée sous les conditions décrites dans le chapitre 3. Nous relatons ici des résultats obtenus.

La figure 8.1 montre les résultats obtenus avec *SEV* en faisant varier le nombre α_{SEV} de trames minimum requises pour estimer de manière robuste les poids associés à une classe acoustique. Ces expériences furent réalisées en utilisant $\theta_{SEV} = 800$, c'est-à-dire que les classes acoustiques ne sont utilisées que si au moins 800 trames acoustiques (ce qui correspond à trois phrases de *RM)* sont disponibles.
Nous pouvons remarquer que α_{SEV} a une influence marginale sur les performances du système adapté obtenue avec *SEV*. Par la suite, nous avons donc décidé d'utiliser $\alpha_{SEV} = 60$.

La figure 8.2 et le tableau 8.1 présentent les performances de *SEV*, de *SMLLR* et de *EV*, lorsqu'elles sont utilisées pour une adaptation par lot supervisée.

Par rapport à *EV*, *SEV* fournit des performances égales lorsqu'une seule phrase d'adaptation est utilisée. *SEV* donne de meilleures performances lorsque plus de trois phrases d'adaptation sont disponibles. Ceci est dû à la capacité de *SEV* à faire varier le nombre de variables d'adaptation en fonction de la quantité de données disponibles.

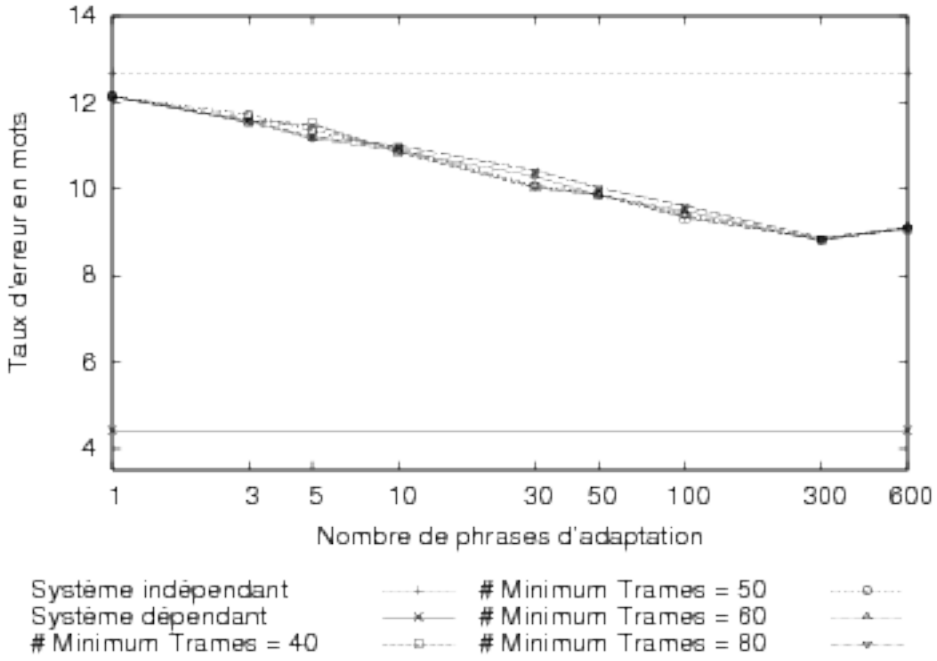


FIGURE 8.1 – Résultats *SEV* - Adaptation par lot supervisée - Variation du nombre minimum de trames requises pour estimer de manière robuste un vecteur de poids

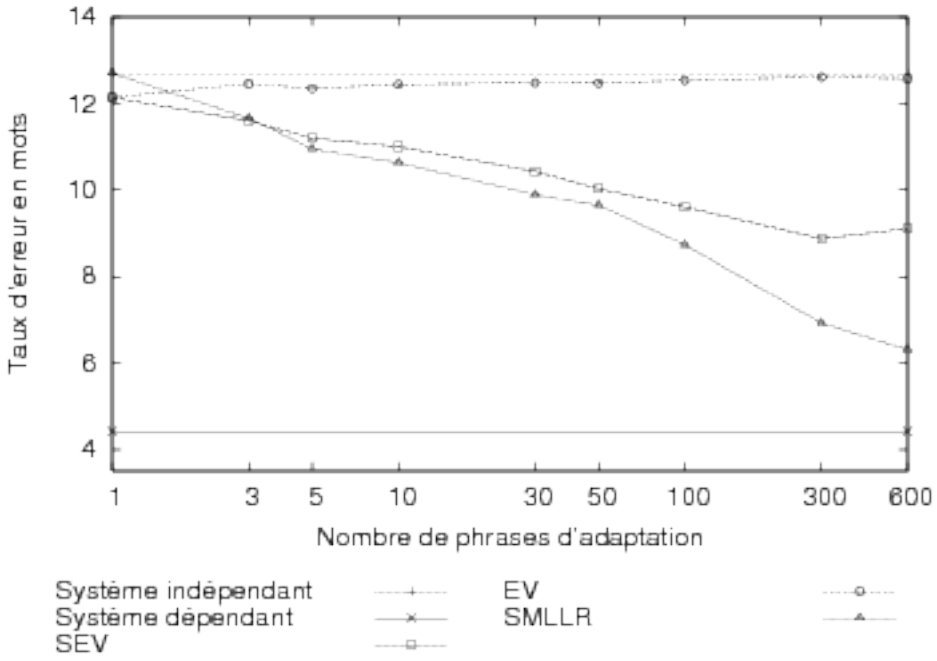


FIGURE 8.2 – Comparaison des performances de *SEV*, *EV* et de *SMLLR* dans le cas d'une adaptation par lot supervisée

	1	5	10	30	50	100	300	600
SEV	**4.4** %	12.1 %	14.3 %	19.0 %	22.4 %	25.3 %	30.6 %	28.3 %
EV	**4.4** %	2.8 %	2.0 %	1.7 %	1.8 %	1.3 %	0.7 %	1.0 %
SMLLR	0 %	**13.8** %	**16.3** %	**22.1** %	**24.0** %	**31.3** %	**45.5** %	**50.3** %

TABLE 8.1 – Comparaison de l'amélioration du taux d'erreur en mots du système indépendant du locuteur en utilisant *SEV*, *SMLLR* et *EV* en mode par lot supervisé

SEV reste cependant moins performant que *SMLLR* lorsque plus de trois phrases d'adaptation sont disponibles. Nous expliquons ce phénomène par le fait que la quantité de données d'adaptation extraite à partir de trois phrases permet d'estimer de manière robuste plus de variables d'adaptation que celles utilisées effectivement dans *SEV*. Le nombre plus important des variables d'adaptation utilisées dans *SMLLR* permet alors d'adapter plus précisément le système indépendant du locuteur, par rapport à *SEV*.

La figure 8.3 et le tableau 8.2 présentent les performances de *SEV*, de *SMLLR* et de *EV*, lorsqu'elles sont utilisées pour une adaptation incrémentale non supervisée.

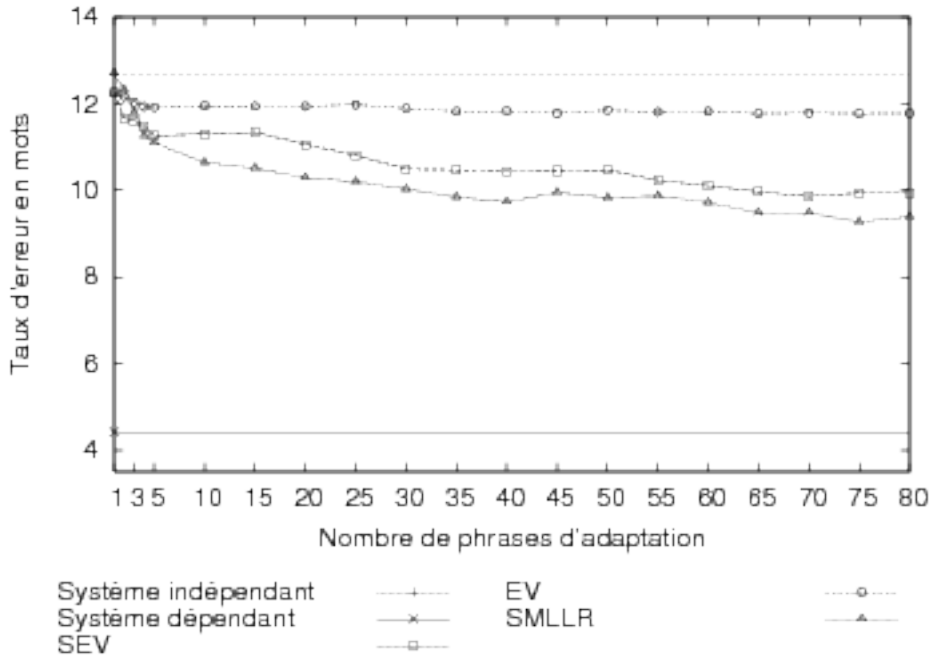


FIGURE 8.3 – Comparaison des performances de *SEV*, *EV* et de *SMLLR* dans le cas d'une adaptation incrémentale non supervisée

Comme c'est le cas pour l'adaptation par lot supervisée, *SEV* est plus performant que *EV* lorsque plus de trois phrases d'adaptation sont disponibles. Elle est plus efficace que *SMLLR* lorsqu'une seule ou trois phrases d'adaptation sont disponibles, mais elle n'amé-

	1	3	5	10	20	30	50	80
SEV	**3.5** %	**8.7** %	11.3 %	11.0 %	12.9 %	17.4 %	17.6 %	21.7 %
EV	**3.5** %	5.2 %	6.1 %	5.9 %	6.0 %	6.4 %	6.7 %	7.2 %
SMLLR	0 %	7.1 %	**12.7** %	**16.2** %	**19.0** %	**21.0** %	**22.6** %	**26.1** %

TABLE 8.2 – Comparaison de l'amélioration du taux d'erreur en mots du système indépendant du locuteur en utilisant *SEV*, *SMLLR* et *EV* en mode incrémental non supervisé

liore pas les performances du système indépendant du locuteur aussi significativement que *SMLLR* lorsque plus de cinq phrases d'adaptation sont disponibles. Les mêmes raisons que celles évoquées précédemment, dans le cas d'une adaptation par lot supervisée, peuvent être données ici dans le cas où *SEV* est utilisée en mode incrémental non supervisé.

La figure 8.4 montre les performances de *SEV* pour une adaptation par lot supervisée et pour une adaptation incrémentale non supervisée.
Les performances du système adapté en mode par lot supervisé sont assez similaires à celles du système adapté en mode incrémental non supervisé. *SEV* peut donc être utilisée efficacement aussi bien en mode par lot qu'en mode incrémental.

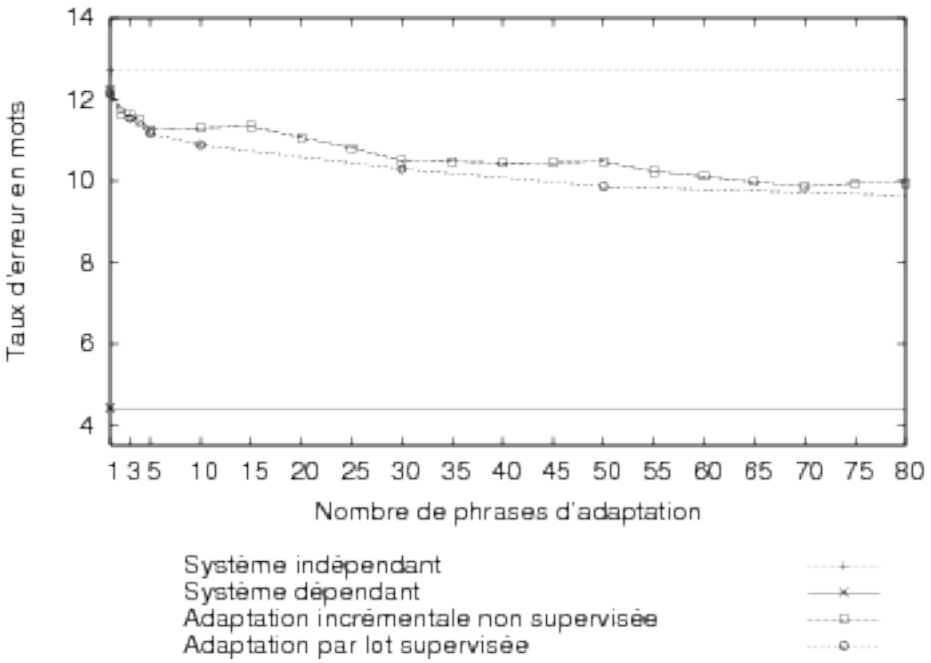


FIGURE 8.4 – Résultats *SEV* - Adaptation par lot supervisée et adaptation incrémentale non supervisée

8.1.3 Etude de complexité

Nous traitons ci-après de la complexité de *SEV* ainsi que de la place mémoire nécessaire pour l'implanter. Les mêmes hypothèses que celles formulées dans le paragraphe 7.2, page 110, seront utilisées.

8.1.3.1 Complexité

L'estimation d'un vecteur de K poids associé à une classe acoustique nécessite l'inversion d'une matrice de dimension $(K+1)\times(K+1)$ (équation 8.6). Cette inversion requiert $\mathcal{O}(K^3)$ opérations. Si *SEV* utilise N classes acoustiques, N inversions de matrices devront alors être réalisées, ce qui nécessite $\mathcal{O}(N \times K^3)$ opérations. Une fois les variables d'adaptation estimées, l'adaptation des N_G vecteurs de moyenne nécessite $\mathcal{O}(N_G \times N_D \times K)$ opérations.
Le nombre total d'opérations N^{SEV} nécessaires pour adapter les moyennes des gaussiennes d'un *SIL* en utilisant *SEV* est donc :

$$N^{SEV} = \mathcal{O}(N \times K^3 + N_G \times N_D \times K) \tag{8.7}$$

8.1.3.2 Place mémoire nécessaire

Comme pour *EV*, la place mémoire majoritairement utilisée par *SEV* correspond essentiellement au stockage des K voix propres, ce qui représente $K \times N_G \times N_D$ paramètres.

8.1.4 Conclusions

Nous avons présenté et évalué dans les précédents paragraphes une nouvelle technique d'adaptation : *SEV*. Cette technique permet d'améliorer les performances de *EV* lorsque la quantité de données d'adaptation disponibles s'accroît, aussi bien dans le cas d'une adaptation par lot que dans le cas d'une adaptation incrémentale.
En outre, la place mémoire requise par *SEV* reste équivalente à celle utilisée par *EV*, et *SEV* reste temps réel dès l'instant où le nombre de phrases d'adaptation disponibles n'est pas trop faible (dans notre cas lorsque plus de trois phrases sont utilisées).

8.2 Combinaisons des techniques *EV* ou *SEV* et *SMLLR*

La technique *SEV* exposée précédemment s'est révélée être aussi performante que *EV* dans le cas d'une adaptation rapide, c'est-à-dire lorsque la quantité de données d'adaptation disponibles est faible (une phrase d'adaptation). Lorsque la quantité disponible de données d'adaptation s'accroît, les performances du système indépendant du locuteur peuvent être améliorées de manière continue avec *SEV*, ce qui n'est pas le cas avec *EV*. Néanmoins, *SEV* reste moins efficace que *SMLLR* dans le cas où la quantité de données d'adaptation est importante.
Afin de disposer d'une technique d'adaptation qui soit efficace quelle que soit la quantité de données d'adaptation disponibles, nous nous sommes orientés vers la conception

de techniques d'adaptation au locuteur qui combinent soit les concepts de *SMLLR* et de *EV* d'une part, soit les concepts de *SMLLR* et de *SEV* d'autre part. Ces travaux ont donné naissance à quatre techniques : *SMLLR+EV*, *SMLLR+SEV*, *EV+SMLLR* et *SEV+SMLLR*. Après avoir exposé les concepts de chacune de ces techniques, nous présenterons les résultats expérimentaux, en terme de taux d'erreurs en mots, que ces techniques permettent de fournir. Ces résultats seront suivis d'une étude de complexité, en terme de temps de calcul et en terme de place mémoire. Nous donnerons enfin nos conclusions sur l'utilisation de ces techniques pour une adaptation continue.

8.2.1 Adaptation *SMLLR* suivie d'une adaptation *EV* ou *SEV* : techniques *SMLLR+EV* et *SMLLR+SEV*

Dans les techniques *SMLLR+EV* et *SMLLR+SEV*, le système indépendant du locuteur est tout d'abord adapté en utilisant *SMLLR*. Le système généré par *SMLLR* est ensuite adapté en utilisant *EV*, dans le cas de la technique *SMLLR+EV*, ou *SEV*, dans le cas de *SMLLR+SEV*. L'origine de l'espace propre des locuteurs, utilisé dans *EV* et dans *SEV*, est le supervecteur construit à partir des moyennes du système généré par *SMLLR*.

Ces approches suggèrent qu'une adaptation basée sur *EV* (respectivement *SEV*) est plus robuste si elle est réalisée après une adaptation *SMLLR*.

8.2.2 Adaptation *EV* ou *SEV* suivie d'une adaptation *SMLLR* : techniques *EV+SMLLR* et *SEV+SMLLR*

Les techniques *EV+SMLLR* et *SEV+SMLLR* consistent à adapter dans un premier temps le système indépendant du locuteur en utilisant respectivement *EV* ou *SEV*. Le système généré par *EV* (respectivement *SEV*) est ensuite adapté en utilisant *SMLLR*. Nous supposons donc ici qu'une adaptation *SMLLR* est plus robuste si elle est réalisée après une adaptation basée sur *EV* (respectivement *SEV*).

8.2.3 Evaluations expérimentales

Nous donnons dans cette section les résultats des expériences que nous avons menées pour évaluer les techniques *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* et *SMLLR+SEV*.

La figure 8.5 présente les performances du système adapté obtenu à partir du système indépendant du locuteur après une adaptation par lot supervisée en utilisant soit *EV+SMLLR*, soit *SEV+SMLLR*, soit *SMLLR+EV*, soit *SMLLR+SEV*. Le tableau 8.3 indique l'amélioration des performances du système adapté par rapport à celles du système indépendant du locuteur, en utilisant l'une des quatre méthodes proposées dans le cas d'une adaptation par lot supervisée.

Nous pouvons remarquer que les quatres méthodes proposées améliorent les performances du système indépendant du locuteur quelle que soit le nombre de phrases d'adaptation

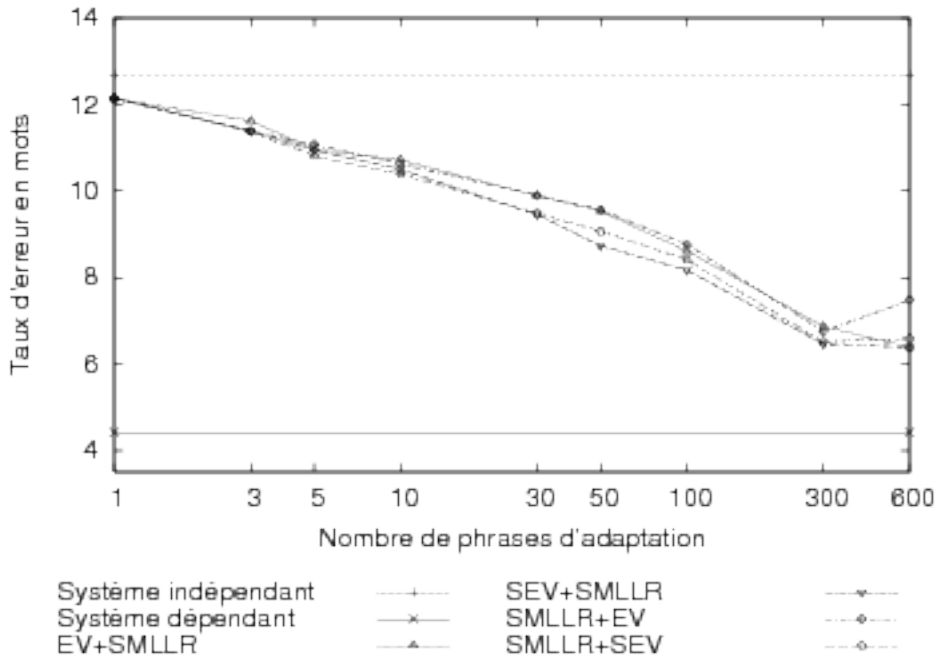


FIGURE 8.5 – Comparaison des performances de *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* et de *SMLLR+SEV* dans le cas d'une adaptation par lot supervisée

	1	5	10	30	50	100	300	600
EV+SMLLR	**4.4** %	13.7 %	15.6 %	22.1 %	25.0 %	32.2 %	45.9 %	**49.8** %
SEV+SMLLR	**4.4** %	13.9 %	17.3 %	**25.6** %	**31.3** %	**35.7** %	**49.1** %	**49.8** %
SMLLR+EV	**4.4** %	12.8 %	16.4 %	22.0 %	24.6 %	30.9 %	46.9 %	41.1 %
SMLLR+SEV	**4.4** %	**14.9** %	**18.2** %	25.4 %	28.6 %	33.8 %	48.7 %	48.1 %

TABLE 8.3 – Comparaison de l'amélioration du taux d'erreur en mots du système indépendant du locuteur en utilisant *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* et *SMLLR+SEV* en mode par lot supervisé

utilisées. En outre, l'amélioration des performances augmente avec le nombre de phrases d'adaptation. Lorsqu'une seule phrase est disponible, l'utilisation de chacune de ces techniques, combinant *SMLLR* avec *EV* ou *SMLLR* avec *SEV*, revient en fait à utiliser uniquement la technique *EV*. En effet, la paramétrisation de *SMLLR* (avec $\alpha_{SMLLR} = 800$) est telle qu'une seule phrase ne suffit pas à opérer une adaptation des modèles acoustiques. La paramétrisation de *SEV* (avec $\theta_{SEV} = 800$) requièrent qu'au moins 800 frames doivent être disponibles afin d'utiliser la flexibilité de *SEV*. Si moins de 800 frames sont disponibles, *SEV* revient donc à effectuer une adaptation *EV*.
Les techniques combinant *SMLLR* et *SEV* donnent systématiquement de meilleurs résultats que celles combinant *SMLLR* et *EV*, quelle que soit le nombre de phrases d'adaptation utilisées. Ceci révèle que les techniques *SMLLR* et *SEV* sont complémentaires, dans le sens où une adaptation obtenue avec la première technique peut être rafinée par une adaptation successive avec la seconde méthode et inversement.
Des quatres techniques, la technique *SEV+SMLLR* fournit les meilleurs résultats lorsque le nombre de phrases d'adaptation disponibles devient important (plus de 30 phrases), tandis que la technique *SMLLR+SEV* est la plus efficace lorsqu'au plus dix phrases d'adaptation sont utilisées.
En fonction de l'application, il sera alors plus judicieux de choisir l'une plutôt que l'autre de ces deux techniques. Si l'application nécessite une adaptation rapide, où peu de phrases sont disponibles, alors il sera avantageux d'utiliser *SMLLR+SEV*. Dans le cas contraire, le choix portera alors plutôt sur *SEV+SMLLR*.

La figure 8.6 et le tableau 8.4 montrent que, dans le cas d'une adaptation par lot supervisée, *SEV+SMLLR* fournit dans l'ensemble des performances supérieures à celles obtenues soit avec *SEV* uniquement, soit avec *SMLLR* uniquement.

	3	5	10	30	50	100	300	600
SEV+SMLLR	**10.3** %	**13.9** %	**17.3** %	**25.6** %	**31.3** %	**35.7** %	**49.1** %	49.8 %
SEV	9.1 %	12.1 %	14.3 %	19.0 %	22.4 %	25.3 %	30.6 %	28.3 %
SMLLR	8.2 %	13.8 %	16.3 %	22.1 %	24.0 %	31.3 %	45.5 %	**50.3** %

TABLE 8.4 – Comparaison de l'amélioration du taux d'erreur en mots du système indépendant du locuteur en utilisant *SEV*, *SMLLR* et de *SEV+SMLLR* en mode par lot supervisé

La figure 8.7 montre les performances du système adapté obtenu à partir du système indépendant du locuteur après une adaptation incrémentale non supervisée, en utilisant soit *EV+SMLLR*, soit *SEV+SMLLR*, soit *SMLLR+EV*, soit *SMLLR+SEV*. Le tableau 8.5 indique l'amélioration des performances du système adapté par rapport à celles du système indépendant du locuteur, dans le cas d'une adaptation incrémentale en utilisant l'une des quatre méthodes proposées.

Dans ce mode, les meilleures performances sont obtenues :

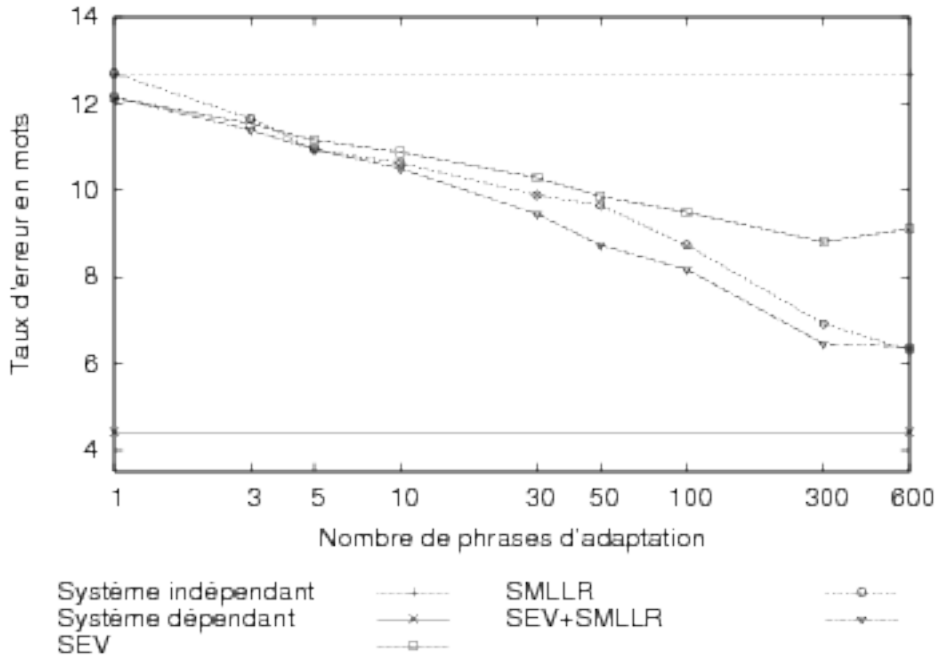


FIGURE 8.6 – Comparaison des performances de *SEV*, *SMLLR* et de *SEV+SMLLR* dans le cas d'une adaptation par lot supervisée

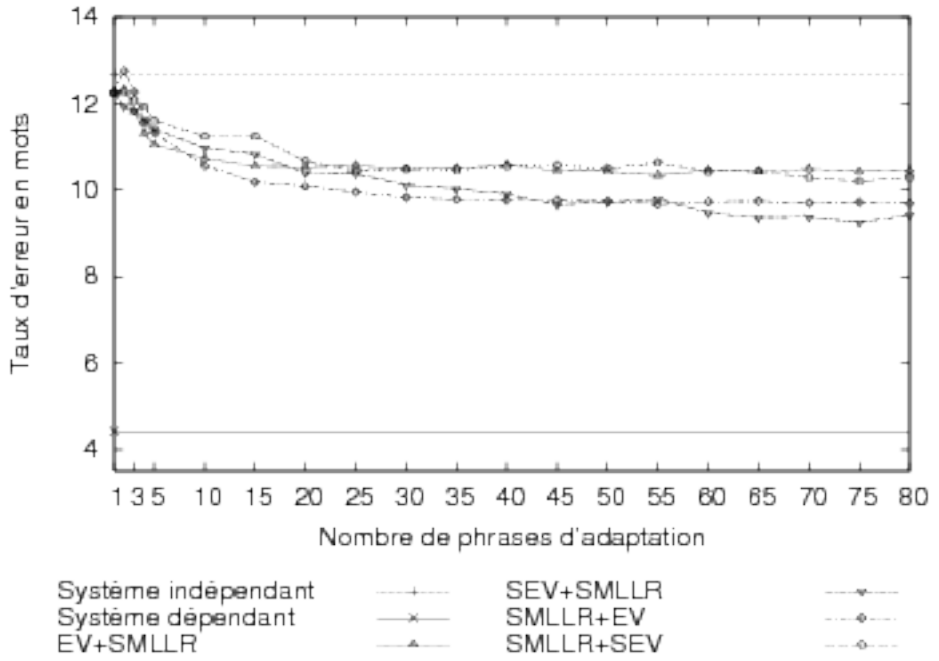


FIGURE 8.7 – Comparaison des performances de *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* et de *SMLLR+SEV* dans le cas d'une adaptation incrémentale non supervisée

	1	3	5	10	20	30	50	80
EV+SMLLR	**3.5** %	**6.9** %	**13.0** %	15.5 %	17.1 %	17.5 %	17.9 %	17.6 %
SEV+SMLLR	**3.5** %	6.8 %	10.2 %	13.5 %	18.1 %	20.3 %	**23.5** %	**25.8** %
SMLLR+EV	**3.5** %	3.5 %	10.9 %	**16.9** %	**20.6** %	**22.6** %	23.3 %	23.7 %
SMLLR+SEV	**3.5** %	5.0 %	8.7 %	11.5 %	16.0 %	17.4 %	17.3 %	18.9 %

TABLE 8.5 – Comparaison de l'amélioration du taux d'erreur en mots du système indépendant du locuteur en utilisant *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* et *SMLLR+SEV* en mode incrémental non supervisé

- avec *EV+SMLLR* lorsqu'au plus cinq phrases d'adaptation sont disponibles,
- avec *SMLLR+EV* lorsqu'au moins dix phrases et au plus trente phrases sont utilisées,
- avec *SEV+SMLLR* lorsqu'entre cinquante phrases et quatre-vingt phrases sont utilisées.

Les techniques combinant *SMLLR* et *SEV* fournissent des résultats décevants comparativement à ceux des techniques combinant *SMLLR* et *EV*. Nous pensons que ces résultats sont dûs aux erreurs de transcription, qui influence plus fortement les performances de *SEV* que celles de *EV*. Dans *SEV*, les variables d'adaptation, qui sont utilisées pour adapter les moyennes de gaussiennes similaires (regroupées dans une même classe acoustique), sont estimées à l'aide des données d'adaptation correspondant aux gaussiennes de la classe. Une erreur de transcription de la phrase d'adaptation peut donc faire correspondre des données incorrectes pour plusieurs gaussiennes appartenant à des classes acoustiques différentes, si bien que les moyennes de ces gaussiennes peuvent être adaptées de manière incorrecte. Ce problème n'apparaît pas dans *EV*, qui utilise l'ensemble des données d'adaptation pour estimer les poids qui entrent en jeu dans l'adaptation de toutes les moyennes des gaussiennes. Le paramétrage de *SMLLR+SEV* et de *SEV+SMLLR* (en l'occurence le choix de θ_{SEV}) peut donc être responsable des résultats décevants de *SMLLR+SEV* et de *SEV+SMLLR*.

La figure 8.8 et le tableau 8.6 montrent que, pour une adaptation incrémentale non supervisée, aucune des techniques *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV*, *SEV* ou *SMLLR* fournit les meilleures performances quelle que soit le nombre de phrases d'adaptation utilisées. Par exemple, la technique *SMLLR+EV* ne donne les meilleures performances que lorsqu'au moins 10 phrases et au plus 30 phrases d'adaptation sont utilisées.

8.2.4 Etude de complexité

Nous présentons ci-après une étude de complexité des quatre techniques proposées, en utilisant les hypothèses formulées dans le paragraphe 7.2, page 110.

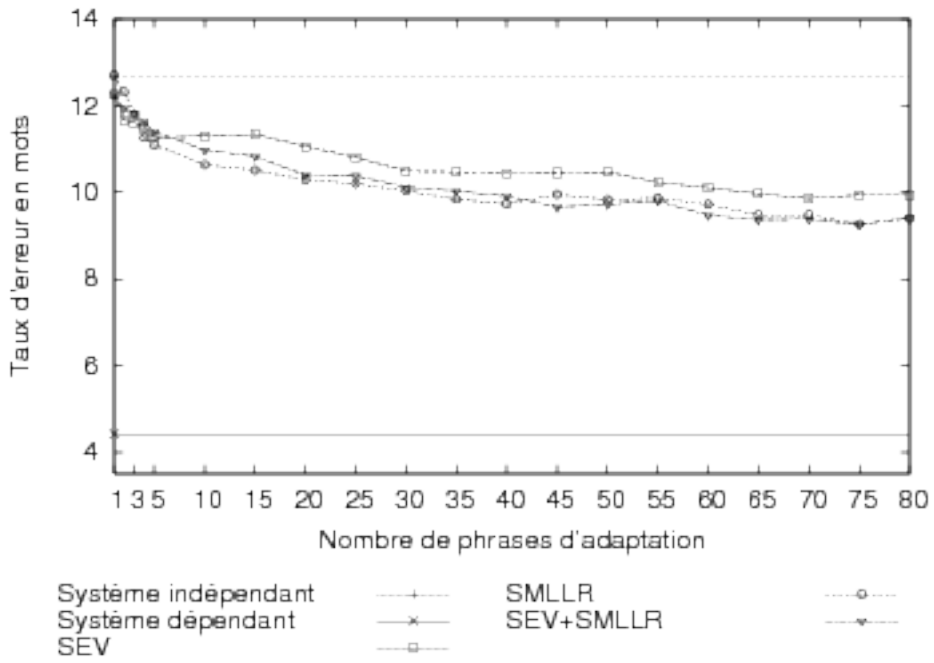


FIGURE 8.8 – Comparaison des performances de *SEV*, *SMLLR* et de *SEV+SMLLR* dans le cas d'une adaptation incrémentale non supervisée

	1	3	5	10	20	30	50	80
EV+SMLLR	**3.5** %	6.9 %	**13.0** %	15.5 %	17.1 %	17.5 %	17.9 %	17.6 %
SEV+SMLLR	**3.5** %	6.8 %	10.2 %	13.5 %	18.1 %	20.3 %	**23.5** %	25.8 %
SMLLR+EV	**3.5** %	3.5 %	10.9 %	**16.9** %	**20.6** %	**22.6** %	23.3 %	23.7 %
SEV	**3.5** %	**8.7** %	11.3 %	11.0 %	12.9 %	17.4 %	17.6 %	21.7 %
SMLLR	0 %	7.1 %	12.7 %	16.2 %	19.0 %	21.0 %	22.6 %	**26.1** %

TABLE 8.6 – Comparaison de l'amélioration du taux d'erreur en mots du système indépendant du locuteur en utilisant *SEV*, *SMLLR* et *SEV+SMLLR* en mode incrémental non supervisé

8.2.4.1 Complexité

La complexité des techniques combinant *SMLLR* et *EV* est la somme des complexités de *SMLLR* et de *EV*. Le nombre total d'opérations $N^{EVSMLLR}$ effectuées dans *EV+SMLLR* et dans *SMLLR+EV* est donc :

$$\begin{aligned} N^{EVSMLLR} &= N^{SMLLR} + N^{EV} \\ &= \mathcal{O}\left(M \times {N_D}^4 + K^3 + N_G \times ({N_D}^2 + N_D \times K)\right) \end{aligned} \tag{8.8}$$

où M est le nombre de matrices de transformation utilisées dans *SMLLR*, K le nombre de voix propres utilisées dans *EV*, N_G le nombre total de gaussiennes constituant les modèles acoustiques et N_D la taille d'un vecteur d'observation.

La complexité des techniques combinant *SMLLR* et *SEV* est la somme des complexités de *SMLLR* et de *SEV*. Le nombre total d'opérations $N^{SEVSMLLR}$ effectuées dans *SEV+SMLLR* et dans *SMLLR+SEV* est donc :

$$\begin{aligned} N^{SEVSMLLR} &= N^{SMLLR} + N^{SEV} \\ &= \mathcal{O}\left(M \times {N_D}^4 + N \times K^3 + N_G \times ({N_D}^2 + N_D \times K)\right) \end{aligned} \tag{8.9}$$

où N est le nombre de vecteurs de K poids estimés dans *SEV*.

8.2.4.2 Place mémoire nécessaire

Dans chacune des quatre techniques *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* ou *SMLLR+SEV*, une adaptation *SMLLR* succède ou précède une adaptation basée sur les *EigenVoices* (adaptation *EV* ou adaptation *SEV*). La place mémoire requise pour implanter chacune de ces techniques correspond donc à la quantité de stockage nécessaire pour sauvegarder les paramètres utilisés par la technique la plus gourmande en place mémoire, à savoir *EV* (et, de manière équivalente, *SEV*). Chacune des quatre techniques proposées nécessitent donc le stockage de K voix propres, ce qui représente $K \times N_G \times N_D$ paramètres.
Le fait de combiner *SMLLR* avec *EV* ou avec *SEV* ne nécessite donc pas de disposer d'un surplus de place mémoire.

8.2.5 Conclusions

Nous avons proposé dans cette section quatres nouvelles techniques d'adaptation au locuteur : *EV+SMLLR*, *SEV+SMLLR*, *SMLLR+EV* et *SMLLR+SEV*.
Pour une adaptation par lot supervisée, la technique *SEV+SMLLR* s'est révélée être la plus performante parmi toutes les techniques évaluées, à savoir parmi *EV+SMLLR*, *SMLLR+EV*, *SMLLR+SEV*, *SEV* et *SMLLR*.
Pour une adaptation incrémentale non supervisée, par contre, aucune des techniques évaluées ne surpassent les autres, en terme d'amélioration des performances du *SIL*. Le choix

d'utiliser, dans ce mode d'adaptation, une technique plutôt qu'une autre pourra alors être guidé soit par des considérations de complexité algorithmique (comme le temps de calcul ou la place mémoire nécessaire à leur utilisation), soit en fonction de l'estimation du nombre total de phrases mises à la disposition pour adapter le *SIL* à un locuteur donné.

Chapitre 9

Algorithmes génétiques pour l'adaptation rapide

Toutes les techniques d'adaptation des modèles acoustiques résolvent un problème d'optimisation numérique. Ces techniques tentent en effet de trouver les meilleurs paramètres des *HMMs* afin qu'ils maximisent une fonction de gain, la log vraisemblance. Cependant, toutes les techniques basées sur l'algorithme *EM*, et étudiées dans les chapitres précédents, sont sous-optimales, dans le sens où elles sont uniquement capables de trouver une solution localement optimale.

Dans le présent chapitre, nous proposons d'utiliser des algorithmes génétiques pour adapter rapidement (avec peu de phrases d'adaptation) les modèles acoustiques d'un système indépendant du locuteur. Plusieurs raisons ont motivé l'exploitation de cette famille d'algorithmes.
D'une part, de tels algorithmes sont théoriquement capables de fournir une solution globalement optimale, en explorant une population de solutions. En outre, les algorithmes génétiques peuvent estimer directement les paramètres des modèles acoustiques, sans utiliser de transformations (comme les régressions linéaires dans *SMLLR*). Ainsi, aucune contrainte *a priori* n'est supposée en ce qui concerne la structure des transformations qui sont appliquées aux paramètres des modèles acoustiques. De cette manière, une adaptation plus précise peut être obtenue.

Ce chapitre est organisé de la manière suivante. Dans la première section sont présentés la genèse et les domaines d'application actuels des algorithmes génétiques. Nous décrivons ensuite les principes généraux de ces algorithmes et évoquons dans la troisième section une preuve de leur convergence. La quatrième section présente un bref état de l'art des composants d'un algorithme génétique. Nous proposons dans la cinquième et la sixième section deux nouvelles techniques d'adaptation basées sur un algorithme génétique. La première technique permet d'adapter directement les moyennes des gaussiennes des modèles acoustiques d'un *SIL*. La deuxième technique utilise la population de solutions générée par un algorithme génétique pour enrichir l'ensemble des systèmes dépendant du locuteur employé par *EV*. Les deux techniques proposées ont été évaluées expérimentalement à l'aide du moteur *ESPERE*, en utilisant le corpus de parole *RM*. Nous présentons dans la sep-

tième section les résultats de ces expériences, et donnons une étude de complexité dans la huitième section. Nous concluerons enfin dans la neuvième section sur l'utilisation de ces deux nouvelles méthodes proposées.

9.1 Genèse et domaines d'application des algorithmes génétiques

L'évolution naturelle a permis d'engendrer des systèmes biologiques qui sont à la fois extrêmement complexes, d'une étonnante diversité et fondamentalement adaptés à leur environnement. Selon Darwin [23], l'évolution des êtres vivants est régie par deux principes fondamentaux : la sélection et la reproduction.

La sélection naturelle agit sur les individus de n'importe quel environnement, au niveau de leur phénotype. Le phénotype d'un individu représente l'ensemble de ces traits caractéristiques, comme la forme et la couleur de ses yeux, le nombre de membres, etc.
Seuls les individus les mieux adaptés à leur environnement, c'est-à-dire ceux possédant certains traits caractéristiques déterminants, survivent, les autres sont condamnées à disparaître.

La reproduction agit au niveau du génotype. Le génotype d'un individu correspond à son patrimoine génétique. Il est constitué d'un ensemble de chromosomes composés de gènes. Chacun des gènes code un trait caractéristique de l'individu.
La reproduction permet aux meilleurs individus uniquement (ceux qui ont survécu à la sélection naturelle) de perpétuer leur patrimoine génétique. De génération en génération, ce matériel génétique subit des modifications constantes, par recombinaisons et mutations des gènes.

Dans les années 1950, sur la base des travaux de Darwin, les biologistes Barricelli et Fraser, entre autres, simulèrent des structures biologiques sur ordinateur : les algorithmes génétiques étaient nés.
Ces travaux furent repris par J. Holland [49], qui établit entre 1960 et 1970 les principes fondamentaux des algorithmes génétiques dans le cadre de l'optimisation mathématique. Il développa également la théorie et les procédures nécessaires à la création de programmes et de machines dotés de capacités illimitées pour s'adapter à des environnements arbitraires [50]. Pour Holland, l'étude de l'adaptation devait passer par l'étude à la fois des systèmes adaptatifs [28] et de leur environnement. Il utilisa alors ces algorithmes pour concevoir de tels systèmes.
L'ouvrage de D. Goldberg [44], qui décrit l'utilisation des algorithmes génétiques dans le cadre de la résolution de problèmes concrets d'optimisation, a permis de mieux faire connaître ces algorithmes et a marqué le début d'un nouvel intérêt pour ces derniers.

Les domaines d'application des algorithmes génétiques sont actuellement nombreux, et

28. c'est-à-dire des systèmes capables d'ajuster leur structure pour s'adapter à leur environnement

concernent, entre autres, l'apprentissage, l'optimisation numérique, l'optimisation combinatoire, la fouille de données et la planification.
Les algorithmes génétiques ont également été utilisés par Spalanzani [102] dans le cadre de l'adaptation à l'environnement pour la reconnaissance automatique de lettres isolées. Ils furent employés d'une part pour adapter des systèmes de *RAP* basés sur des réseaux connexionnistes, d'autre part pour compenser les données acoustiques bruitées. Dans le premier cas, ils ont permis d'améliorer fortement les performances du système de *RAP* dans des environnements difficiles. Ils n'ont toutefois pas permis d'améliorer des systèmes de complexité plus grande (c'est-à-dire comportant un nombre plus élevé de neurones). Dans le second cas, ils ont permis d'améliorer sensiblement les performances d'un réseau de neurones, préalablement entrainé dans un environnement non bruité, afin qu'il puisse convertir les données acoustiques bruitées en données moins bruitées. Le système de *RAP* était alors capable de s'adapter à une séquence de changements d'environnement.

9.2 Principes généraux

Les algorithmes génétiques sont capables de résoudre efficacement des problèmes d'optimisation numérique. La résolution d'un problème d'optimisation numérique est caractérisé par la recherche dans un espace d'états d'une ou de plusieurs solutions optimales maximisant une fonction de gain. Pour résoudre de tels problèmes, les algorithmes génétiques s'appuyent sur des principes de codage dérivés de la génétique et sur les mécanismes de croisements, de mutations et de sélection inspirés de la théorie de l'évolution naturelle de Darwin [23].

Dans la terminologie des algorithmes génétiques, un individu représente une solution potentielle au problème d'optimisation posé. Le génotype d'un individu constitue l'ensemble des paramètres du problème, tandis que son phénotype représente son adéquation au problème.
Mathématiquement, le génotype d'un individu est généralement représenté par un vecteur de paramètres. Ce vecteur symbolise un chromosome. Chaque paramètre de ce vecteur peut alors être considéré comme un gène.
Le phénotype d'un individu est modélisé par une fonction d'évaluation sur ce vecteur de paramètres. Cette fonction représente la fonction à optimiser. La valeur retournée par cette fonction est appelée *fitness*.
Dans le cas d'un problème de maximisation, plus la *fitness* est grande (respectivement petite s'il s'agit d'un problème de minimisation), plus l'individu est adapté et représente une solution adéquate au problème.

Un algorithme génétique exploite une population d'individus, qui évolue génération après génération. A chaque génération, des couples de parents sont formés. Ils sont ensuite recombinés à l'aide d'un opérateur de croisement [29], et engendrent des enfants. Un opérateur de mutation est enfin appliqué aux gènes des enfants avec une certaine probabilité p_m. Les individus les plus aptes, selon leur *fitness*, sont choisis à l'aide d'un opérateur de sélection

29. appelé aussi opérateur de recombinaison

pour faire partie de la population de la génération suivante.
A l'issue d'un certain nombre de générations, l'individu le mieux adapté est considéré comme représentant la solution optimale au problème d'optimisation.
Le fonctionnement générale d'un algorithme génétique est résumé à la figure 9.1.

9.3 Convergence des algorithmes génétiques

La tentative la plus convaincante pour rendre compte de la convergence inhérente des algorithmes génétiques est celle de R. Cerf. Il montra dans [14] qu'un algorithme génétique converge asymptotiquement vers tous les optima lorsque la population initiale dépasse une taille critique (dépendant fortement du problème d'optimisation posé). La démonstration fût établie pour un algorithme génétique muni des trois opérateurs classiques de croisement, de mutation et de sélection, en utilisant la mutation comme opérateur prépondérant et des gènes définis sur des ensembles finis.

9.4 Description détaillée des composants d'un algorithme génétique

Pour rechercher le ou les extrema d'une fonction définie dans un espace de recherche, un algorithme génétique nécessite la définition des cinq éléments suivants :

La structure de données associée aux individus. La structure de données est constitué de l'ensemble des variables du problème à optimiser : c'est le génotype d'un individu. La qualité du codage des données choisi influence fortement la vitesse de convergence de l'algorithme génétique.

La fonction d'évaluation d'un individu. Elle représente la fonction à optimiser.

Un mécanisme de génération de la population initiale. Il doit être capable de générer une population d'individus non homogènes, répartis un peu partout dans l'espace de recherche. Le choix de la population initiale est important car il influence la vitesse de convergence de l'algorithme vers l'optimum global.

Des opérateurs génétiques. Ils permettent de diversifier la population et d'explorer ainsi l'espace de recherche dans toute son étendue. Ces opérateurs sont généralement les opérateurs de croisement, de mutation et de sélection.

Des paramètres liés à l'exploitation de l'algorithme. Il s'agit traditionnellement de la taille de la population, du nombre total de générations ou du critère d'arrêt de l'algorithme, et des probabilités d'application des opérateurs de croisement et de mutation.

9.4.1 Représentation du génotype des individus

Michalewicz [86] suggère de définir une représentation des variables adapté à l'espace de recherche associé au problème à optimiser, et de concevoir des opérateurs génétiques adaptés à cette représentation. Une bonne représentation des variables est caractérisé par

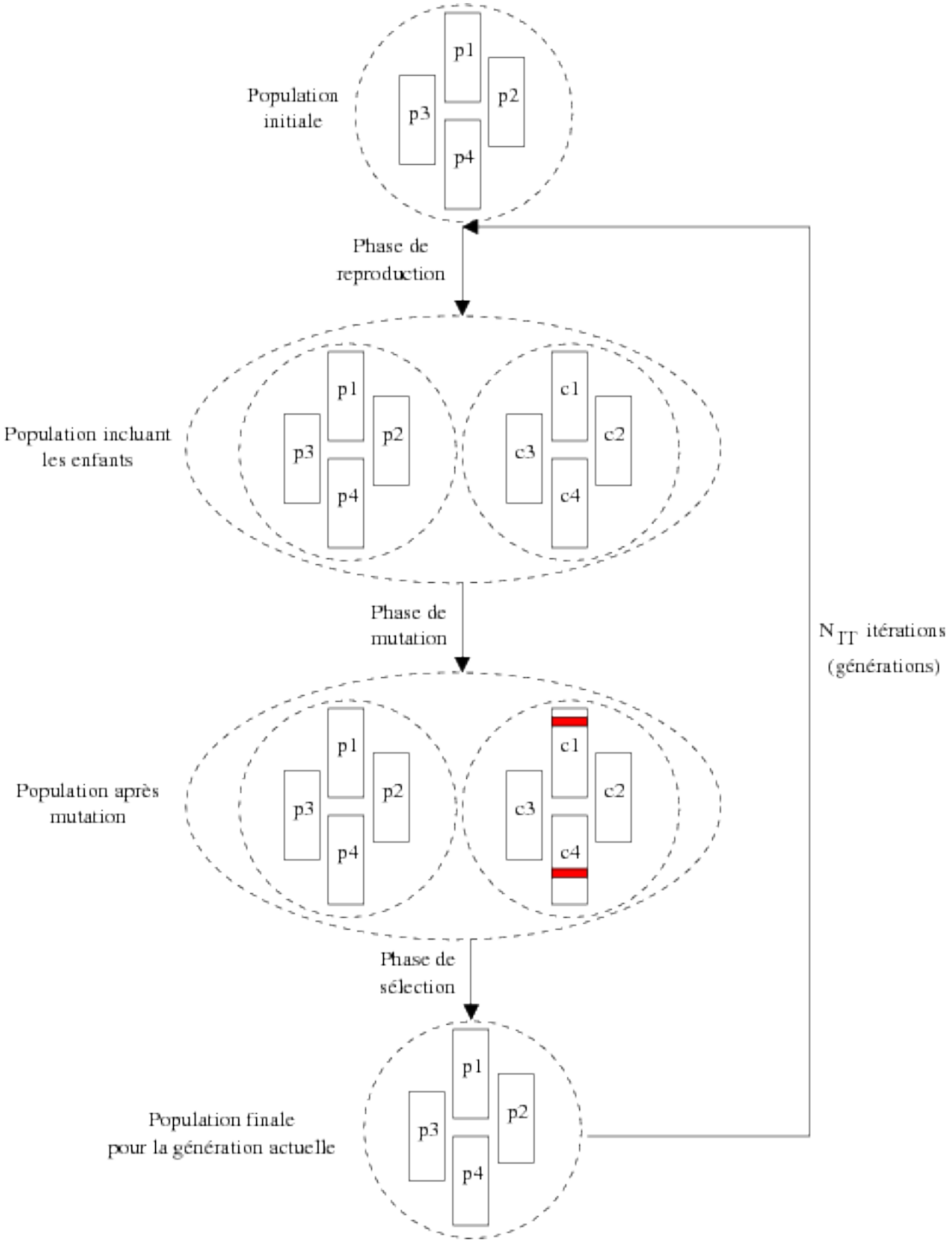


FIGURE 9.1 – Principe général des algorithmes génétiques

le fait que deux génotypes fournissant des solutions proches diffèrent de peu dans leur représentation.

Historiquement, le génotype des individus utilisé par les algorithmes génétiques était représenté par une chaîne de bits. Le codage binaire permet de créer des opérateurs génétiques de croisement et de mutation qui sont simples. Toutefois, dans le cas d'un problème d'optimisation dans un espace de grande dimension, le codage binaire peut rapidement devenir mauvais [86, Chapitre 5, page 97].

Actuellement, la plupart des algorithmes génétiques utilisent des vecteurs réels, qui sont moins longs que les vecteurs de chaînes binaires codant les mêmes variables, donc plus facile à manipuler. En outre, la structure du problème est conservée dans un tel codage.

9.4.2 Génération de la population initiale

Le choix de la population initiale d'individus conditionne fortement la rapidité de convergence de l'algorithme vers un optimum.

Si la position de l'optimum dans l'espace de recherche est inconnue, il est préférable de générer aléatoirement des individus en faisant des tirages uniformes dans chacun des domaines associés aux composantes de l'espace de recherche.

Si par contre des informations *a priori* sur le problème sont disponibles, les individus pourront être générés dans des sous-domaines particuliers de l'espace de recherche, afin d'accélérer la convergence.

9.4.3 Opérateur de croisement

Le croisement a pour objectif d'accroître la diversité de la population en manipulant la structure des chromosomes. Classiquement, l'opérateur est appliqué à un couple de parents et génèrent deux enfants. Plusieurs opérateurs de croisement ont été proposés. Les plus connus sont :

Croisement par découpage : Il consiste à déterminer aléatoirement une ou plusieurs positions de césure (ou **locus**) dans un chromosome. Les parties de chromosome des deux parents situées entre deux locus sont permutées successivement pour générer les chromosomes des deux enfants (figure 9.2). Ce type de croisement est particulièrement efficace pour les problèmes à variables entières.

Croisement barycentrique : Utilisé de préférence dans le cas d'un problème à variables continues. Il consiste à sélectionner deux gènes $p_1(i)$ et $p_2(i)$ dans chacun des parents P_1 et P_2. Ces deux gènes sont situés à la même position i dans les chromosomes des parents. Les deux gènes $e_1(i)$ et $e_2(i)$ situé à la position i dans les chromosomes respectivement du premier enfant E_1 et du second enfant E_2 sont obtenus par combinaison linéaire selon la formule :

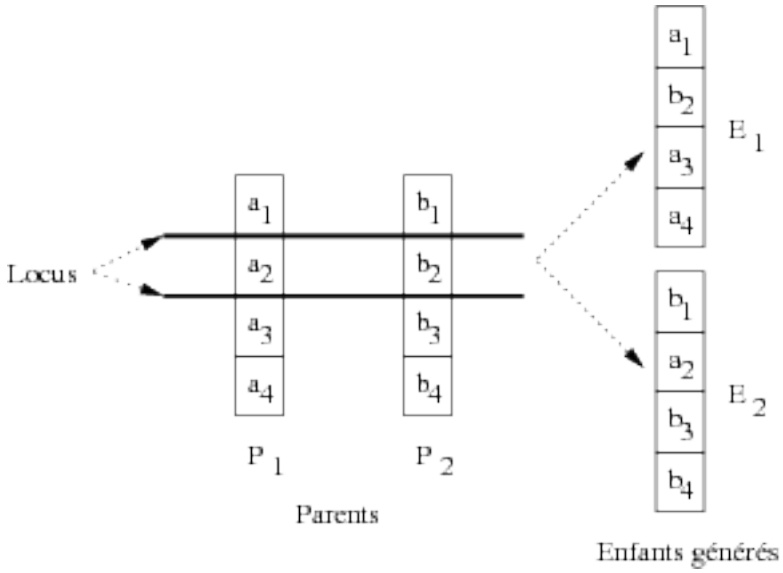


FIGURE 9.2 – Croisement par découpage

$$e_1(i) = i_f\, p_1(i) + (1 - i_f)\, p_2(i) \quad (9.1)$$
$$e_2(i) = (1 - i_f)\, p_1(i) + i_f\, p_2(i) \quad (9.2)$$

i_f est un facteur de pondération aléatoire. Il permet d'étendre le domaine des gènes. Il est déterminé généralement de façon empirique et n'est pas nécessairement compris entre 0 et 1. Il peut, par exemple, prendre des valeurs dans l'intervalle [-0,5 ;1,5], ce qui permet de générer des points entre ou à l'extérieur des deux gènes considérés. Dans le cas où tous les gènes sont croisés de manière barycentrique avec le même facteur i_f, ce principe de croisement peut être étendu aux vecteurs de chromosomes (figure 9.3) :

Cette liste n'est évidemment pas exhaustive. D'autres opérateurs de recombinaison ont été élaborés, notamment dans [86; 95; 103].

9.4.4 Opérateur de mutation

La mutation apporte aux algorithmes génétiques la propriété d'ergodicité de parcours d'espace. Cette propriété indique que l'algorithme est capable d'atteindre tous les points de l'espace, sans pour autant les visiter tous dans le processus de résolution. En toute rigueur, un algorithme génétique peut converger sans opérateur de croisement, et certains fonctionnent d'ailleurs de cette manière : ils sont essentiellement basés sur l'opérateur de mutation. Les propriétés de convergence des algorithmes génétiques sont donc fortement dépendantes de cet opérateur.

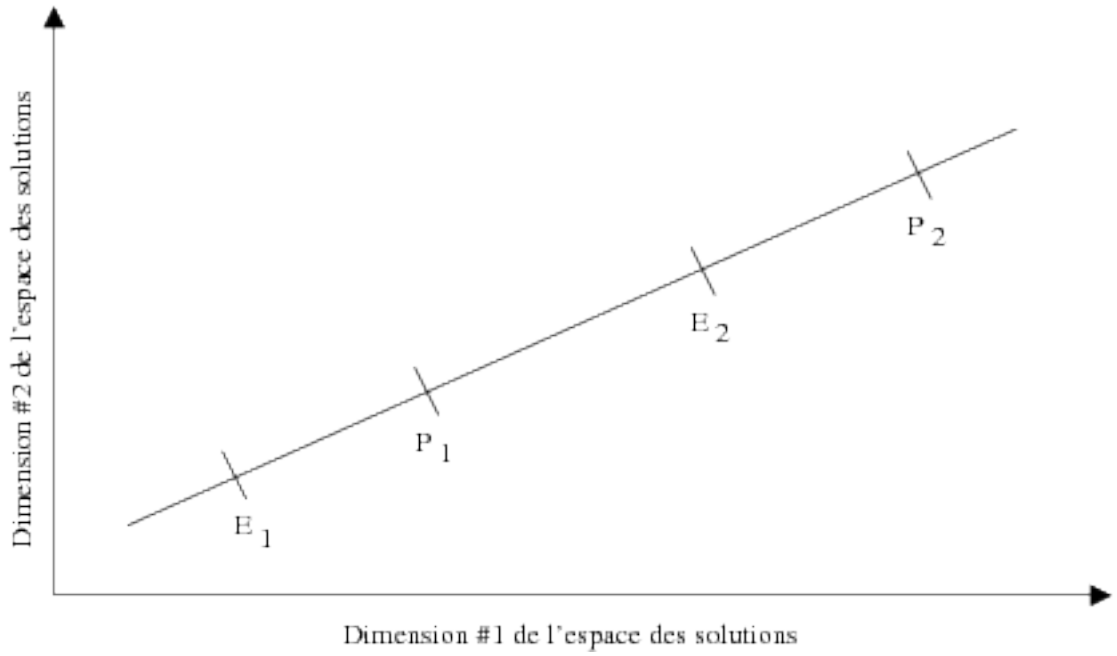


FIGURE 9.3 – Croisement barycentrique

La mutation consiste le plus souvent à déterminer, parmi l'ensemble des N_G gènes composant les chromosomes de la totalité des enfants générés, ceux qui seront modifiés aléatoirement selon une probabilité de mutation p_m. Par exemple, dans la figure 9.4 où la population est de quatre enfants, deux gènes ont été mutés à l'issue de la phase de mutation : le gène a_2 du premier enfant E_1 et le gène c_3 du troisième enfant E_3.

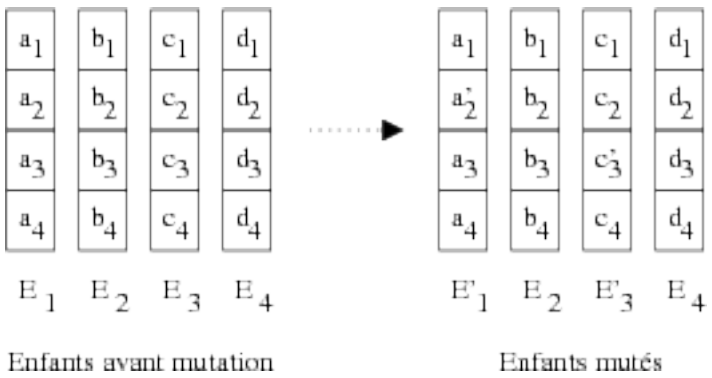


FIGURE 9.4 – Principe de la mutation

Selon la théorie probabiliste, $N_G \times p_m$ gènes ont ainsi été mutés à l'issue de la phase de mutation.

On distingue traditionnellement la **mutation discrète** et la **mutation gaussienne** :

Mutation discrète Utilisée préférentiellement pour les problèmes à variables discrètes, la mutation discrète consiste généralement à remplacer le gène sélectionné par une valeur aléatoire. Une mesure de distance peut être définie afin de choisir la valeur du gène muté comme étant dans le voisinage de la valeur initiale.

Mutation gaussienne Utilisée pour les problèmes à variables continus, la mutation gaussienne consiste à ajouter un bruit gaussien au gène sélectionné. L'écart-type de ce bruit peut être connu *a priori*, ou faire partie du génotype d'un individu. Dans ce dernier cas, il est alors lui-même susceptible d'être modifié par les opérateurs de croisement et de mutation.
Soient μ_i un vecteur de variables réelles codant le gène i sélectionné et σ_i l'écart-type (supposé connu) associé au vecteur μ_i. La formule suivante [86, page 160] permet alors de muter μ_i comme suit :

$$\hat{\mu}_i \quad = \quad \mu_i \; + \; \mathcal{N}(0, \sigma_i) \tag{9.3}$$

$\hat{\mu}_i$ est la nouvelle valeur du gène i, $\mathcal{N}(0, \sigma_i)$ représente un vecteur de variables réelles aléatoires et indépendantes choisies selon la loi normale d'espérance 0 et d'écart-type σ_i.

9.4.5 Opérateur de sélection

La sélection permet de sélectionner les meilleurs individus d'une population et d'éliminer les moins adaptés. Elle est à la fois utilisée pour former les couples de parents et pour déterminer les individus qui appartiendront à la population de la génération suivante.
La sélection peut être déterministe ou stochastique. Une sélection déterministe consiste à choisir les individus en fonction de leur position dans la liste ordonnée des meilleurs individus. Une sélection stochastique consiste à choisir les individus proportionnellement à leur *fitness* : plus la *fitness* d'un individu est grande, plus cet individu aura des chances d'être sélectionné.
Dans le cas où la sélection porte sur la constitution de la population de la génération suivante, la population entière ou seulement une partie de la population peut être remplacée.

La littérature consacrée aux algorithmes génétiques propose un nombre important de principes de sélection plus ou moins adaptés au problème à optimiser. Trois opérateurs de sélection sont toutefois courramment utilisés. Le premier est stochastique, les deux derniers sont déterministes :

- *Roulette Wheel Selection* [30] [44] consiste à sélectionner les individus proportionnellement à leur *fitness*. Un individu ayant une *fitness* élevée aura plus de chance d'être

30. également appelé méthode de Monte-Carlo

séléctionné qu'un individu ayant une petite *fitness*. Le principe de cet opérateur est illustré à la figure 9.5.

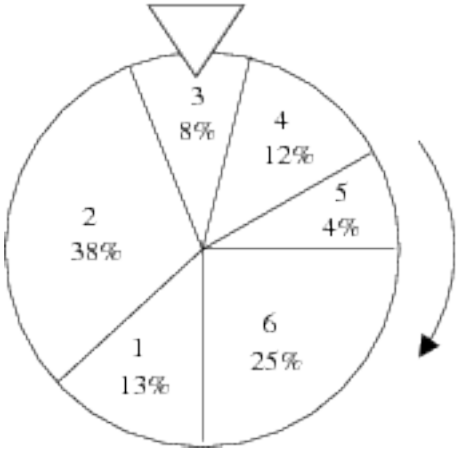


FIGURE 9.5 – Méthode *Roulette Wheel Selection*

Les individus sont répartis sur les secteurs d'un disque (ou roulette). L'aire de chaque secteur est proportionnelle à la *fitness* de l'individu correspondant. La sélection (ou tirage) d'un individu consiste alors à tourner la roulette et à sélectionner l'individu qui appartient au secteur choisi par le hasard. Dans l'exemple de la figure, c'est l'individu 3, possédant 8% de chance d'être sélectionné à chaque tirage, qui a été choisi à l'issue de ce tour de roulette.
Soit f_i la *fitness* de l'individu i. p_i est la probabilité de sélectionner l'individu i et telle que :

$$p_i = \frac{f_i}{\sum_{j=1}^{\mu} f_j}$$

Un tirage consiste alors à générer un nombre aléatoire r de distribution uniforme entre 0 et 1. Si r est compris entre les probabilités cumulées du i-ème et du $(i+1)$-ème individu, alors l'individu i est choisi.

- La stratégie (μ, λ) [86, page 162] consiste à sélectionner les μ meilleurs individus parmi les λ enfants engendrés.

- La stratégie $(\mu + \lambda)$ [86, page 162] consiste à sélectioner les μ meilleurs individus parmi les μ parents et les λ enfants engendrés. Il est supposé que chaque parent engendre λ/μ enfants, avec $\lambda > \mu$. A l'inverse de la stratégie précédente, cette stratégie permet de ne pas éliminer les meilleurs individus d'une génération à une autre. Elle accroît cependant les chances que l'algorithme converge prématurément vers une solution qui n'est pas un optimum global mais local.

9.5 Algorithme génétique appliqué à l'adaptation directe des moyennes des gaussiennes d'un *SRAP*

Nous décrivons dans les paragraphes suivants l'algorithme génétique que nous avons proposé [72] afin d'adapter directement les moyennes des gaussiennes d'un *SIL*. Par la suite, nous ferons référence à cette technique par l'acronyme *GA*.

9.5.1 Codage du génotype d'un individu

Le génotype d'un individu est représenté par un seul chromosome. Un chromosome est constitué de la concaténation des N_G vecteurs de moyenne des gaussiennes d'un *SRAP*, ce qui représente un vecteur de $D = N_G \times N_D$ paramètres.
Pour la définition d'un gène, nous avons émis deux hypothèses :

Hypothèse 1 Soit un gène est codé par un vecteur de moyenne.

Hypothèse 2 Soit un gène est codé par un ensemble de vecteurs de moyenne semblables au sens d'une certaine mesure de distance. Dans ce cas, un algorithme de classification (l'algorithme *LBG* par exemple, annexe B) est utilisé pour regrouper au sein d'une même classe les vecteurs de moyenne semblables.

9.5.2 Génération de la population initiale

La population initiale est formée de $N_I = 73$ individus. Le premier individu correspond au système indépendant du locuteur. Les 72 autres individus correspondent aux 72 systèmes dépendant du locuteur.

9.5.3 Définition de la fonction de *fitness*

La fonction de *fitness* $f(s)$ d'un individu s est définie par :

$$f(s) = \frac{exp\left(\frac{log\ p(O/\Theta_s)}{T}\right)}{\sum_s\ exp\left(\frac{log\ p(O/\Theta_s)}{T}\right)} \tag{9.4}$$

où O, T et Θ_s représentent respectivement la séquence de trames acoustiques extraite des phrases d'adaptation, le nombre de trames acoustiques et les modèles acoustiques de l'individu s.
Le terme $exp\left(\frac{log\ p(O/\Theta_s)}{T}\right)$ représente en fait une approximation de la probabilité $p(O/\Theta_s)$. La valeur de cette probabilité $p(O/\Theta_s)$ dépasse effectivement la capacité numérique des ordinateurs, si bien qu'elle ne peut être calculée directement. C'est ce qui explique le passage au logarithme, sa division par le nombre de trames acoustiques et enfin le recours à l'exponentielle.
L'utilisation de cette fonction $f(s)$ garantit que la *fitness* de chaque individu est comprise entre 0 et 1.

9.5.4 Opérateur de croisement

Cette opérateur consiste, d'une part, à définir $\frac{N_I}{2}$ couples de parents, d'autre part à générer deux enfants pour chaque couple de parents choisis.

9.5.4.1 Sélection des parents

Chaque parent d'un couple est sélectionné parmi les individus de la population actuelle. En outre, nous avons supposé qu'un couple de parents est composé d'individus différents.

Pour sélectionner chacun des parents d'un couple, nous avons utilisé la procédure *Roulette Wheel Selection*, exposée dans le paragraphe 9.4.5, page 143.

9.5.4.2 Génération des enfants

Une fois que les $\frac{N_I}{2}$ couples de parents sont formés, les parents de chaque couple sont fusionnés pour engendrer deux enfants.
Nous avons considéré que le chromosome de chaque enfant est obtenu selon une procédure en deux temps. Dans un premier temps, les chromosomes des enfants sont obtenus en échangeant des groupes de gènes des chromosomes des parents, selon la procédure de croisement par découpage décrite dans le paragraphe 9.4.3, page 140. Dans un second temps, les gènes des chromosomes des enfants, qui ont été recomposés à partir des chromosomes des parents, sont combinés en utilisant un facteur de pondération à l'aide de la procédure de croisement barycentrique (paragraphe 9.4.3).

Les chromosomes des deux enfants sont ainsi obtenus de la manière suivante. Considérons deux parents p_1 et p_2, représentés par des vecteurs contenant trois gènes, tels que :

$$p_1 = [a_1,\ a_2,\ a_3]$$

et

$$p_2 = [b_1,\ b_2,\ b_3]$$

Si un locus a été déterminé après le deuxième gène et si i_f est le facteur de pondération, alors les deux enfants e_1 et e_2 générés à partir des parents p_1 et p_2 sont obtenus comme suit :

$$e_1 = [a_1 \times i_f + b_1 \times (1 - i_f),\ a_2 \times i_f + b_2 \times (1 - i_f),\ b_3 \times i_f + a_3 \times (1 - i_f)]$$

et

$$e_2 = [b_1 \times i_f + a_1 \times (1 - i_f),\ b_2 \times i_f + a_2 \times (1 - i_f),\ a_3 \times i_f + b_3 \times (1 - i_f)]$$

La position de chaque locus est déterminé aléatoirement pour chaque couple de parents. Le nombre de locus et le facteur de pondération i_f sont définis par l'utilisateur et restent inchangés pour chaque couple de parents.

9.5.5 Opérateur de mutation

Soient p_m la probabilité de mutation d'un gène, μ_g un gène (correspond au vecteur de moyenne de la gaussienne g), σ_g la matrice de variances-covariances associée à la gaussienne g dans le système indépendant du locuteur. Pour chaque gène g de chaque enfant généré pendant la phase de reproduction, les deux opérations suivantes sont réalisées :

1. Génération d'un nombre r compris entre 0 et 1 exclu,
2. Mutation du gène g si $r < p_m$.

Dans le cas où le gène g est muté, sa nouvelle valeur $\hat{\mu}_g$ s'obtient selon l'équation :

$$\hat{\mu}_g \;=\; \mu_g \;+\; s \;\times\; \sigma_g \tag{9.5}$$

où s est un nombre généré aléatoirement et compris dans l'intervalle $[-\gamma_m; \gamma_m]$. γ_m est le coefficient de mutation. Il représente le degré de conservation d'un gène : plus γ_m est élevé, plus un gène sera altéré de manière radicale par une mutation.

9.5.6 Opérateur de sélection

La sélection des N_I individus qui seront présents dans la population de la génération future est réalisée selon la stratégie $(\mu+\lambda)$ exposée précédemment (paragraphe 9.4.5, page 144).
Selon cette stratégie, les N_I meilleurs individus (au sens de leur *fitness*) de la population actuelle (parents+enfants générés) sont sélectionnés pour appartenir à la population de la génération suivante.

9.6 Algorithme génétique utilisé pour enrichir l'espace des locuteurs employé par *EigenVoices*

L'algorithme génétique présenté dans la section précédente permet de générer une population finale de N_I solutions. Chaque solution représente un système adapté au nouveau locuteur. Nous avons alors imaginé d'inclure quelques unes de ces solutions dans l'espace de locuteurs employé par la technique *EV* afin d'améliorer ses performances.
Pour cela, parmi les N_I solutions générées par l'algorithme génétique (tel que décrit dans la section précédente), les N_S meilleures solutions sont tout d'abord sélectionnées selon leur *fitness*. Ces N_S solutions sont ensuite ajoutées à l'ensemble des N_I supervecteurs extraits des systèmes dépendant du locuteur et représentant l'espace initial des locuteurs employé par *EV*. C'est ce nouvel espace de locuteurs, formé de $N_I + N_S$ locuteurs, qui est enfin utilisé par *EV* pour adapter le système indépendant du locuteur (figure 9.6).

Nous espérons que l'inclusion de quelques systèmes adaptés au nouveau locuteur dans l'espace initial des locuteurs permet d'enrichir cet espace, ce qui permet à *EV* d'estimer les K poids de manière plus précise.

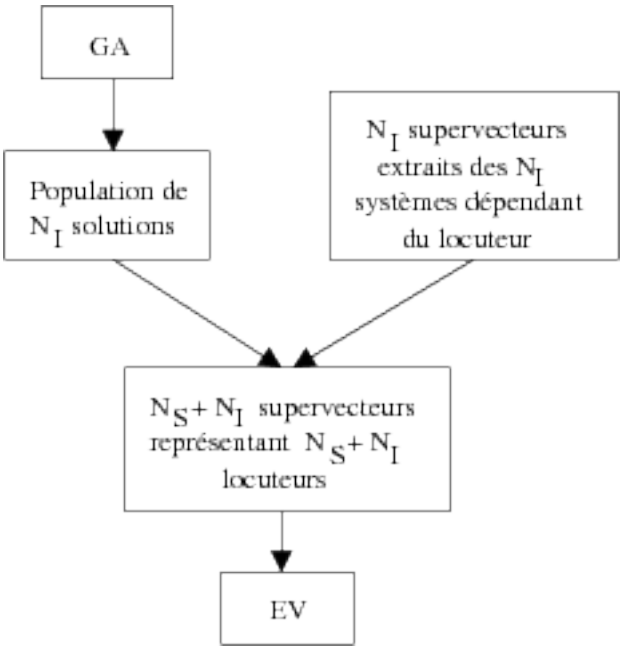


FIGURE 9.6 – Enrichissement de l'espace des locuteurs employé par *EV* en utilisant un algorithme génétique

9.7 Evaluations expérimentales

Les techniques décrites précédemment furent utilisées pour adapter le système indépendant du locuteur à l'aide d'une seule phrase d'adaptation. Nous présentons dans cette section les résultats de ces expériences.

Nous montrons dans le tableau 9.1 l'influence du nombre de générations, de la définition d'un gène (selon l'hypothèse 1 ou selon l'hypothèse 2) et des paramètres i_f (facteur de pondération pour l'extension du domaine des gènes), et γ_m (coefficient de mutation) sur l'amélioration relative des performances du *SRAP*. Etant donné que chacune de ces expériences est coûteuse en temps de calcul, nous n'avons pas eu assez de temps pour évaluer également l'influence de la probabilité de mutation p_m sur l'amélioration des performances.
Dans le meilleur des cas, l'utilisation de la technique *GA* permet d'obtenir une amélioration relative des performances de l'ordre de 5.1%, amélioration qui est supérieure à celle obtenue en utilisant *EV* (qui était de l'ordre de 4.4%).

L'amélioration des performances du système indépendant du locuteur en utilisant la technique *GA*+*EV* est présenté dans le tableau 9.2, en faisant varier le nombre de solutions N_S ajoutées dans l'espace des locuteurs employé par *EV*.
La technique *GA* fut utilisée avec le paramétrage suivant :

- Gènes définis selon l'hypothèse 2,
- $p_m = 0.001$,

				Amélioration relative du taux d'erreur en mots du *SIL*	
Définition des gènes	p_m	i_f	γ_m	20 générations	50 générations
Hyp. 1	0.001	0.4	1.0	**5.1** %	4.3 %
Hyp. 1	0.001	0.4	0.01	**5.1** %	**5.1** %
Hyp. 1	0.001	0.4	0.001	4.6 %	4.7 %
Hyp. 1	0.001	0.2	1.0	1.7 %	2.3 %
Hyp. 1	0.001	0.2	0.01	4.0 %	4.3 %
Hyp. 1	0.001	0.2	0.001	3.5 %	3.1 %
Hyp. 2	0.001	0.4	1.0	4.2 %	*4.6* %
Hyp. 2	0.001	0.4	0.01	4.2 %	4.3 %
Hyp. 2	0.001	0.4	0.001	3.0 %	4.4 %
Hyp. 2	0.001	0.2	1.0	3.3 %	3.3 %
Hyp. 2	0.001	0.2	0.01	2.6 %	3.2 %
Hyp. 2	0.001	0.2	0.001	4.0 %	4.0 %

TABLE 9.1 – Adaptation avec *GA* en utilisant une phrase

- $i_f = 0.4$,
- $\gamma_m = 1.0$
- et 20 générations.

En ce qui concerne *EV*, dix itérations de *EM* furent utilisées et 50 poids furent estimés. Ce paramétrage de *GA* et de *EV* nous a permis d'obtenir les meilleurs résultats.
Comme nous pouvons le constater, cette technique permet, dans le meilleur des cas (pour $N_S = 2, 5,$ ou 10), d'améliorer les performances du *SRAP* de l'ordre de 5.7%, ce qui représente encore une augmentation des performances par rapport à celles de *GA*.

N_S	Amélioration relative du taux d'erreur en mots du *SIL*
2	**5.7** %
5	**5.7** %
10	**5.7** %
20	5.0 %

TABLE 9.2 – Adaptation avec *GA*+*EV* en utilisant une phrase

9.8 Etude de complexité de *GA*

Nous traitons dans cette section de la complexité de *GA* ainsi que de la place mémoire nécessaire pour l'implanter. Pour cette étude, nous utiliserons les mêmes hypothèses que celles formulées dans le paragraphe 7.2, page 110.

9.8.1 Complexité

Une population de solutions dans GA est générée à l'aide des opérateurs de croisement, de mutation et de sélection.
Supposons que N_I est le nombre de parents et que N_E est le nombre d'enfants générés. Après l'application de l'opérateur de croisement, la population est donc composée de $N_I + N_E$ individus. Nous supposerons également que chaque individu est composé de N_G gènes, tous de dimension N_D.
L'opérateur de croisement engendre N_I enfants à partir de N_I parents. La génération d'un enfant requiert $N_G \times N_D$ opérations, ce qui représente un total de $N_I \times N_G \times N_D$ opérations qui sont exécutées par cet opérateur.
L'opérateur de mutation affecte les gènes des N_E enfants générés. Au mieux, cela concerne $p_m \times N_G \times N_E$ gènes, où p_m est la probabilité de mutation. Ainsi, $p_m \times N_G \times N_E \times N_D$ opérations sont réalisées par cet opérateur.
L'opérateur de sélection nécessite le calcul de la *fitness* pour chacun des N_I+N_E individus. Le calcul de la *fitness* d'un individu i requiert le calcul de la vraisemblance des données d'adaptation $p(O/\Theta_i)$. Si T est le nombre total des trames acoustiques O disponibles et que N_S est le nombre total d'états des modèles acoustiques, alors $2 \times T \times N_S$ opérations sont exécutées pour calculer $p(O/\Theta_i)$. Ainsi, $(N_I + N_E) \times T \times N_S$ opérations sont réalisées par l'opérateur de sélection.
Le nombre total d'opérations N^{GA} nécessaires pour adapter les moyennes des gaussiennes d'un *SIL* en utilisant I itérations de GA est donc :

$$N^{GA} = \mathcal{O}\left(I \times [(N_G \times N_D) \times (N_I + p_m \times N_E) + (N_I + N_E) \times T \times N_S]\right) \quad (9.6)$$

A titre indicatif, sur un PC cadencé à 1,4 GHz et disposant de 512 Mo de RAM, le temps de calcul nécessaire pour générer une population de solutions à l'issue de 20 générations en utilisant GA est d'approximativement 30 minutes.

9.8.2 Place mémoire nécessaire

La place mémoire majoritairement utilisée par GA correspond essentiellement au stockage d'une population de solutions. Chaque solution est un supervecteur de dimension $N_G \times N_D$, où N_G représente le nombre total de gaussiennes des modèles acoustiques d'un *SRAP* et N_D représente la taille d'un vecteur d'observation. Si N_I est le nombre de solutions initialement déterminées à partir des systèmes dépendant du locuteur, alors $2\ N_I$ solutions [31] devront être stockées afin de pouvoir être manipulées. Ainsi, $2 \times N_I \times N_G \times N_D$ paramètres devront être sauvegardés.

31. puisque deux parents engendre deux enfants.

9.9 Conclusions

Nous avons proposé dans ce chapitre deux techniques basées sur un algorithme génétique, afin d'adapter un système indépendant du locuteur lorsqu'une seule phrase d'adaptation est disponible.
La première technique, *GA*, adapte directement les moyennes du *SIL* en employant les concepts de reproduction, de mutation et de sélection propres aux algorithmes génétiques. Il a été montré expérimentalement que cette technique permet d'obtenir des performances supérieures à celles obtenues en utilisant la technique *EV*. Cependant, cette technique est plus coûteuse en temps de calcul et plus gourmande en place mémoire que *EV*. Avec une architecture matérielle monoprocesseur, *GA* ne peut pas être utilisée en temps-réel.
La deuxième technique, *GA+EV*, utilise la population finale générée par *GA* pour enrichir l'espace des locuteurs employé par *EV*. Nous avons montré expérimentalement que cette technique est capable d'améliorer encore les performances obtenues par *GA*. A l'instar de *GA*, cette technique n'est cependant pas temps-réel.

Synthèse et perspectives de recherche

Synthèse

L'adaptation en reconnaissance automatique de la parole est devenue une thématique très prolifique depuis une vingtaine d'années. L'idée en est que, à l'instar de notre cerveau, qui est capable d'apprendre continuellement à identifier divers sons véhiculant un message, les systèmes de *RAP* actuels nécessite une adaptation de leur structure interne afin de pouvoir reconnaître une phrase inconnue. C'est pour cette raison que l'utilisation de techniques d'adaptation permet de rendre les systèmes de *RAP* plus robustes au changement de locuteur ou au changement des conditions de l'environnement.

L'adaptation peut être réalisée à différents niveaux d'un système de reconnaissance automatique de la parole : au niveau du signal de parole, des trames acoustiques, des modèles acoustiques, des modèles de langage ou du dictionnaire de prononciation. Dans le cadre de cette thèse, nous nous sommes intéressés à l'adaptation au locuteur des modèles acoustiques d'un système de reconnaissance automatique de la parole, en particulier lorsque les modèles acoustiques sont représentés par des modèles de Markov cachés. Nous avons alors choisi de restreindre notre étude à l'adaptation des moyennes des gaussiennes des modèles acoustiques.

Dans la première partie de cette thèse, nous nous sommes consacrés à disposer d'un ensemble de techniques d'adaptation. Ces techniques nous ont permis d'une part, de mieux comprendre les principes sous-jacents au problème de l'adaptation des modèles acoustiques. D'autre part, étant donné que toutes les techniques étudiées ont été développées et évaluées sur la même plateforme logicielle, elles nous ont également permis de donner une évaluation plus juste des techniques que nous avons proposées ultérieurement.

Nous avons implanté et évalué trois des techniques d'adaptation au locuteur les plus utilisées actuellement. Il s'agit des techniques *Structural Maximum Likelihood Linear Regression*, *Structural Maximum A Posteriori* et *EigenVoices*. Ces techniques furent implantées dans le moteur de reconnaissance *ESPERE* (développé au LORIA) et évaluées à l'aide du corpus de parole *Resource Management (RM)*. Elles furent utilisées pour adapter un système indépendant du locuteur en mode par lot supervisé et en mode incrémental non supervisé. L'étude théorique et expérimentale de ces techniques nous a permis de dégager les deux remarques majeures suivantes. *SMLLR* est la technique la plus efficace (aussi bien en mode par lot supervisé qu'en mode incrémental non supervisé) pour adapter les modèles acoustiques d'un *SRAP* dans le cas où plus de cinq secondes de parole sont disponibles. Dans le cas où une adaptation rapide doit être réalisée, c'est-à-dire en utilisant

moins de cinq secondes de parole, c'est la technique *EV* qui s'est révélée être la plus performante.

Fort de ces constatations, nous avons donc décider de concevoir une technique d'adaptation qui combinent les concepts de *SMLLR* et de *EV*, afin de disposer d'une technique qui améliore les performances d'un *SRAP* lors d'une adaptation continue, c'est-à-dire quelle que soit la quantité disponible de données d'adaptation.

Pour cela, nous nous sommes attachés dans un premier temps à améliorer les performances de la technique classique *EV* lorsqu'elle utilise une quantité de données d'adaptation équivalente à plus de cinq secondes de parole. Du fait que *EV* estime le même nombre de variables d'adaptation quelle que soit la quantité de données d'adaptation disponibles, ses performances sont rapidement saturées dès qu'un certain nombre de phrases sont disponibles. Pour lui permettre de faire varier le nombre de variables d'adaptation à estimer en fonction de la quantité disponible de données d'adaptation, nous avons doté *EV* de la possibilité de définir des classes de gaussiennes. Un vecteur de poids est alors associé à chaque classe et permet d'adapter les gaussiennes qui y sont regroupées. En outre, le nombre de classes varie avec la quantité de données disponibles, à l'instar de *SMLLR*. Cette nouvelle technique, *Structural EigenVoices* (*SEV*), permet ainsi d'améliorer les performances du *SIL* quelle que soit la quantité de données d'adaptation disponibles. Cependant elle reste moins efficace que *SMLLR* lorsque le nombre de phrases d'adaptation est important.

Dans un second temps, nous avons évalué les performances de techniques combinant *SMLLR* avec *EV* d'une part, et *SMLLR* avec *SEV* d'autre part. Quatres techniques ont été validées expérimentalement. Il s'agit des techniques *SMLLR+EV*, *EV+SMLLR*, *SMLLR+SEV* et *SEV+SMLLR*. Le principe de chacune d'elles consiste en fait à exécuter successivement *SMLLR* et *EV* (ou *SEV*) dans un certain ordre. Les expériences menées afin d'évaluer les performances respectives de ces techniques ont montré que *SEV+SMLLR* est la technique qui permet d'obtenir globalement les meilleures performances en mode par lot supervisé, quel que soit le nombre de phrases d'adaptation utilisées. Par contre, en mode incrémental non supervisé, aucune conclusion n'a pu être clairement formulée.

Enfin, dans le cadre de l'adaptation rapide (c'est-à-dire lorsque moins de cinq secondes de parole sont disponibles), nous avons proposé d'utiliser un algorithme génétique pour adapter les modèles acoustiques d'un système indépendant du locuteur. A l'inverse des techniques précédemment étudiées, qui sont toutes basées sur l'algorithme *EM* pour estimer les variables d'adaptation, et qui permettent donc d'obtenir un optimum local de la vraisemblance des données d'adaptation, les algorithmes génétiques permettent théoriquement de déterminer son optimum global.

Deux techniques basées sur un algorithme génétique ont été proposées. La première technique, nommée *GA*, permet d'adapter directement les moyennes des gaussiennes d'un système de reconnaissance automatique de la parole. L'évaluation de *GA* en utilisant une

phrase d'adaptation a montré que *GA* permet d'obtenir une amélioration des performances du *SIL* par rapport à celle obtenue en utilisant *EV*. Ce résultat encourageant nous a aussitôt poussé à considérer la combinaison d'un algorithme génétique avec *EV*, étant donné que la combinaison de techniques s'était révélée fructueuse dans le cadre de l'adaptation continue. La deuxième technique proposée, *GA*+*EV*, utilise la population de solutions générée par *GA* pour enrichir l'espace des locuteurs employé par *EV*. L'idée sous-jacente à cette méthode est d'apporter des informations relatives à un locuteur afin d'améliorer sa localisation par *EV*. Nous avons montré expérimentalement que l'emploi de *GA*+*EV* permet d'améliorer encore les performances du système indépendant du locuteur par rapport à celles obtenues en utilisant *GA* uniquement.

Perspectives de recherche

Les résultats et les conclusions que nous avons évoqués précédemment, en particulier ceux concernant les algorithmes génétiques, ainsi que les hypothèses simplificatrices que nous avons formulées tout au long de cette thèse, et qui sont liées à toute démarche scientifique, nous poussent à poursuivre nos recherches dans plusieurs directions.

En ce qui concerne les algorithmes génétiques...

D'une part, afin de valider l'emploi d'algorithmes génétiques dans le cadre de l'adaptation des modèles acoustiques (et pas uniquement de l'adaptation *rapide*), d'autres expériences doivent être menées en utilisant plusieurs phrases d'adaptation. Nous nous sommes consacrés dans cette thèse à montrer leur efficacité lorsqu'une seule phrase d'adaptation est disponible pour un locuteur donné. Des expériences avec plus d'une phrase d'adaptation doivent donc compléter notre étude.

Par ailleurs, trois voies de recherche restent à approfondir en ce qui concerne l'utilisation des algorithmes génétiques pour la reconnaissance automatique de la parole : la réduction de leur temps de calcul, l'amélioration de leurs performances et leur emploi en tant que méthode alternative à *EM* pour l'estimation de paramètres.

Une manière simple de réduire le temps de calcul de l'algorithme génétique proposé, tout en conservant son efficacité actuelle, est de l'implanter sur une architecture multi-processeurs. Les algorithmes génétiques sont effectivement réputés pour être intrinsèquement parallèles. Pour chaque phase de l'algorithme (croisement, mutation ou sélection), chaque processeur aurait par exemple en charge une partie de la population, ce qui permettrait ainsi de réduire le temps de traitement de la population entière. En outre, au delà de la réduction du temps nécessaire à l'algorithme génétique pour fournir une solution finale, l'implantation sur une architecture multi-processeurs permettrait d'accélerer le processus de conception d'un algorithme génétique plus efficace pour l'adaptation des modèles acoustiques.

En ce qui concerne l'amélioration des performances de l'algorithme génétique, plusieurs directions peuvent être explorées. La première serait de définir une autre fonction de *fitness*. La fonction de *fitness* actuellement utilisée représente la vraisemblance des données d'adaptation calculée sur l'ensemble des chemins possibles, à l'aide de l'algorithme *forward-backward*. On peut également envisager d'utiliser l'algorithme de Viterbi pour

calculer cette vraisemblance.
La seconde manière d'accroître les performances de l'algorithme génétique serait d'explorer l'utilisation d'autres opérateurs génétiques et d'autres combinaisons d'opérateurs génétiques.
La troisième manière serait de disposer d'une population initiale comportant plus d'individus. En effet, plus une population comporte d'individus, plus grandes sont les chances de trouver ou d'engendrer un individu qui est adapté au locuteur actuel. Pour cela, une phase préliminaire de diversification et d'aggrandissement de la population initiale pourrait précéder la recherche d'une solution. Un algorithme génétique, appliqué à l'ensemble des systèmes dépendant du locuteur et ne disposant que d'un opérateur de mutation par exemple, pourrait être employé à cette fin.
Enfin, la dernière manière susceptible d'accroître les performances de l'algorithme génétique porterait sur la définition de nouveaux critères d'arrêt. Actuellement, l'utilisateur doit spécifier le nombre de générations à l'issue duquel une solution est fournie. Or ce paramètre peut influencer grandement la qualité de la solution. Un critère d'arrêt plus pertinent serait de considérer que le meilleur individu de la population actuelle représente la solution finale si sa *fitness* n'a pas changé pendant n générations, n devenant par ailleurs un paramètre plus facile à déterminer pour l'utilisateur que le nombre de générations.

Comme nous l'avons montré expérimentalement dans le chapitre 9, il semblerait que les algorithmes génétiques constituent une alternative performante à l'algorithme *EM* pour l'estimation de paramètres. Dans toutes les techniques classiques étudiées dans ce mémoire (*SMLLR*, *SMAP* et *EV*), l'estimation des variables d'adaptation est réalisée à l'aide de *EM*. Pourquoi alors ne pas estimer ces variables à l'aide d'un algorithme génétique ? Nous envisageons ainsi dans un premier temps d'estimer les poids de *EV* à l'aide d'un algorithme génétique. Si les résultats sont concluants, son emploi pour l'estimation de matrices de régression linéaire pourra par la suite être tenté.
La simplicité et la versatilité des algorithmes génétiques nous permettraient également d'estimer des poids qui représenteraient, non plus une combinaison linéaire de modèles acoustiques, comme c'est le cas dans *EV*, mais une combinaison polynomiale. Nous anticipons alors qu'une telle combinaison pourrait produire des modèles acoustiques qui seraient plus précis que des modèles acoustiques construits à partir d'une combinaison linéaire.

Dans un tout autre cadre, et pour des échéances plus lointaines, nous envisageons également d'utiliser un algorithme génétique afin d'apprendre les paramètres des modèles acoustiques d'un *SRAP*. Il serait effectivement intéressant d'évaluer également leur efficacité dans le cadre de l'apprentissage, d'autant plus que dans ce cas, leur emploi n'est pas limité par des contraintes de temps de calcul, comme cela peut être à l'inverse le cas dans le cadre de l'adaptation.

Sur le choix des modèles acoustiques à adapter

Les modèles acoustiques utilisés actuellement par les systèmes de reconnaissance automatique de la parole sont généralement ceux d'un système indépendant du locuteur. Ces modèles comportent un nombre très important de paramètres (plusieurs centaines de milliers), afin de pouvoir capturer le plus fidèlement possible la variabilité inter-locuteurs et la variabilité intra-locuteur. Or la phase d'adaptation au locuteur nécessite *idéalement* de mettre à jour l'ensemble de ces paramètres sur la base d'une quantité réduite de données d'adaptation. Quelque soit la méthode d'adaptation, nous avons pu nous apercevoir que son efficacité est limitée pour cette raison, dans le cas où un système indépendant du locuteurest utilisé.
L'amélioration de l'efficacité des techniques d'adaptation, qu'elles soient classiques ou basées sur un algorithme génétique, nécessite donc la réduction préalable du nombre de paramètres des modèles acoustiques.
Plusieurs méthodes permettent d'obtenir de tels modèles. Les plus simples, largement employées actuellement, sont celles qui ajustent, pendant l'apprentissage des modèles, le nombre de gaussiennes par état, en éliminant les gaussiennes n'ayant pas reçues d'observations et en "factorisant" éventuellement les gaussiennes qui sont semblables dans plusieurs modèles.
Le mode opératoire que nous préconisons est de disposer d'un ensemble de systèmes de référence dont les modèles acoustiques sont appris à l'aide de phrases provenant de locuteurs ayant des caractéristiques vocales (acoustiques, phonétiques, etc.) semblables. De cette manière, chaque système est supposé ne capturer que la variabilité intra-locuteur, si bien que le nombre de gaussiennes par état peut être choisi à la baisse par rapport à celui fixé pour l'apprentissage d'un système indépendant du locuteur. Chaque système de référence comporterait ainsi moins de paramètres que le système indépendant du locuteur. Le système à adapter, en employant une des méthodes étudiées ou proposées dans ce mémoire, serait alors choisi en fonction de sa proximité par rapport au nouveau locuteur, au sens d'une certaine mesure de distance entre locuteurs.

Pour conclure...

Nous sommes convaincus qu'une technique d'adaptation rapide au locuteur ne peut être efficace que si elle porte sur des modèles acoustiques comportant un nombre restreint de paramètres et que ces paramètres ne devraient prendre en compte qu'*une* variabilité intra-locuteur.
Par ailleurs, étant convaincu que l'architecture future des ordinateurs sera multi-processeurs, et étant donné que les algorithmes génétiques sont intrinsèquement parallèles et que l'un d'eux s'est révélé efficace dans le cas d'une adaptation rapide des modèles acoustiques d'un *SRAP*, nous gageons sur le fait qu'ils constituent une voie de recherche prometteuse dans le cadre de l'estimation des paramètres des modèles acoustiques d'un système de reconnaissance automatique de la parole.

Annexes

Annexe A

Dérivation des formules de réestimation de l'algorithme de Baum-Welch

Etant donné la suite d'observations $O = (o_1\ o_2\ \ldots\ o_T)$, nous cherchons à trouver le modèle $\Theta = ((\pi_i), (a_{ij}), (b_i))$ qui maximise la vraisemblance $p(O/\Theta)$.
L'algorithme de Baum-Welch se base sur le fait suivant.
Soit $\mathcal{P}(\Theta)$ une fonction de la variable Θ qui a la représentation suivante :

$$\mathcal{P}(\Theta) = \int p_+(\alpha, \Theta)\ d\alpha$$

où p_+ est une fonction positive et α est un sous-ensemble de Θ.
Considérons la fonction auxiliaire $Q(\Theta, \hat{\Theta})$ telle que :

$$Q(\Theta, \hat{\Theta}) = \frac{1}{\mathcal{P}(\Theta)} \int p_+(\alpha, \Theta)\ log\ p_+(\alpha, \hat{\Theta}) d\Theta$$

Il est facile de montrer que :

$$Q(\Theta, \hat{\Theta}) - Q(\Theta, \Theta) \leq log\ \mathcal{P}(\Theta)\ -\ log\ \mathcal{P}(\hat{\Theta})$$

Ainsi, si l'on note $T(\Theta) = \underset{\hat{\Theta}}{\text{argmax}}\ Q(\Theta, \hat{\Theta})$, alors $\mathcal{P}(T(\Theta)) \geq \mathcal{P}(\Theta)$.
La suite $\Theta_{(n)} = T(\Theta_{(n-1)})$, avec Θ_0 pris au hasard, est une suite qui augmente la fonction objective $\mathcal{P}(\Theta)$. $\Theta_{(n)}$ converge en fait vers un maximum local de $\mathcal{P}(\Theta)$.

Dans le cas des *HMMs* nous avons :

$$\mathcal{P}(\Theta) = p(O/\Theta) = \sum_{q=q_1,q_2,\ldots,q_T} p(O, q/\Theta) = \sum_{q=q_1,q_2,\ldots,q_T} \underbrace{\pi_{q_1} \prod_{t=2}^{T} a_{q_{t-1}q_t} b_{q_t}(o_t)}_{\text{joue le rôle de } p(q,\Theta)} \qquad (A.1)$$

La fonction auxiliaire $Q(\Theta, \hat{\Theta})$ est définie comme :

$$
\begin{aligned}
Q(\Theta,\hat{\Theta}) &= \frac{1}{p(\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(q,\Theta)\ log\ p(q,\hat{\Theta}) \\
&= \frac{1}{p(\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(q,\Theta)\ \left(log\ \hat{\pi}_{q_1} + \sum_{t=2}^{T} log\ \hat{a}_{q_{t-1}q_t} + \sum_{t=2}^{T} log\ \hat{b}_{q_t}(o_t) \right) \\
&= \frac{1}{p(\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(q,\Theta)\ log\ \hat{\pi}_{q_1} \\
&+ \frac{1}{p(\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(q,\Theta) \sum_{t=2}^{T} log\ \hat{a}_{q_{t-1}q_t} \\
&+ \frac{1}{p(\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(q,\Theta) \sum_{t=2}^{T} log\ \hat{b}_{q_t}(o_t) \\
&= \underbrace{\frac{1}{p(O/\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(O,q/\Theta)\ log\ \hat{\pi}_{q_1}}_{F(\pi)} \\
&+ \underbrace{\frac{1}{p(O/\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(O,q/\Theta) \sum_{t=2}^{T} log\ \hat{a}_{q_{t-1}q_t}}_{F(A)} \\
&+ \underbrace{\frac{1}{p(O/\Theta)} \sum_{q=q_1,q_2,\ldots,q_T} p(O,q/\Theta) \sum_{t=2}^{T} log\ \hat{b}_{q_t}(o_t)}_{F(B)}
\end{aligned}
$$

où $\pi = (\pi_1, \pi_2, \ldots, \pi_{N_S})$, $A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1N_S} \\ a_{21} & a_{22} & \ldots & a_{2N_S} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_S1} & a_{N_S2} & \ldots & a_{N_SN_S} \end{pmatrix}$, et $B = (b_1, b_2, \ldots, b_{N_S})$.
Chaque paramètre π, A et B peut donc être maximisé à part. Dans les paragraphes qui suivent, nous allons voir comment sont calculés A et B.

A.1 Calcul des probabilités de transition A

Il s'agit de trouver A qui maximise :

$$F(A) = \sum_{q=q_1,q_2,\ldots,q_T} p(q/O,\Theta) \sum_{t=2}^{T} log\ \hat{a}_{q_{t-1}q_t} \tag{A.2}$$

sous les contraintes $a_{ij} \geq 0\ \forall i,j = 1,2,\ldots,N_S$ et $\sum_{j=1}^{N_S} a_{ij} = 1\ \forall i = 1,2,\ldots,N_S$.

$$\begin{aligned} F(A) &= \sum_{t=2}^{T} \sum_{q=q_1,q_2,\ldots,q_T} p(q/O,\Theta)\ log\ a_{q_{t-1}q_t} \\ &= \sum_i \sum_j \sum_{t=2}^{T} \sum_{q=q_1,q_2,\ldots,q_T} p(q/O,\Theta)\ log\ a_{ij} \\ &= \sum_i \sum_j a_{ij} \underbrace{\sum_{t=2}^{T} p(q_{t-1}=i, q_t=j/O,\Theta)}_{\xi(i,j)} \end{aligned}$$

donc $F(A) = \sum_{i=1}^{N_S}\ F(A_i)$ où $F(A_i) = \sum_{j=1}^{N_S} \xi(i,j)\ log\ a_{ij}$ et $A_i = (a_{i1}, a_{i2}, \ldots, a_{iN_S})$.
Il suffit donc de maximiser $F(A_i)$ sous les contraintes $a_{ij} \geq 0\ \forall j$ et $\sum_{j=1}^{N_S}\ a_{ij} = 1$.

Selon le théorème des multiplicateurs de Lagrange :

Théorème 1 *Soient $f : \Re^n \to \Re$ et $h_i : \Re^n \to \Re$, $i = 1, 2, m$, des fonctions continument dérivables.*
On s'intéresse à minimiser $f(x)$ sous la contrainte $h(x) = (h_1(x), h_2(x), \ldots, h_m(x)) = 0$. Soit $x^\star$ un minimum local de $h(x) = 0$. Alors il existe un unique $\lambda^\star = (\lambda_1^\star, \lambda_2^\star, \ldots, \lambda_m^\star)$ tel que :

$$\nabla f(x^\star)\ +\ \sum_{i=1}^{m} \lambda_i^{star} \nabla h_i(x^\star) = 0$$

Pour notre problème, il s'agit donc de trouver $(\lambda, \mathcal{A}i)$ tel que $\nabla F(A_i) + \lambda \nabla h(A_i) = 0$, où $h(A_i) = \sum_{j=1}^{N_S} a_{ij} - 1$, c'est-à-dire :

$$\begin{cases} \frac{\partial F(A_i)}{\partial a_{ij}} + \lambda = 0\ \ \forall j \\ \sum_{j=1}^{N_S}\ a_{ij} = 1 \end{cases} \tag{A.3}$$

Or :

$$\frac{\partial F(A_i)}{\partial a_{ij}} = \frac{\xi(i,j)}{a_{ij}}$$

donc le système A.3 devient :

$$\begin{cases} \xi(i,j)\ +\ \lambda\ a_{ij} = 0\ \ \forall j \\ \sum_{j=1}^{N_S}\ a_{ij} = 1 \end{cases} \tag{A.4}$$

En sommant les premières équations on obtient : $\lambda = -\sum_{j=1}^{N_S}\ \xi(i,j)$, d'où :

$$a_{ij} \ = \ \frac{\xi(i,j)}{\sum_{j=1}^{N_S}\ \xi(i,j)} \tag{A.5}$$

A.2 Calcul des probabilités d'observation B

En suivant la même démarche que dans la section précédente, nous avons :

$$\begin{aligned} F(B) &= \sum_{q=q_1,q_2,\ldots,q_T} p(q/O,\Theta) \sum_{t=2}^{T} log\ b_{q_t}(o_t) \\ &= \sum_{i=1}^{N_S} \sum_{t=2}^{T} \sum_{q=q_1,q_2,\ldots,q_T} p(q/O,\Theta)\ log\ b_i(o_t) \\ &= \sum_{i=1}^{N_S} \sum_{t=2}^{T} log\ b_i(o_t)\ p(q_t = i/O,\Theta) \\ &= \sum_{i=1}^{N_S} \underbrace{\sum_{t=2}^{T} log\ b_i(o_t)\ \varphi_t(i)}_{F(B_i)} \end{aligned}$$

Il suffit donc de maximiser $F(B_i)$:

$$\nabla_{\mu_i,\sigma_i} F(B_i) = \sum_{t=2}^{T} \varphi_t(i) \frac{\nabla_{\mu_i,\sigma_i} b_i(o_t)}{b_i(o_t)} \tag{A.6}$$

Or :

$$b_i(o_t) = \frac{1}{\sqrt{(2\pi)^{N_D}\ |\sigma_i|}}\ exp\ \left[-\frac{1}{2}\ (o_t - \mu_i)'\ {\sigma_i}^{-1}\ (o_t - \mu_i)\right]$$

$$\nabla_{\mu_i} b_i(o_t) = b_i(o_t)\ \sigma_i^{-1}\ (o_t - \mu_i)$$

$$\nabla_{\sigma_i} b_i(o_t) = \frac{1}{2}\ b_i(o_t)\ (\sigma_i - (o_t - \mu_i)(o_t - \mu_i))$$

car :

$$\nabla_x (x'\ \sigma^{-1}\ x) = 2\ \sigma^{-1}\ x$$

$$\nabla_\sigma (|\sigma|) = |\sigma| \sigma^{-1}$$

$$\nabla_{\sigma^{-1}} (x'\ \sigma^{-1}\ x) = x\ x'$$

d'où :

$$\begin{aligned} \nabla_{\mu_i} &= \sum_{t=2}^{T} \varphi_t(i)\ \sigma_i^{-1}\ (o_t - \mu_i) \\ &= \sigma_i^{-1}\ \left[\sum_{t=2}^{T} \varphi_t(i) o_t\ - \varphi_t(i)\mu_i\right] \equiv 0 \end{aligned} \tag{A.7}$$

donc :

$$\mu_i = \frac{\sum_{t=2}^{T} \varphi_t(i) o_t}{\sum_{t=2}^{T} \varphi_t(i)} \tag{A.8}$$

et :

$$\nabla_{\sigma_i} F(B_i) = \frac{1}{2} \sum_{t=2}^{T} \varphi_t(i) \left[\sigma_i - (o_t - \mu_i)(o_t - \mu_i)'\right]$$

d'où :

$$\sigma_i = \frac{\sum_{t=2}^{T} \varphi_t(i) (o_t - \mu_i)(o_t - \mu_i)'}{\sum_{t=2}^{T} \varphi_t(i)} \tag{A.9}$$

Annexe B

Algorithme *LBG*

L'algorithme Linde-Buzo-Gray (ou *LBG* [81]) est un algorithme de classification. Il permet de regrouper au sein d'une même classe des individus semblables. Ces individus peuvent être représentés sous forme vectorielle ou sous la forme de lois gaussiennes, par exemple. L'algorithme *LBG* est utilisé lorsque le nombre de classes est connu. Il nous a permis de disposer d'une classification des gaussiennes d'un système indépendant du locuteur à l'aide d'un arbre binaire.

Soient :
- N la profondeur de l'arbre binaire à générer,
- $d(c, g)$ la mesure de distance entre un centre de gravité c et une gaussienne g,
- ε un scalaire utilisé pour perturber le centre de gravité d'un nœud de l'arbre.

N, $d(c, g)$ et ε sont les paramètres de *LBG* qui doivent être spécifiés par l'utilisateur.
La distance de Mahalanobis est la mesure de distance $d(c, g)$ la plus couramment utilisée. Elle est définie par :

$$Mahalanobis(c, g) = (\mu_c - \mu_g)' \, \sigma_g^{-1} \, (\mu_c - \mu_g)$$

où μ_g et σ_g sont respectivement le vecteur moyenne et la matrice de variances-covariances de la gaussienne g ; μ_c est le vecteur associé au centre de gravité c. μ_g et μ_c sont des vecteurs de dimension D, σ_g est une matrice de dimension $D \times D$.

L'algorithme *LBG* construit un arbre binaire de la manière suivante :

1. Soit G le nœud racine de l'arbre contenant la totalité des gaussiennes des modèles.
2. A chaque niveau n de l'arbre, du niveau 1 [32] au niveau $N - 1$

   Pour chaque nœud i du niveau n

   (a) Calcul du centre de gravité c du nœud i. Soit N_i le nombre de gaussiennes regroupées dans le nœud i et $g_{(i,j)}$ la j-ème gaussienne ajoutée dans le nœud i. Alors :

$$\mu_c = \frac{\sum_{j=1}^{N_i} \mu_{g_{(i,j)}}}{N_i}$$

32. On considère que la racine de l'arbre est au niveau 1.

(b) Perturbation du centre de gravité c.
Cette perturbation engendre deux centres de gravité c_1 et c_2 tels que $\mu_{c_1} = \mu_c * (1 - \varepsilon)$ et $\mu_{c_2} = \mu_c * (1 + \varepsilon)$.

(c) Création de deux nœuds fils f_1 et f_2 liés au nœud père i et ayant respectivement pour centre de gravité c_1 et c_2.

(d) Affectation des gaussiennes regroupées dans le nœud père i dans l'un ou l'autre des nœuds fils f_1 ou f_2 :
Pour chaque gaussienne g contenue dans i :

Si $d(c_1, g) < d(c_2, g)$ Alors
Affectez la gaussienne g au nœud f_1

Sinon
Affectez la gaussienne g au nœud f_2

FinSi

FinPour

(e) Affinement de l'affectation des gaussiennes dans les nœuds fils f_1 et f_2 à l'aide de la procédure K-Means :

i. Soit

$$\sigma_1 = \sum_{i=1}^{2} \sum_{j=1}^{N_{f_i}} d(c_i, g_{(f_i,j)}) \tag{B.1}$$

la somme des distances des gaussiennes du nœud père i avec les centres de gravité respectifs des nœuds fils dans lesquelles elles ont été affectées ; $g_{(f_i,j)}$ est la j-ème gaussienne affectée au nœud f_i et N_{f_i} est le nombre de gaussiennes afectées au nœud fils f_i.

ii. Calcul des centres de gravité c_1 et c_2 des nœuds fils respectifs f_1 et f_2.

iii. Réaffectation des gaussiennes dans f_1 et f_2 en utilisant les nouveaux centres de gravité.

iv. Calcul de σ_2 avec la formule B.1.

v. Si $|\sigma_1 - \sigma_2| > \varepsilon$ Alors
$\sigma_1 = \sigma_2$
Retour à l'étape 2.(e)
FinSi

FinPour

FinPour

Bibliographie

[1] M. Afify, Y. Gong, and J. P. Haton. Correlation based Predictive Adaptation of Hidden Markov Models. *Proceedings of the European Conference on Speech Communication and Technology*, 1997.

[2] S. Ahadi and P. Woodland. Combined Bayesian and Predictive Techniques for Rapid Speaker Adaptation of Continuous Density Hidden Markov Models. *Computer Speech and Language*, 11 :187–206, 1997.

[3] L.R. Bahl, F. Jelinek, and R.L. Mercer. A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5 :179–190, 1983.

[4] J.-K. Baker. Stochastic modeling for automatic speech understanding. *Speech Recognition*, 1975.

[5] R. Bakis. Continuous speech recognition via centisecond acoustic states. *91th meeting of the ASA, Washington DC*, 1976.

[6] C. Barras. *Reconnaissance de la parole continue : Adaptation au locuteur et contrôle temporel dans les modèles de Markov cachés.* PhD thesis, Université de Paris VI, 1996.

[7] L.E. Baum. An Inequality and Association Maximisation Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. In *Inequalities*, volume 3, pages 1–8. Academic Press, 1972.

[8] H. Botterweck. Very Fast Adaptation for Large Vocabulary Continuous Speech Recognition using Eigenvoices. *International Conference on Spoken Language Processing*, pages 354–357, 2000.

[9] A. Brun. *Détection de thème et adaptation des modèles de langage pour la reconnaissance automatique de la parole.* PhD thesis, Université Henri Poincaré, Nancy I, 2003.

[10] W. Byrne and A. Gunawardana. Discounted Likelihood Linear Regression for Rapid Speaker Adaptation. *Eurospeech*, pages 203–206, 1999.

[11] W. Byrne and A. Gunawardana. Robust estimation for rapid speaker adaptation using discounted likelihood techniques. *International Conference on Acoustics, Speech, and Signal Processing*, 2000.

[12] W. Byrne and A. Gunawardana. Discounted Likelihood Linear Regression for Rapid Speaker Adaptation. *Computer Speech and Language*, 15(1) :15–38, 2001.

[13] Calliope. *La parole et son traitement automatique*. CNET, Masson, 1989.

[14] R. Cerf. *Une théorie asymptotique des algorithmes génétiques*. PhD thesis, Université Montpellier II, 1994.

[15] J. Chang and V. Zue. A Study in Speech Recognition System Robustness to Microphone Variations : Experiments in Phonetic Classification. *International Conference on Spoken Language Processing*, pages 995–998, 1994.

[16] S. Chen and P. DeSouza. Speaker adaptation by correlation (ABC). *Proceedings of the European Conference on Speech Communication and Technology*, pages 2111–2114, 1997.

[17] C. Chesta, O. Siohan, and C.H. Lee. Maximum A Posteriori Linear Regression for Hidden Markov Model Adaptation. *Eurospeech*, pages 211–214, 1999.

[18] W. Chou. Maximum A Posteriori Linear Regression with Elliptically Symmetric Matrix Priors. *Eurospeech*, pages 1–4, 1999.

[19] W. Chou and X. He. Maximum A Posteriori Linear Regression Variance Adaptation For Continuous Density HMMs. *Eurospeech*, pages 1513–1516, 2003.

[20] D. Van Compernolle, J. Smolders, P. Jaspers, and T. Hellemans. Speaker clustering for dialectic robustness in speaker independent recognition. *Proceedings of the European Conference on Speech Communication and Technology*, pages 723–726, 1991.

[21] S. Cox. Speaker Adaptation using a Predictive Model. *Proceedings of the European Conference on Speech Communication and Technology*, pages 2283–2286, 1993.

[22] S. Cox. Predictive speaker Adaptation in Speech Recognition. *Computer Speech and Language*, 9 :1–17, 1995.

[23] C. Darwin. On the Origin of Species, 1859. Disponible à l'adresse URL : http ://www.infidels.org/library/historical/charles_darwin/origin_of_species/index.shtml.

[24] M.H. DeGroot. *Statistical Decision Theory and Bayesian Analysis*. New York : McGraw-Hill, 1970.

[25] O. Deroo. *Modèles dépendant du contexte et méthodes de fusion de données appliqués à la reconnaissance de la parole par modèles hybrides HMM/MLP*. PhD thesis, Faculté Polytechnique de Mons, Belgique, 1998.

[26] V. Digalakis and L.G. Neumeyer. Speaker adaptation using combined transformation and Bayesian methods. *Transactions on Speech and Audio Processing*, 4(4), 1996.

[27] V. Digalakis, D. Ritchev, and L. Neumeyer. Speaker adaptation using Constrained Estimation of Gaussian Mixtures. *Transactions on Speech and Audio Processing*, 3 :357–366, 1995.

[28] V.V. Digalakis. Online Adaptation Hidden Markov Models using Incremental Estimation Algorithms. *Transactions on Speech and Audio Processing*, 7(3) :253–261, 1999.

[29] S.-J. Doh and R.M. Stern. Inter-Class MLLR for Speaker Adaptation. *International Conference on Acoustics, Speech, and Signal Processing*, pages 1543–1546, 2000.

[30] Dragon naturally speaking. Publié par Dragon Systems, Inc., 1997. Disponible à l'adresse URL : http ://www.dragonsystems.com/marketing/pcproducts.html.

[31] E. Eide and H. Gish. A parametric approach to Vocal Ttract Length Normalisation. *International Conference on Acoustics, Speech, and Signal Processing*, pages 346–349, 1996.

[32] D. Fohr, O. Mella, and C. Antoine. The Automatic Speech Recognition Engine ESPERE : Experiments on Telephone Speech. *International Conference on Spoken Language Processing*, pages 246–249, 2000.

[33] G.D. Forney. The Viterbi Algorithm. *Proceedings of the IEEE*, 61(3), 1973.

[34] S. Furui. Unsupervised speaker adaptation method based on hierarchical spectral clustering. *International Conference on Acoustics, Speech, and Signal Processing*, pages 286–289, 1989.

[35] M.J.F. Gales. The Generation and Use of Regression Class Trees for MLLR Adaptation. Technical report, Cambridge University Engineering Department, 1996.

[36] M.J.F. Gales. Cluster Adaptive Training for Speech Recognition. *International Conference on Spoken Language Processing*, pages 1783–1786, 1998.

[37] M.J.F. Gales. Maximum Likelihood Linear Transformations for HMM-based Speech Recognition. *Computer Speech and Language*, 12 :75–98, 1998.

[38] M.J.F. Gales. Cluster Adaptive Training of Hidden Markov Models. *Transactions on Speech and Audio Processing*, 8(4) :417–428, 2000.

[39] M.J.F. Gales, D. Pye, and P.C. Woodland. Variance Compensation within the MLLR Framework for Robust Speech Recognition and Speaker Adaptation. In *International Conference on Spoken Language Processing*, volume 3, pages 1832–1835, 1996.

[40] M.J.F. Gales and P.C. Woodland. Mean and Variance Adaptation within the MLLR Framework. *Computer Speech and Language*, 10 :249–264, 1996.

[41] Y. Gao, M. Padmanabhan, and M. Picheny. Speaker adaptation based on pre-clustering training speakers. *Proceedings of the European Conference on Speech Communication and Technology*, pages 2091–2094, 1997.

[42] J.-L. Gauvain and C.-H. Lee. Maximum A Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains. *Transactions on Speech and Audio Processing*, 2(2) :291–298, 1994.

[43] V. Goel, K. Visweswariah, and R. Gopinath. Rapid Adaptation with Linear Combinations of Rank-One Matrices. *International Conference on Acoustics, Speech, and Signal Processing*, 1 :581–584, 2002.

[44] D.E. Goldberg. *Genetic Algorithm in search, optimization and machine learning.* Addison-Wesley, 1989.

[45] Y. Gotoh and H.F. Silverman. Incremental ML Estimation of HMM Parameters for Efficient Training. *International Conference on Acoustics, Speech, and Signal Processing*, pages 585–588, 1996.

[46] J.-P. Haton. Les modèles neuronaux et hybrides en reconnaissance automatique de la parole : états des recherches. *Ecole Thématique sur les Fondements et Perspectives en Traitement Automatique de la Parole, Centre de Formation du CNRS de Marseille-Luminy*, 1995.

[47] T. J. Hazen. Probabilistic Transfer Vector Prediction for Speaker Adaptation. Technical report, 1995.

[48] T.J. Hazen. *The Use of Speaker Correlation Information for Automatic Speech Recognition*. PhD thesis, Massachusetts Institute of Technology, 1998.

[49] J.H. Holland. Springer-Verlag, 1962.

[50] J.H. Holland. *Adaptation in natural and artificial systems*. Michigan Press, 1975.

[51] J.J. Humphries and P.C. Woodland. Accent modelling and adaptation in automatic speech recognition. *Transactions on Speech and Audio Processing*, 1999.

[52] Q. Huo and C.-H. Lee. On-Line Adaptive Learning of the Continuous Density Hidden Markov Model based on Approximate Recursive Bayes Estimate. *Transactions on Speech and Audio Processing*, 5(2) :161–171, 1997.

[53] IBM ViaVoice. Publié par IBM Corporation, 1997. Disponible à l'adresse URL : http ://www.software.ibm.com/is/voicetype/us_vv.html.

[54] I. Illina. *Extension du modèle stochastique des mélanges de trajectoires pour la reconnaissance automatique de la parole continue*. PhD thesis, Université Henri Poincaré, Nancy I, 1997.

[55] F. Jelinek. Continuous Speech Recognition by Statistical Methods. *Proceedings of the IEEE*, 64(4) :532–556, 1976.

[56] F. Jelinek. Self-Organized Language Modeling for Speech Recognition. *Ed. by A. Waibel and Kai-fu Lee, Morgan Kauffmann*, pages 450–506, 1990.

[57] F. Jelinek, R.L. Mercer, and L.R. Bahl. Continuous Speech Recognition : Statistical methods. In *Handbook of statistics*, volume 2, pages 549–573. North-Holland Publishing Company, P.R. Krishnaiah and L.N. Kanal editions edition, 1982.

[58] D. Jouvet. Reconnaissance de mots connectés indépendamment du locuteur par des méthodes statistiques, 1988.

[59] D. Jouvet, M. Dautremont, and A. Gossart. Comparaison des multimodèles et des densités multigaussiennes pour la reconnaissance de la parole par modèles de Markov. *Journées d'Etude sur la Parole*, pages 159–164, 1994.

[60] J.C. Junqua. The Lombard Reflex and its Role on Human Listeners and Automatic Speech Recognizers. *Journal of Acoustical Society of America*, pages 510–524, 1993.

[61] T. Kamm, A. Andreou, and J. Cohen. Vocal Ttract Normalisation in speech recognition : compensating for systematic speaker variability. *Proceedings of the Annual Speech Research Symposium*, 1997.

[62] J. Kim and J. Chung. Reduction of Dimension of HMM Parameters Using ICA and PCA in MLLR Framework for Speaker Adaptation. *Eurospeech*, pages 1461–1464, 2003.

[63] T. Kosaka, S. Matsunaga, and S. Sagayama. Tree-structured speaker clustering for speaker-independent continuous speech recognition. *International Conference on Spoken Language Processing*, pages 1375–1378, 1994.

[64] T. Kosaka and S. Sagayama. Tree-structured speaker clustering for fast speaker adaptation. *International Conference on Acoustics, Speech, and Signal Processing*, 1 :245–248, 1994.

[65] R. Kuhn, J.-C. Junqua, P. Nguyen, and N. Niedzielski. Rapid Speaker Adaptation in Eigenvoice Space. *Transactions on Speech and Audio Processing*, 8(6) :695–707, 2000.

[66] R. Kuhn, P. Nguyen, J.-C. Junqua, and al. Eigenvoices for Speaker Adaptation. *International Conference on Spoken Language Processing*, 1998.

[67] R. Kuhn, P. Nguyen, J.-C. Junqua, R. Boman, N. Niedzielski, S. Fincke, K. Field, and M. Contolini. Fast Speaker Adaptation using A Priori Knowledge. *International Conference on Acoustics, Speech, and Signal Processing*, pages 1587–1590, 1999.

[68] D. Langlois. *Notions d'évenements distants et d'évenements impossibles en modélisation stochastique du langage : application aux modèles n-grammes de mots et de séquences*. PhD thesis, Université Henri Poincaré, Nancy I, 2003.

[69] M. J. Lasry and R.M. Stern. A Posteriori Estimation of Correlated Jointly Gaussian Mean Vectors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1984.

[70] F. Lauri, I. Illina, and D. Fohr. Comparaison de SMLLR et de SMAP pour une adaptation au locuteur en utilisant des modèles acoustiques markoviens. *Journées d'Etude sur la Parole*, 2002.

[71] F. Lauri, I. Illina, and D. Fohr. Combining Eigenvoices and Structural MLLR for Speaker Adaptation. *International Conference on Acoustics, Speech, and Signal Processing*, 2003.

[72] F. Lauri, I. Illina, and D. Fohr. Using Genetic Algoritms for Rapid Speaker Adaptation. *Eurospeech*, 2003.

[73] K.-F. Lee and F. Alleva. Continuous Speech Recognition. In Sadaoki Furui and M. Mohan Sondhi, editors. *Advances in Speech Signal Processing*, pages 623–650, 1992.

[74] L. Lee and R.C. Rose. Speaker Normalisation using Efficient frequency warping procedures. *International Conference on Acoustics, Speech, and Signal Processing*, pages 353–356, 1996.

[75] S. Lee, A. Potamianos, and S. Narayanan. Analysis of Children's Speech Duration, Pitch and Formants. *Eurospeech*, 1997.

[76] C.J. Leggetter. *Improved Acoustic Modeling for HMMs using Linear Transformations*. PhD thesis, Cambridge University, 1995.

[77] C.J. Leggetter and P.C. Woodland. Speaker Adaptation of Continuous Density HMMs using Multivariate Linear Regression. In *International Conference on Spoken Language Processing*. Cambridge University Engineering Department, 1994.

[78] C.J. Leggetter and P.C. Woodland. Flexible Speaker Adaptation using Maximum Likelihood Linear Regression. *Eurospeech*, pages 1155–1158, 1995.

[79] C.J. Leggetter and P.C. Woodland. Maximum Likelihood Linear Regression for Speaker Adaptation of Continous Density Hidden Markov Models. *Computer Speech and Language*, 9 :171–185, 1995.

[80] S.E. Levinson, L.R. Rabiner, and M.M. Sondhi. An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition. *The Bell System Technical Journal*, 62(4) :1035–1074, 1983.

[81] Y. Linde, A. Buzo, and R.M. Gray. An algorithm for vector quantizer. *IEEE Transactions On Communication*, 28 :84–95, 1980.

[82] A. Ljolje. Speaker clustering for improved speech recognition. *Proceedings of the European Conference on Speech Communication and Technology*, pages 631–634, 1993.

[83] J.F. Mari. Perception de signaux complexes et interaction homme-machine. Habilitation à diriger des recherches, spécialité informatique, Centre de Recherche en Informatique de Nancy, 1996.

[84] J.F. Mari, D. Fohr, and J.P. Haton. Modèles stochastiques d'ordres 1 et 2. *XXèmes Journées d'Etude sur la Parole, Trégastel*, pages 199–202, 1994.

[85] L. Mathan and L. Miclet. Speaker hierarchical clustering for improving speaker-independent HMM word recognition. *International Conference on Acoustics, Speech, and Signal Processing*, pages 149–152, 1990.

[86] Z. Michalewicz. *Genetic Algorithm + Data Structures = Evolution Programs.* Springer-Verlag, 1996.

[87] R.M. Neal and G.E. Hinton. A View of the EM Algorithm that justifies Incremental, Sparse, and other Variants. In *Learning in Graphical Models*, pages 355–368. Kluwer Academic Publishers, Norwell MA, 1998.

[88] S.K. Ng and G.J. McLachlan. On the Choice of the Number of Blocks with the Incremental EM Algorithm for the Fitting of Normal Mixtures. Technical report, Centre for Statistics, University of Queensland, 2001.

[89] P. Nguyen. Fast Speaker Adaptation. Technical report, Speech Technology Laboratory, 1998.

[90] M. Padmanabhan, L.R. Bahl, D. Nahamoo, and M.A. Picheny. Speaker Clustering and Transformation for Speaker Adaptation in Large-Vocabulary Speech Recognition Systems. *International Conference on Acoustics, Speech, and Signal Processing*, pages 701–704, 1996.

[91] D. Pye and P.C. Woodland. Experiments in Speaker Normalisation and Adaptation for Large Vocabulary Speech Recognition. *International Conference on Acoustics, Speech, and Signal Processing*, pages 1047–1050, 1997.

[92] L.R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2) :257–286, 1989.

[93] A. Sankar, F. Beaufays, and V. Digalakis. Training Data Clustering for Improved Speech Recognition. *Eurospeech*, pages 503–506, 1995.

[94] R. Schwartz and Y.L. Chow. The optimal N-Best algorithm : an efficient procedure for finding multiple sentence hypotheses. *International Conference on Acoustics, Speech, and Signal Processing*, pages 81–84, 1990.

[95] M. Sebag and M. Schoenauer. Controlling crossover through inductive learning. *Conference on Parallel Problems Solving From Nature*, 1994.

[96] K. Shinoda and C.-H. Lee. Structural MAP speaker adaptation using hierarchical priors. *IEEE Workshop on Speech Recognition Understanding*, 1997.

[97] K. Shinoda and C.-H. Lee. Unsupervised Adaptation using Structural Bayes Approach. *International Conference on Acoustics, Speech, and Signal Processing*, pages 793–796, 1998.

[98] K. Shinoda and C.-H. Lee. A Structural Bayes Approach to Speaker Adaptation. *Transactions on Speech and Audio Processing*, 9(3) :276–287, 2001.

[99] O. Siohan, C. Chesta, and C.-H. Lee. Joint Maximum A Posteriori Adaptation of Transformation and HMM Parameters. *Transactions on Speech and Audio Processing*, 9(4) :417–428, 2001.

[100] O. Siohan, T.A. Myrvoll, and C. H. Lee. Structural Maximum A Posteriori Linear Regression for fast HMM Adaptation. *Workshop on Automatic Speech Recognition : Challenges for the new Millenium, Paris, France*, pages 120–127, 2000.

[101] O. Siohan and A.C. Surendran. Structural Bayesian Predictive Adaptation of Hidden Markov Models. *Workshop on Adaptation Methods for Speech Recognition*, 2001.

[102] A. Spalanzani. *Algorithmes évolutionnaires pour l'étude de la robustesse des systèmes de reconnaissance automatique de la parole.* PhD thesis, Université Joseph Fourier, Grenoble, 1999.

[103] W.M. Spears and K.A. De Jong. On the Virtues of Parameterized Uniform Crossover. *International Conference on Genetic Algorithms*, pages 230–236, 1991.

[104] Y. Tsao, S.-M. Lee, F.-C. Chou, and L.-S. Lee. Segmental Eigenvoice for Rapid Speaker Adaptation. *Eurospeech*, pages 1269–1272, 2001.

[105] L.F. Uebel and P.C. Woodland. An Investigation into Vocal Tract Length Normalisation. *Eurospeech*, pages 2519–2522, 1999.

[106] J. Ueberla. *Analyzing and Improving Statistical Language Models for Speech Recognition.* PhD thesis, B. Sc. Technische Universitaet Muenchen, M. Sc. Université J. Fourier, Grenoble, France, 1994.

[107] A.J. Viterbi. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory*, 3(2), 1967.

[108] M. Woszczyna. *Fast speaker independant large vocabulary continuous speech recognition.* PhD thesis, 1998.

[109] P. Zhan, M. Westphal, M. Finke, and A. Waibel. Speaker normalisation and speaker adaptation - a combination for conversational speech recognition. *Proceedings of the European Conference on Speech Communication and Technology*, pages 2087–2090, 1997.