Traitement automatique de la parole en milieu bruité : étude de modèles connexionnistes statiques et dynamiques

Laurent Buniet

Université Henri Poincaré - Nancy 1
UFR STMIA

Centre de Recherche
en Informatique de Nancy
CNRS URA 262

École Doctorale IAE+M
DFD Informatique

INRIA Lorraine

Traitement automatique de la parole en milieu bruité : étude de modèles connexionnistes statiques et dynamiques

THÈSE

présentée et soutenue publiquement le
lundi 10 février 1997
par

Laurent BUNIET

pour l'obtention du
Doctorat de l'Université Henri Poincaré - Nancy 1
spécialité informatique

Composition du Jury :

Président :	Dominique MÉRY	
Rapporteurs :	Jean-Claude JUNQUA	DR, Panasonic Corp., STL, Santa-Barbara, ÉUA
	Dominique MÉRY	Pr., IUF, UHP & CRIN-CNRS, Nancy
	Jean-Luc SCHWARTZ	CR, ICP-INPG, Grenoble
Examinateurs :	Frédéric ALEXANDRE	CR, INRIA Lorraine, Nancy
	Dominique FOHR	CR, CRIN-CNRS, Nancy
	Jean-Marie PIERREL	Pr., UHP & CRIN-CNRS, Nancy

Je dédis cette thèse à ma mère et à la mémoire de mon père
qui m'ont tous deux permis de la réaliser
et à ma femme, Anne-Caroline, pour son amour et son soutien.

"I've seen things you people wouldn't believe,
attack ships on fire off the shoulder of Orion,
I watch C-beams glitter in the dark near the Tannhauser gate.
All those moments will be lost in time like tears in rain."

Épitaphe de Roy Batty,
in Blade Runner, film de Ridley Scott

REMERCIEMENTS

Au terme de cette thèse et en préambule au mémoire rédigé, je tiens à remercier tous les membres du jury qui ont accepté de juger le travail effectué tout au long de ces années :

Jean-Marie Pierrel, professeur de l'Université Henri Poincaré - Nancy 1, directeur du Centre de Recherche en Informatique de Nancy (CNRS UMR 262) et responsable de l'équipe Dialogue commune au CRIN et à l'INRIA Lorraine, qui m'a encadré tout au long de ces années de recherche malgré la charge qui était la sienne,

Jean-Claude Junqua, chargé de recherche chez Matsushita, au Speech Technology Laboratory de la Panasonic Corp. à Santa Barbara, en Californie, qui est en partie à l'origine de ces recherches et qui me fait aujourd'hui le plaisir de juger un travail proche du sien après avoir dû survoler le tiers de la circonférence du globe,

Jean-Luc Schwartz, chargé de recherche à l'Institut de la Communication Parlée de l'Institut National Polytechnique de Grenoble, qui m'aura permis de soutenir cette thèse à la date et à l'heure prévues en acceptant au pied levé d'être rapporteur de ce travail, merci beaucoup,

Dominique Méry, membre de l'Institut Universitaire de France, professeur de l'Université Henri Poincaré - Nancy 1 et directeur de recherches au CRIN, qui a bien voulu participer à ce jury après avoir dû me supporter comme ATER,

Frédéric Alexandre, chargé de recherche INRIA à l'INRIA Lorraine et responsable du groupe Cortex au sein de l'équipe RFIA/SYCO commune au CRIN et à l'INRIA Lorraine, avec lequel j'ai pû avoir de bons contacts au cours de cette thèse bien qu'aucune hiérarchie officielle n'ait existée,

Dominique Fohr, chargé de recherche CNRS au CRIN au sein de l'équipe RFIA/SYCO, qui aura, pendant toutes ces années, suivi mon travail de thèse au pas de charge et m'aura véritablement accordé toute sa confiance.

Je tiens également à remercier trois chercheurs du CRIN pour leurs activités et leur passion paraprofessionnelles. Je tiens ainsi à remercier Jean-Pierre Finance, président de l'université Henri Poincaré, qui préside également l'Aéro-Club de Lorraine à Lunéville dont je suis un heureux membre. Je remercie également Jacques Guyard pour m'avoir fait connaître l'ACL et, last but not least, je tiens à remercier très chaleureusement Jean-François Mari qui m'a, patiemment, très patiemment, appris à voler et m'a conduit jusqu'au TT et a toujours, ou presque, gardé son calme bien que des manœuvres aient quelques fois été exotiques. Pour continuer à suivre le fil de l'air, je tiens à remercier le Père Gilles Silvy-Leligois, aumônier de TAT-BA, de m'avoir marié à ma chère et tendre et Patrick "doudou" Doucet, pour ce qui reste, aujourd'hui encore, la frayeur de ma vie...

Ma vie dans ce laboratoire ne serait pas ce qu'elle est sans les membres de l'équipe Dialogue, de l'équipe RFIA/SYCO et de son groupe Cortex et de toutes les personnes qui fourmillent, le jour et la nuit, dans le bâtiment Loria. Qu'ils en soient tous vivement remerciés de même que, hors de ce campus, les Supélec de Metz et Rennes, Ludo et Gilles, Monseigneur Stéphane 1er et Madame, ainsi que tous les membres de l'ACTH.

Je tiens également à remercier les membres de ma famille qui m'ont soutenu tout au long de ces années et tous les membres de ma toute nouvelle belle famille pour leur amour et leur amitié. Ma vie extraprofessionnelle aurait par ailleurs été bien terne sans toutes les personnes qui m'ont offert leur amitié : Christelle, Christophe, Corinne, Emmanuel, Éric, Fabrice, Frédérique, Jean-Baptiste, Jean-Luc, Jean-Paul, Jean-Philippe, Laurence, Sophie, Stéphanie, ... et tous ceux que j'oublie. Mes amitiés à vous tous ! J'adresse également un petit clin d'œil à Loukhoum pour son sens de la vie...

Enfin, je remercie tout particulièrement les relecteurs hors du domaine et j'adresse une mention spéciale au découvreur des néologismes bunietiens...

LISTE DES FIGURES

LISTE DES TABLES

RÉSUMÉ INTRODUCTIF

Le travail réalisé lors de cette thèse s'inscrit dans le cadre général de la reconnaissance automatique de la parole, RAP. Les travaux entrepris jusqu'à présent ont permis de réaliser des systèmes qui, s'ils permettent une reconnaissance de vocabulaires de plus en plus étendus, restaient, jusqu'à ces dernières années, assez sensibles aux conditions sonores de l'environnement d'utilisation, les conditions parfaites rencontrées en laboratoire ayant longtemps masquées ces contraintes. Cette sensibilité au bruit est un des freins majeurs à l'emploi de la reconnaissance automatique de la parole dans des applications dites grand public qui supposent que l'utilisation d'un système de dialogue oral homme machine, DOHM, permette de reconnaître tout locuteur quelques puissent être les conditions spécifiques de l'environnement sonore.

Les techniques actuellement les plus employées pour augmenter la résistance des systèmes de RAP reposent sur l'emploi de méthodes statistiques permettant le calcul du spectre moyen du bruit. De par leur nature, ces méthodes sont spécifiques à des types de bruits dont la caractéristique principale est la stationnarité. Notre objectif, lors de cette thèse, est de définir un système de RAP offrant une bonne résistance au bruit sans utiliser le paradigme de spectre moyen mais se fondant sur l'exploitation d'indices robustes de la parole, permettant ainsi de traiter des bruits d'ordre plus général. Les modèles mathématiques que nous avons choisi d'employer pour atteindre notre objectif sont les réseaux connexionnistes, ou neuromimétiques, qui présentent des capacités très intéressantes en reconnaissance et en classification des formes.

La première partie de cette thèse est constituée de deux chapitres donnant un état de l'art général dans le domaine de la parole et du connexionnisme.

Le premier chapitre a pour intention de présenter les notions élémentaires et les termes relatifs à la description de la parole et de son traitement. Nous y exposerons tout d'abord les grands principes du traitement automatique de la langue avant de présenter les appareils auditif et phonatoire de l'être humain. Nous présenterons ensuite deux des taxonomies possibles pour les sons observables dans un signal de parole, l'une étant spécifique au français tandis que l'autre est spécifique à l'anglais. Nous traiterons enfin les problèmes de variabilité du signal de parole et énoncerons quelques unes des méthodes de représentation graphique du signal, qu'elles soient ou non dédiées à la parole et qu'elles soient reconnues ou non comme résistantes au bruit.

Le deuxième chapitre nous permet de présenter les trois grandes techniques de reconnaissance des formes qui sont utilisées en reconnaissance automatique de la parole : l'alignement temporel, les réseaux de Markov et les modèles connexionnistes. La présentation des modèles connexionnistes, qui sera plus approfondie, sera précédée d'une brève présentation des connaissances de la neurobiologie qui ont servi de fondement à l'établissement des techniques neuromimétiques.

La deuxième partie de cette thèse permet de présenter les travaux réalisés pour tenter de développer un système de reconnaissance de petits vocabulaires, tels que les chiffres ou les lettres épelées, prononcés de manière continue dans un environnement sonore dont les caractéristiques sont inconnues a priori.

Le chapitre 3 permettra au lecteur d'avancer dans la compréhension du problème que pose les environnements bruités en compréhension de la parole, tant par l'homme que par la machine. Nous exposerons tout d'abord le sujet de cette thèse et le cadre plus général auquel il se rapporte. Nous donnerons ensuite un bref aperçu des premiers travaux que nous avons effectués avec des méthodes d'énergie, travaux qui ont été abandonnés du fait des résultats peu probants qui ont été obtenus.

Le chapitre 4 présente le premier système nous ayant permis d'obtenir de bons résultats en reconnaissance automatique de petits vocabulaires prononcés de manière continue en milieu bruité.

Ce système est fondé sur l'utilisation séquentielle et hiérarchique de plusieurs réseaux connexionnistes statiques tels que les perceptrons multicouches ou les *Selectively Trained Neural Networks*. Trois étapes successives ont été mises en place : elles correspondent à une première segmentation du signal, puis à une étape d'identification des voyelles et, enfin, à une étape d'identification des mots. Les résultats obtenus dans ce chapitre sont intéressants puisque de bons taux de reconnaissance sont obtenus jusqu'à des rapports signal-sur-bruit de 6 décibels avec des conditions de bruits variées qui prouvent qu'il est possible de mettre en œuvre un système résistant à des environnements sonores différents et qui n'ont pas été rencontrés lors de la phase d'apprentissage, même si ces bruits ne sont pas stationnaires. Ces résultats permettent également de montrer qu'il est possible, à partir de l'exploitation d'indices robustes, de mettre en œuvre des systèmes résistants n'exploitant aucune connaissance statistique sur le spectre du bruit. Les résultats obtenus se dégradent cependant à mesure que le niveau du bruit augmente, ne permettant pas à notre système de RAP d'atteindre des performances équivalentes à celles observées chez des sujets humains. Le problème majeur que nous avons observé correspond à la fusion de différents noyaux vocaliques en un seul lors de la phase de segmentation prouvant que les informations purement phonétiques ne sont pas totalement suffisantes pour résoudre le problème posé même si elles permettent d'atteindre un niveau de résultat satisfaisant par rapport à d'autres systèmes.

Le chapitre 5 est l'occasion de poser le problème de la modélisation du temps et d'étudier la manière dont celui-ci intervient dans la représentation et le traitement de la parole, du bruit et de la musique. Nous verrons ensuite comment peuvent être appréhendés les phénomènes temporels par l'intermédiaire de systèmes capables de mémoriser et de restituer, à brève échéance, des informations passées, que ces informations soient représentées par l'intermédiaire d'automates ou de modèles permettant de simuler une décroissance progressive de l'activité.

Le chapitre 6, moins général que celui qui le précède, nous permettra de présenter un état de l'art des différents modèles connexionnistes dynamiques existants en nous attachant tout particulièrement à la présentation des modèles neuromimétiques à récurrence locale. Cette présentation permettra au lecteur de mieux comprendre le choix architectural que nous avons effectué pour modéliser la durée des phonèmes dans notre étape de segmentation, qui nous a posé des problèmes lors du chapitre 4. Ce chapitre est l'occasion pour nous de justifier le choix pour un modèle neurobiologiquement plus plausible que les simples perceptrons multicouches ou les réseaux connexionnistes à récurrence par plaque.

Le chapitre 7 présente l'architecture gamma et les extensions que nous avons jugé bon d'y apporter vis-à-vis de nos connaissances en reconnaissance automatique de la parole et du problème que nous avions à résoudre. Ce chapitre s'attachera à énoncer les caractéristiques principales du modèle gamma et ses relations avec les systèmes dynamiques non linéaires ainsi que tous les problèmes d'apprentissage qui en découlent. Nous présenterons ensuite les extensions architecturales apportées au modèle. Deux séries de tests sont enfin présentées qui permettent de juger des qualités et des capacités d'un réseau connexionniste utilisant des filtres gamma, tant pour la reconnaissance de diverses séquences temporelles que pour la segmentation de la parole selon différentes classifications. Ce chapitre présente également notre exploration du problème de l'apprentissage dans les réseaux connexionnistes à récurrence locale puisque notre incapacité à exploiter correctement le mécanisme de filtre a été la cause principale de l'obtention de résultats assez moyens en segmentation de la parole. Cette qualité assez moyenne est cependant contrebalancer par les résultats que nous avons obtenus en reconnaissance et classification de séquences temporelles abstraites. Le modèle gamma peut, en effet, obtenir de bons résultats sur des séquences qui peuvent même être bruitées sans pour autant implanter un automate de quelque manière que ce soit. Les résultats obtenus dans ces expériences particulières l'ont cependant été avec un algorithme d'apprentissage différent de celui employé lors des tâches de segmentation de la parole, cet algorithme étant inadapté au flot continu que représente un signal de parole.

Le chapitre 8 de conclusions nous permettra de donner un point de vue sur la thèse qui a été développée et sur ce que nous aurions voulu y développer si notre exploration des mécanismes et des lois d'apprentissage dans les systèmes dynamiques non linéaires avait abouti. Ainsi présenterons nous quelques modèles que nous avions envisagé d'étudier et que nous avons, finalement, laissé au domaine de l'inconnu et de l'inexploré.

La dernière partie de cette thèse est constituée de trois annexes relatives tant aux différents bruits utilisés qu'au modèle gamma. Ces trois annexes précèdent la bibliographie.

La première annexe, l'annexe A1, permettra au lecteur de comparer les équations d'apprentissages mises en œuvre pour un perceptron multicouche et celles utilisées pour l'apprentissage dans le modèle gamma, objet des derniers chapitres de cette thèse, tant pour les coefficients synaptiques que pour les coefficients de régression μ permettant d'ajuster le comportement local de mémorisation de chacune des unités connexionnistes mettant ce paradigme en œuvre.

L'annexe A2 permettra au lecteur de mieux appréhender le comportement des filtres gamma, tant lorsque ceux-ci sont observés de manière isolée que lorsqu'il forment des lignes de délais encastrés. Cette annexe permettra ainsi d'observer les comportement des filtres tels qu'ils ont été utilisés dans cette thèse, pour des valeurs de μ comprises entre 0, borne exclue, et 1, borne incluse, et où le filtre gamma se comporte comme un filtre passe-bas, que pour des valeurs de μ qui n'ont pas été retenues mais qui pourraient être intéressantes comme, par exemple, le cas où μ est égal à 0 et qui permet d'obtenir une mémoire sur un temps infini, ou pour des valeurs de μ comprises entre 1 et 2, bornes exclues, où le mécanisme gamma se comporte comme un filtre passe-haut.

Enfin, la dernière annexe, numérotée A3, permettra au lecteur d'appréhender visuellement les différents bruits qui ont été utilisés lors de cette thèse pour bruiter les signaux de parole "propres". Cette annexe présente donc tous les bruits du corpus NOISEX 92, qui sont extraits du corpus NOISE-ROM 0, comprenant à l'origine plus de types de bruits. L'annexe présente le corpus NOISEX 92 dans un ordre de non stationnarité, et donc de difficulté, croissante. Les premiers bruits présentés sont donc les bruits les plus stables tandis que les derniers sont ceux présentant le plus d'aléa. La chronologie employée peut également être vue comme présentant les différentes étapes à franchir pour réaliser un système de RAP encore inexistant qui pourra être utilisé dans une très grande majorité de conditions. Seuls les bruits stables ou ne présentant que des "microvariations" peuvent en effet être traités par les systèmes actuellement existants.

PARTIE 1

ÉTAT DE L'ART

CHAPITRE 1 : PAROLE

"For millions of years, mankind live just like the animals,
Then something happened which unleashed the power of our imagination:
We learned to talk."

Pink Floyd
Keep Talking

"Lorsque ***moi*** j'emploie un mot", répliqua Heumpty Deumpty d'un ton de voix quelque peu dédaigneux, "il signifie exactement ce qu'il me plaît qu'il signifie... ni plus, ni moins".
"La question", dit Alice, "est de savoir si vous avez le pouvoir de faire que les mots signifient autre chose que ce qu'ils veulent dire".
"La question", riposta Heumpty Deumpty, "est de savoir qui sera le maître... un point, c'est tout".

Lewis Carroll
De l'autre côté du miroir

Résumé

Ce chapitre a pour intention de présenter les notions élémentaires de la parole et de son traitement automatique. Nous y exposerons tout d'abord les grands principes du traitement automatique de la langue avant de présenter les appareils auditif et phonatoire de l'être humain. Nous présenterons ensuite deux des taxonomies possibles pour les sons observables dans un signal de parole. Nous traiterons enfin les problèmes de variabilités de ce signal et énoncerons quelques unes des méthodes de représentation graphique de celui-ci.

1.1/ Introduction

La parole est le principal moyen de communication dans toute société humaine. Son apparition peut être considérée comme concomitante à l'apparition des outils, l'homme ayant alors besoin de raisonner et de communiquer pour les façonner [leroi92]. Son abstraction par rapport à un support physique en fait un moyen de communication très simple à utiliser. L'ère industrielle a par ailleurs permis de mettre en place des moyens d'enregistrement, et donc de sauvegarde, qui permettent à la parole de se hisser au rang de l'écrit pour la conservation de la connaissance.

L'importance de la parole fait que toute interaction homme-machine devrait plus ou moins passer par elle. D'un point de vue humain, la parole permet de se dégager de toute obligation de contact physique avec la machine, libérant ainsi l'utilisateur qui peut alors effectuer d'autres tâches. Sans pour autant imposer la parole là où elle pourrait être un frein à l'interaction (il est par exemple difficile d'imaginer une application graphique où seule la parole serait utilisée), son utilisation permettrait de commencer à limiter l'emploi des claviers, tablettes graphiques et autres écrans tactiles ou gants de désignation.

Cet avenir alléchant n'est cependant pas encore à notre portée. Bien que plusieurs systèmes de reconnaissance de la parole soient aujourd'hui commercialisés par des sociétés plus ou moins spécialisées dans ce domaine, diverses études d'introduction d'interface vocale dans des applications existantes ont montré que les techniques actuellement mises en œuvre imposaient encore trop de contraintes [maugis95]. Seules des applications d'envergure limitée, restreignant le nombre des paramètres au niveau du vocabulaire ou du nombre de locuteurs, peuvent aujourd'hui aboutir à des résultats satisfaisants [souvay92].

La recherche en reconnaissance automatique de la parole, RAP, tente donc aujourd'hui de mieux comprendre le processus humain de génération et de compréhension de la parole, tant d'un point de vue mécanique par le biais de l'étude et de la modélisation des organes biologiques en charge de ces tâches, que d'un point de vue mathématique par le développement de méthodes de classification toujours plus fines et exactes.

Nous allons présenter de manière générale, dans ce chapitre, certaines méthodes utilisées pour la modélisation de la parole (paragraphe 1.7) ainsi que les différentes taxonomies des sons observables en parole (paragraphe 1.5) et les variations qui peuvent y être constatées (paragraphe 1.6). Nous allons cependant tout d'abord parler des notions qui se rattachent à l'étude des organes biologiques de production (paragraphe 1.3) et de compréhension (paragraphe 1.4) de la parole après avoir exposé les problèmes généraux qui se posent en traitement de la langue (paragraphe 1.2).

1.2/ Le traitement automatique de la langue

1.2.1/ Les règles de la langue

La parole est le support le plus courant de la langue : il est plus facile de parler à quelqu'un que de lui écrire ou de lui fait un schéma. Mais, au delà du mode de transmission de l'information, les bases sont les mêmes. Le message est structuré selon des règles reconnues par toute personne partageant la même culture au sein d'une même société. Mais cette culture évolue au rythme de la société et de ses progrès techniques et scientifiques. L'actuelle grande facilité de communication a permis d'imposer un même langage sur des étendues géographiques de plus en plus importantes. Cependant, une plus grande facilité de déplacement permet à une langue d'évoluer de plus en plus vite, au rythme de ses confrontations à des langues étrangères. Cette évolution est encore accentuée par les tendances actuelles du langage commercial et publicitaire, peu préoccupé par les règles de grammaire et pressé de faire passer son message.

Le traitement automatique de la langue suppose donc d'analyser les structures tout en suivant l'évolution. Il est bien sûr tentant de vouloir figer, ou contrôler, l'évolution d'une langue par la création d'instituts tels que l'Académie Française ou par la Loi [miptce94]. Mais cet endiguement est bien souvent vain et le contrôle se transforme, la plupart du temps, en avalisation a posteriori du changement. Il convient donc d'analyser la langue à des niveaux suffisamment abstraits pour que les différentes évolutions ne soient plus que des phénomènes aux conséquences limitées.

L'étude du traitement automatique de la langue est donc structurée en différentes spécialités interdépendantes étudiant le langage suivant des axes bien définis. Au rang de ces spécialités se trouve par exemple la lexicographie qui étudie la manière selon laquelle un mot est défini à partir d'autres mots, ce processus pouvant aboutir à des liens auto-référentiels. L'étude de la syntaxe essaie, elle, de découvrir quelles peuvent être les structures possibles d'une phrase et quels sont les changements qui peuvent intervenir dans une suite de mots sans que le sens en soit modifié. Ce sens est étudié par la sémantique et la pragmatique, ces deux disciplines intervenant à des niveaux influencés de manière variable par la culture qui est associée à la langue étudiée.

Le problème principal du traitement automatique de la langue est de définir un coefficient de confiance sur la compréhension du message. La mise en place d'une interface vocale impose en effet d'être sûr du sens du message avant toute réaction du système. La compréhension du message n'est

pour l'instant assurée que pour les langages de commandes restreints pour lesquels cette compréhension ne résulte pas d'un processus automatique mais de la simple association entre un mot, ou une phrase de quelques mots, et l'action qui doit résulter de sa prononciation. Dans de tels langages, c'est le mot, et non son sens, qui détermine le traitement à effectuer.

Ce problème de la compréhension est très loin d'être trivial et pourrait même être considéré comme un graal. Bien que ce problème puisse être facilement sous-estimé, la compréhension d'un message entre deux acteurs humains d'une conversation n'est pas aussi évidente qu'il y paraît. Comment, en effet, être sûr qu'un message a été totalement compris par l'auditeur et que l'idée ayant entraîné la génération du message chez le locuteur a été correctement décodée ? Les différents notions et sens attachés à un mot particulier peuvent en effet varier d'une personne à l'autre et un message peut avoir une force et une acception différentes chez les deux intervenants d'une conversation. Nous avons résumé le processus de communication et montré la possible différence sémantique pouvant exister dans la compréhension d'un message dans la figure 1.1.

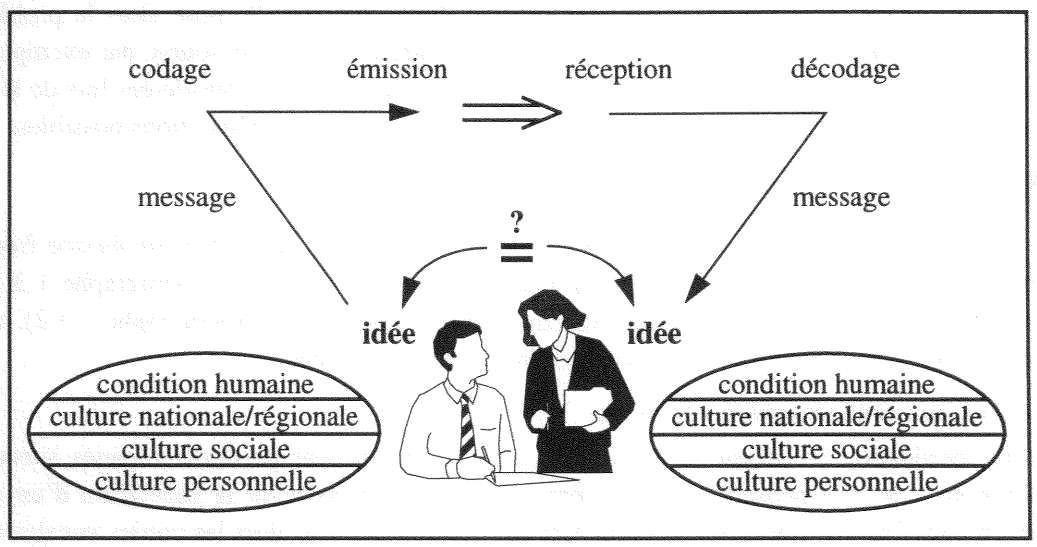


Figure 1.1 : Exemple de dialogue personne-personne.

Après avoir présenté quelques-uns des problèmes que peut poser une langue, vivante ou morte, nous allons maintenant voir quels peuvent être les problèmes posés par le dialogue homme-machine.

1.2.2/ Le dialogue homme-machine

Le dialogue homme-machine essaie de mettre en place un traitement automatique de la langue qui puisse servir d'interface entre la machine, ou une application, et l'homme. Cette mise en place, qui reste limitée dans ses ambitions pour les raisons que nous venons d'exposer, impose de définir plusieurs processus concourant à la compréhension, même restreinte, du dialogue ou, plus simplement, des commandes.

Les applications actuelles de dialogue oral, même si elles utilisent la parole à la manière d'un bouton poussoir pour le déclenchement d'actions, doivent mettre en œuvre des techniques facilitant leur utilisation et les rendant moins rébarbatives. Il est en effet inutile de demander à un utilisateur de répéter toutes les caractéristiques d'une demande si certaines de ces caractéristiques sont contextuelles et peuvent être retrouvées dans une partie du dialogue précédent ou dans l'état courant de l'application. Des mécanismes langagiers tels que l'ellipse ou l'anaphore obligent donc à mettre en place une gestion de l'historique du dialogue pour retrouver des informations utiles à l'application.

Il est cependant difficile de dépasser le stade du langage de commande pour des interactions allant de l'homme vers la machine. Le passage d'informations de la machine vers l'homme est plus simple, que l'information soit fournie à l'homme par l'intermédiaire de la parole ou du texte. La synthèse de

message en langue naturelle est en effet plus simple à réaliser car le concepteur peut utiliser des formes prédéfinies qui suivent des règles strictes, réduisant ainsi le champ des possibles. Le sens de communication inverse, de l'homme vers la machine, est lui beaucoup plus problématique puisqu'il est impossible de demander à un utilisateur quelconque de suivre une grammaire formelle en utilisant un vocabulaire prédéfini, marque d'un langage de commande évolué. Le langage totalement naturel permet d'exprimer une même idée sous plusieurs formes, ces formes pouvant avoir une influence sur la suite du dialogue à moyen ou long terme. L'ensemble des possibles, et donc le facteur de branchement, est alors énorme et pourrait très bien submerger une application par un grand nombre d'analyses sémantiques possibles, lorsque ces analyses sont réalisables.

Le dialogue homme machine doit également être étudié du point de vue de l'efficacité. Il est tentant de vouloir mettre en place des applications n'utilisant que la parole comme interface. La parole peut cependant ne pas être le moyen le plus efficace de communiquer. Il est bien sûr le seul possible pour des applications faisant intervenir un terminal téléphonique mais est concurrencé par des moyens plus efficaces dans des applications plus techniques. Se pose alors le problème de la gestion du canal de communication, un canal de désignation tel que la souris, par exemple, pouvant être utilisé en complément de la parole. Ces possibilités doivent être considérées lors de la définition de l'application et supposent de définir, a priori, les interactions et imbrications possibles.

1.3/ L'appareil phonatoire

L'appareil phonatoire nous permet de produire des sons très variés dans un espace fréquentiel et énergétique pourtant limité (figure 1.5). L'appareil phonatoire humain (paragraphe 1.3.1) a été la base de recherches visant à simuler mécaniquement ses capacités (paragraphe 1.3.2), recherches ayant permis, en retour, de mieux comprendre son fonctionnement.

1.3.1/ L'appareil phonatoire humain

La production de la parole est assurée, chez l'homme, par plusieurs organes successifs. Les poumons sont indispensables dans ce processus puisqu'ils assurent la génération d'un composant incontournable : de l'air sous pression. Cet air, expulsé, traverse alors les cordes vocales qui entrent ou non en action pour produire un voisement. Ce voisement correspond à la fréquence fondamentale qui est le timbre de la voix.

Cette fréquence fondamentale étant produite, elle est propagée dans l'ensemble du conduit vocal. Ce conduit est de forme et de volume variable. Plusieurs organes concourent à ces possibles modifications qui permettent de produire des sons différents. Parmi ces organes se trouve la langue, acteur principal des modifications qui peut agir par constriction ou occlusion du conduit vocal. Les dents et les lèvres agissent également par occlusion ou constriction, à des degrés cependant moindres. Le conduit vocal est, la plupart du temps, constitué du seul conduit buccal. La luette et son prolongement vers le palais, le vélum, assurent normalement la fermeture du conduit nasal pendant la production de parole. Le conduit nasal peut, dans certains cas, être connecté au conduit vocal. Cette connexion permet de générer des sons supplémentaires en modifiant le volume de la caisse de résonance normalement constituée par le seul conduit buccal. Une coupe de l'appareil phonatoire humain est fourni en figure 1.2.

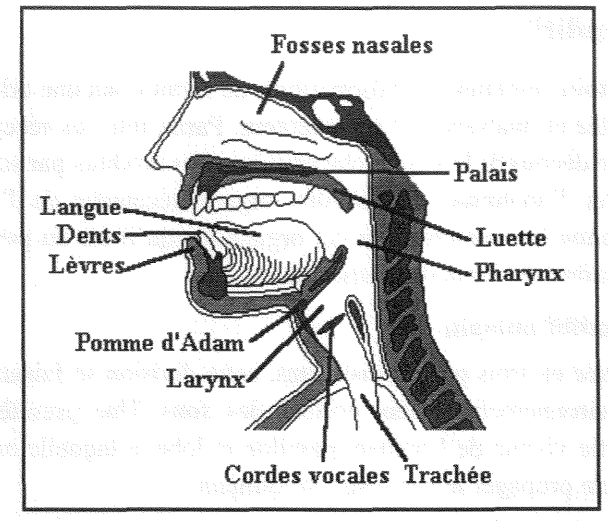


Figure 1.2 : Coupe de l'appareil phonatoire humain (d'après [mella93]).

Les différents organes de la parole et leur agencement peuvent servir de base à des modélisations du conduit vocal

1.3.2/ Modèles articulatoires

Pour mieux comprendre le fonctionnement de l'appareil phonatoire, il semble judicieux d'essayer de simuler physiquement cet organe faisant intervenir la mécanique et la dynamique des fluides. De telles études partent du principe que la parole est un ensemble de mouvements rendus audibles plutôt qu'un ensemble de sons produits par du mouvements [abry95]. Ce type de modélisation peut aider à comprendre les raisons des variations qui existent lors de la production de sons dont nous parlerons plus avant dans ce chapitre. Plusieurs modélisations articulatoires, plus ou moins simplificatrices, ont été réalisées. Certaines d'entre elles sont présentées dans [calliope89].

Un des modèles les plus connus est le modèle de Maeda [maeda79] qui caractérise le conduit vocal grâce à un ensemble de mesures réalisées sur des images radiographiques par le biais d'une grille semi-polaire. Ce modèle n'est pas à proprement parler articulatoire mais est plus simplement descriptif. Un ensemble de mesures détermine cependant une forme de conduit et permet donc de prévoir le son qui est y est associé. Certaines études vont actuellement dans le sens d'une exploration fonctionnelle du modèle de Maeda.

1.3.3/ Systèmes de synthèse de parole

La synthèse de la parole, que nous ne considérons ici qu'à son plus bas niveau et sans aucun regard sur l'étape de génération de texte, peut fort bien être réalisé à partir d'un simple système d'équations. Cette génération de parole se fait à partir d'un ensemble de paramètres déterminés à partir d'un extrait de phrase à un instant bien précis. Cet ensemble de paramètres peut comprendre des informations prosodiques [calliope89]. Une réalisation intéressante, le système *NetTalk*, a obtenu des résultats moyens avec un système connexionniste cependant très simple, montrant ainsi la relative facilité que présente la réalisation de cette tâche [sejnowski87]. La synthèse de la parole peut également être appréhendée sous l'angle des modèles articulatoires, cette approche permettant de prendre en compte les phénomènes articulatoires de manière plus réaliste et, ainsi, d'obtenir une parole de synthèse de meilleure qualité. Il est intéressant de noter que certaines recherches [robert-ribes95] menées dans le cadre de la synthèse de la parole ont montré l'intérêt d'un système de synthèse audio-visuel, le son étant accompagné par la représentation des lèvres. Un tel système possède en effet une assez bonne résistance à un bruit ambiant très fort grâce à l'accompagnement de l'image des lèvres.

1.4/ L'appareil auditif

L'appareil phonatoire, émetteur d'informations, ne serait d'aucune utilité si l'information générée ne pouvait être captée et analysée par un récepteur. Parmi tous les récepteurs existants, l'homme a acquis la capacité de découvrir le sens caché sous les sons produits par son interlocuteur. Nous allons maintenant présenter l'anatomie de l'oreille, organe récepteur de l'information sonore, et les capacités de perception qui caractérisent cet organe lorsqu'il est en parfait état et n'a subi aucune atteinte venue amoindrir ses capacités intrinsèques.

1.4.1/ L'appareil auditif humain

L'oreille est divisée en trois parties distinctes, cette division se faisant en fonction de la distance par rapport à l'environnement aérien, porteur des sons. Une première partie, l'oreille externe, correspond à la partie visible de l'organe, pavillon et lobe, à laquelle est rattaché le conduit auditif externe qui permet de propager le son jusqu'au tympan.

Le tympan marque la frontière entre l'oreille externe et l'oreille moyenne. Les organes de l'oreille moyenne permettent de transformer les sons en vibrations grâce au contact qu'ils ont avec le tympan. Ces vibrations, une fois générées, sont transmises à la cochlée qui constitue l'organe majeur de l'oreille interne. La cochlée permet de transformer les vibrations en influx nerveux par le biais de cellules ciliées qui captent les vibrations produites dans le fluide de la membrane basilaire par l'étrier, le dernier os de l'oreille moyenne. Cet influx nerveux est alors transmis au cerveau en charge du traitement.

Une description détaillée de l'oreille (figure 1.3) permettra au lecteur de mieux appréhender les différents organes la constituant et de mieux visualiser leur répartition.

Il faut noter que la présence de deux oreilles permet d'effectuer, au niveau du cerveau, des traitements plus complexes que le simple décodage d'une scène auditive. Le positionnement des oreilles de chaque côté du crâne permet en effet de profiter des capacités de la binauralité. Cette faculté permet de calculer la provenance d'un son en fonction du retard d'arrivée de ce son dans une oreille par rapport à l'autre. Il est à noter que cette binauralité permet à l'homme de discerner la position horizontale de l'émetteur d'un son mais pas sa position verticale. Ce principe de binauralité a été généralisé par certaines espèces animales de manière à distinguer la provenance d'un son dans un espace tridimensionnel et non plus seulement bidimensionnel, cette généralisation pouvant être effectuée par simple désaxialisation d'une oreille par rapport à l'autre, de chaque côté du crâne.

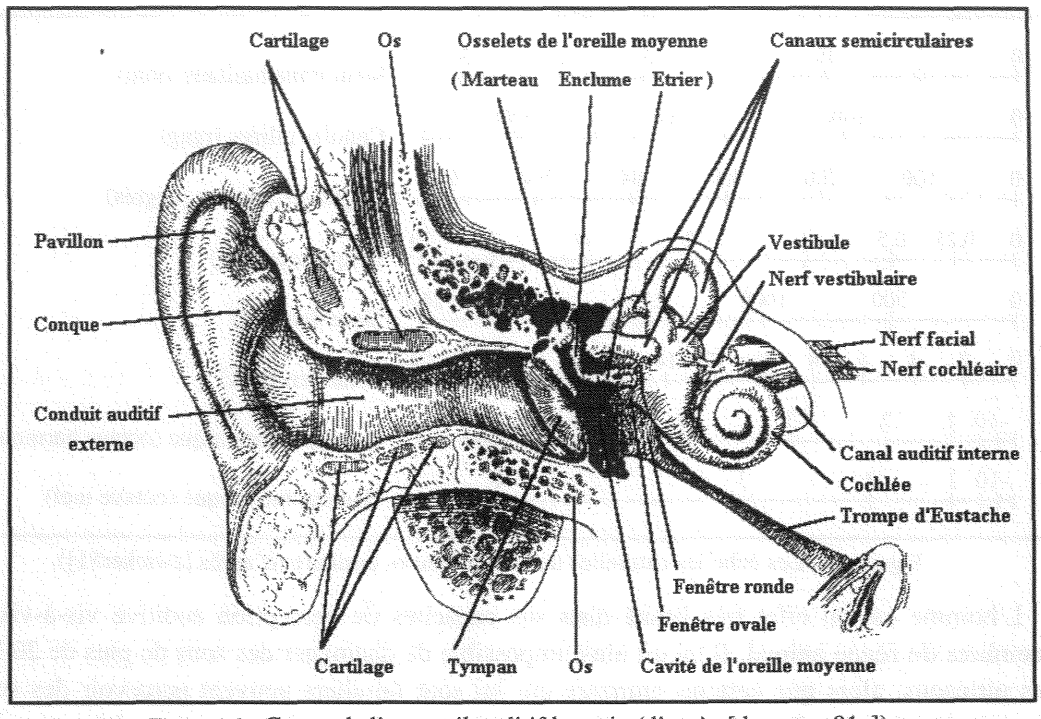


Figure 1.3 : Coupe de l'appareil auditif humain (d'après [ducassou91a]).

L'oreille réagit à des sons de diverses fréquences qui peuvent être regroupées sur des échelles linéaires ou non linéaires.

1.4.2/ Courbes psycho-acoustiques

Plusieurs échelles essaient de rendre compte de la réalité perceptive de l'oreille. Elles peuvent toutes être rapprochées des échelles de la membrane basilaire et du rang des cellules ciliées comme la montre la figure 1.4. Ces échelles ne présentent pas toutes la même morphologie. En effet, celles qui essaient de restituer le plus correctement possible les échelles de la perception humaine sont non linéaires, telles que les échelles Mel ou Bark. Les échelles qui peuvent être qualifiées de plus mathématiques sont en revanche linéaires, telle que l'échelle des fréquences.

Ces différentes échelles essaient de rendre compte du mode de perception de l'homme en permettant de distinguer les plages de plus ou moins grande importance. Ainsi les basses fréquences sont-elles perçues de manière plus fine par l'homme que les hautes fréquences. Cette différence dans la finesse de perception permet de comprendre plus facilement certaines courbes, en particulier les courbes situant l'utilisation du spectre sonore par l'homme.

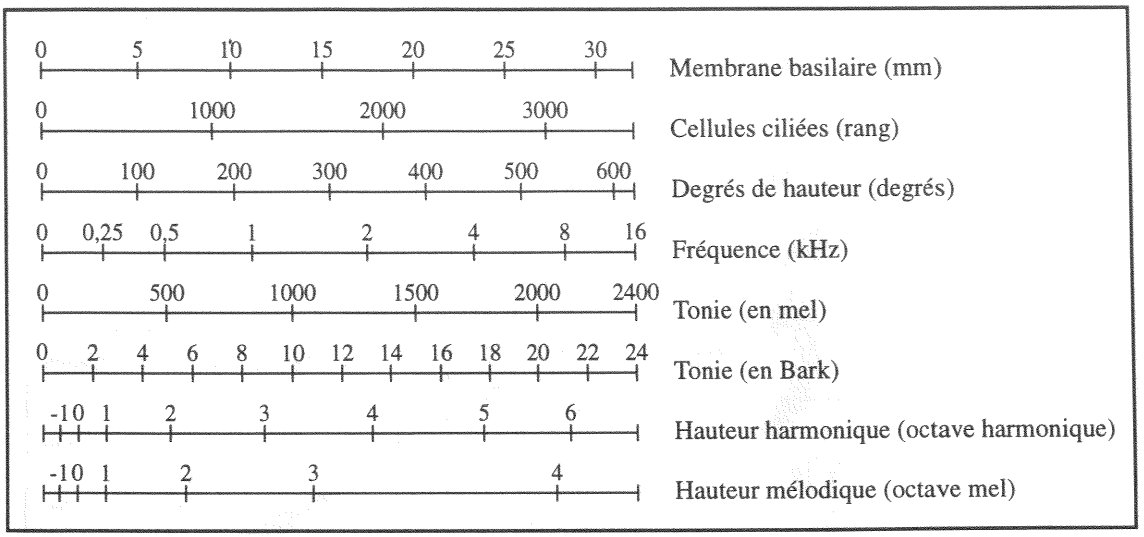


Figure 1.4 : Les échelles naturelles de la membranes basilaire (d'après [zwicker81]).

L'homme est en effet très limité dans ses capacités de perception auditive vis-à-vis d'autres membres du règne animal. Il lui est ainsi impossible de distinguer des sons de plus de 20 kilohertz, les ultrasons, alors que certains animaux qui lui sont familiers peuvent percevoir des sons allant jusqu'à 50 kilohertz. De même lui est-il impossible de distinguer des sons d'une fréquence inférieure à 20-25 hertz, les infrasons. À l'intérieur de cet espace fréquentiel existe un sous-espace délimité par les niveaux d'énergie des sons. Il existe une limite d'énergie en deçà de laquelle l'homme ne percevra pas un son d'une fréquence appartenant pourtant au spectre de l'audition. Cette limite d'énergie est appelée seuil d'audition et il est variable en fonction de la fréquence. Inversement, il existe une limite d'énergie maximale. Cette limite ne doit pas être franchie car la cochlée, et plus particulièrement les cellules ciliées, peuvent être irrémédiablement endommagées. Cette limite s'appelle le seuil de douleur et elle aussi est variable en fonction de la fréquence. Il est intéressant de noter qu'il existe dans l'oreille deux muscles qui permettent à l'homme de débrayer le transfert des vibrations du tympan à la cochlée pour limiter les dégradations qui peuvent survenir dans le cas où un bruit dépassant le seuil de douleur est perçu.

L'espace de fréquences et d'énergies ainsi défini (figure 1.5) constitue la zone d'audition à l'intérieur de laquelle l'homme peut recevoir des informations de son environnement. C'est bien sûr à l'intérieur de cet espace que se trouve le champ de la musique qui circonscrit lui-même le champ de la parole.

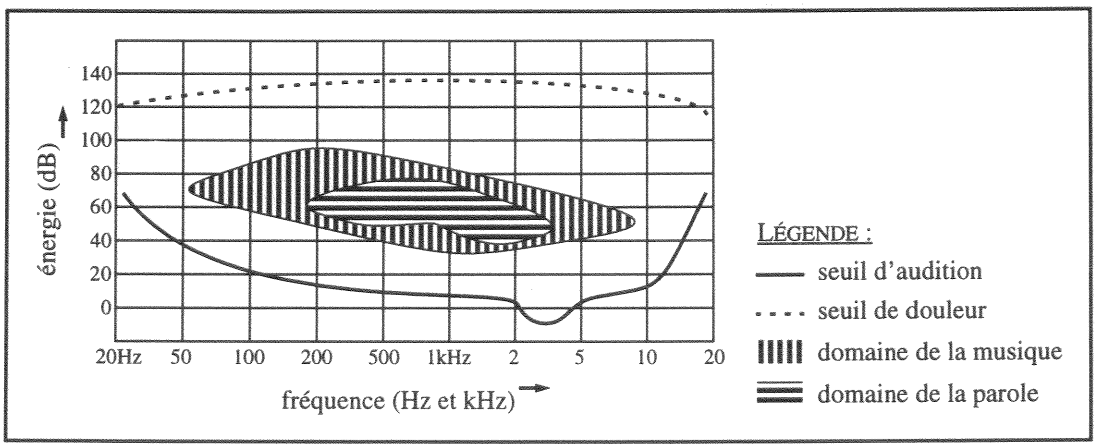


Figure 1.5 : L'aire d'audition (d'après [zwicker81]).

Après avoir énoncé les caractéristiques des organes de génération et de réception de la parole, nous allons maintenant rapidement étudier les caractéristiques du traitement de la langue et les

théories qui ont été développées pour tenter de l'expliquer.

1.5/ Taxonomie des sons

La taxonomie des sons est définie de deux manières, grâce à la phonétique et à la phonologie. Alors que la phonétique peut être considérée comme véritablement descriptive, associant chaque son de la langue à un symbole et à une classe, la phonologie s'intéresse, elle, à la description des interdépendances entre sons et au codage effectif des mots du langage lors du processus d'oralisation. La phonologie essaie donc plus particulièrement d'expliquer les différences qui peuvent exister entre la transcription phonétique d'un mot du langage et la transcription phonétique exacte du mot qui est effectivement prononcé. Il existe plusieurs phonologies, essayant de décrire les phénomènes à partir de règles générales. Notre intérêt dans ce paragraphe ne sera pas de voir quelles peuvent être ces différences mais de donner une vision simplifiée de la classification des sons.

1.5.1/ Phonétique

Les sons produits par le système phonatoire humain peuvent être rattachés à différentes classes. Ces classes permettent de regrouper les sons selon leurs principales caractéristiques qui sont facilement identifiables (paragraphe 1.5.1.1). À l'intérieur de ces classes sont regroupés des sons dont les dissimilarités peuvent être faibles.

La subdivision des sons en éléments de granularités variables et la division de l'ensemble de ces sons, ou phonèmes, en classes distinctes, est à l'origine de la constitution d'alphabets phonétiques qui caractérisent des langues différentes, chaque communauté linguistique n'utilisant pas l'ensemble des capacités de son appareil phonatoire pour générer la parole support de la communication. Nous présenterons ainsi (paragraphe 1.5.1.2) deux alphabets phonatoires qui présentent des différences bien que la majorité des sons soient communs aux deux.

1.5.1.1/ Classes phonétiques

Les différents sons de la parole sont regroupés en classes phonétiques en fonction de leurs caractéristiques principales. Ces caractéristiques représentent des différences qui sont suffisament importantes pour qu'il soit possible de classer les différents sons visibles sur un spectrogramme selon leur classe respective en très peu de temps et sans aucune écoute de la phrase correspondante. Le travail des phonéticiens est à ce titre très intéressant et parfois fort impressionnant [lonchamp90], [lonchamp91a], [lonchamp91b].

Les différentes classes phonétiques existantes, dont nous donnons ci-après la liste, correspondent à des regroupements qui suivent, dans les grands principes, les catégories de l'alphabet. Il existe ici aussi une différence entre voyelles et consonnes par exemple. Mais l'étude des sons de la parole a obligé à nuancer cette répartition et à créer d'autres classes subdivisant l'ensemble des consonnes. Les différentes classes phonétiques présentes en français et en anglais sont :

- les voyelles : cette classe correspond, à quelques nuances supplémentaires près, aux voyelles de l'écrit. Elles se caractérisent principalement par le voisement qui crée des formants. Ces formants, qui sont des zones fréquentielles de forte énergie, correspondent à une résonance dans le conduit vocal de la fréquence fondamentale produite par les cordes vocales. Ces formants peuvent s'élever jusqu'à des fréquences de 5 kHz mais ce sont principalement les formants en basses fréquences qui caractérisent les voyelles. Cette caractéristique permet d'ailleurs de distinguer grossièrement les voyelles en fonction de leur premier et deuxième formant.
- les occlusives : les phonèmes de cette classe se caractérisent oralement par la fermeture du conduit vocal, fermeture précédant un brusque relâchement. Les occlusives sont donc constituées de deux parties successives : une première partie de silence, correspondant à l'occlusion effective, et une deuxième partie d'explosion, au moment du relâchement. Les

occlusives peuvent être voisées, à la manière des voyelles, ou sourdes, c'est à dire non voisées. Les occlusives voisées peuvent également être appelées occlusives sonores.

- les fricatives : dans cette classe sont regroupés les sons produits par la friction de l'air dans le conduit vocal lorsque celui-ci est rétréci au niveau des lèvres, des dents ou de la langue. Cette friction produit un bruit de hautes fréquences et peut être voisée ou sourde.
- les sonantes : cette classe est en fait constituée, pour simplification, du regroupement des trois sous-classes que sont les semi-consonnes, les liquides et les nasales.
- Les semi-consonnes (ou semi-voyelles ou glissantes) : elles ont la structure acoustique des voyelles mais ne peuvent en jouer le rôle car elles ne sont que des transitions vers d'autres voyelles qui sont les véritables noyaux syllabiques. D'un point de vue syntaxique, une règle stricte de la langue française veut que deux voyelles ne puissent jamais se suivre. Cette règle est très largement respectée dans la construction des mots mais présente, comme toute règle, quelques exceptions. La classe des semi-consonnes a été créée pour pallier ces exceptions de manière gracieuse. Les semi-consonnes sont évidement sonores.
- les liquides : Les liquides sont très similaires aux voyelles et aux semi-consonnes mais leur durée et leur énergie sont généralement plus faibles. Elles sont sonores.
- les nasales : les phonèmes sont formés par passage de l'air dans le conduit vocal depuis les cordes vocales. Ce passage exclut normalement toute connexion du conduit normal, le conduit buccal, avec le conduit nasal. Ce dernier peut cependant être employé, dans un nombre limité de cas puisque sa physionomie ne permet pas de créer des sons autrement qu'en modifiant le volume de la caisse de résonnance qu'il constitue par l'intermédiaire de la langue, faisant occlusion dans le conduit buccal. Les nasales sont donc produites de la même manière que les occlusives nasales mais l'air n'est pas, cette fois, comprimé dans le conduit vocal. Le vélum est en effet abaissé pour permettre à l'air d'être expiré. Les nasales sont voisées. Il est à noter que certaines voyelles possèdent également un caractère de nasalité.
- les diphtongues : cette classe phonétique est propre à l'anglo-américain. Les phonèmes qui composent cette classe se caractérisent par deux états stables formantiques et par la transition entre ces deux états.
- Les affriquées : cette classe est, elle aussi, propre à l'anglo-américain mais les affriquées peuvent également être observées dans le français québécois. Les affriquées sont composées d'un occlusive immédiatement suivie par une fricative de durée cependant plus faible que celle des véritables fricatives.

Toutes ces classes peuvent se retrouver dans les différentes classifications phonétiques existantes.

1.5.1.2/ Classifications phonétiques existantes

La classification phonétique la plus répandue est l'Alphabet Phonétique International, également connu sous l'abréviation API. Cet alphabet, dans son souci d'exhaustivité, regroupe peu ou prou tous les phonèmes existants dans les diverses langues humaines. Cet alphabet, que nous présentons à la table 1.1, est très répandu. Il est en particulier employé dans presque tous les dictionnaires bi ou multilingues.

L'API se caractérise scripturalement par l'emploi de caractères tout à fait particuliers qui permet de ne pas les confondre avec les caractères de la langue écrite. Cette particularité scripturale et le peu d'espace libre laissé dans le code ASCII [cerf69] utilisé pour la représentation informatique des lettres de l'alphabet ont favorisé la création d'autres alphabets phonétiques utilisant comme base un codage en ASCII. Si cette solution impose l'emploi de plus d'un caractère par phonème, elle permet cependant de faciliter l'implantation informatique de tels alphabets. Une réalisation de l'API a donc été réalisée dans ce sens. Un autre alphabet, utilisant les même principes, a été défini dans le cadre d'un projet de recherche militaire américain. Cet alphabet, l'ARPABET, tire son nom de l'*Advance Research Program Agency* et du mot alph*abet*. Il se caractérise par l'emploi des seuls caractères de

l'alphabet en majuscules du code ASCII. Il permet principalement de représenter les phonèmes de l'anglo-américain. Nous le présentons en table 1.3.

Les différentes classifications existantes ne permettent cependant pas de donner la pleine mesure du signal de parole. Il existe en effet une différence entre la représentation presque formelle d'un son et son existence effective. Cette distance relative est étudiée par la phonologie. Elle pose, à sa manière, le problème général du lien pouvant exister entre les symboles et leurs réalisations physiques [harnad90], connu sous le nom du problème de l'ancrage. La parole fait en effet preuve d'une grande variabilité comme nous allons le voir maintenant.

symbole phonétique	exemple en langue française	classe	phonétique
a	plat		voyelles
a	mât		
i	pile		
y	rue		
ɔ	bol		
o	pôt		
ə	le		
ɛ	lait		
e	blé		
ø	peu		
œ	heure		
u	roue		
ã	blanc		voyelles nasales
õ	bon		
ɛ̃	lin		
œ̃	brun		
j	hier		semi-consonnes
ɥ	huit		
w	oui		
l	lent		liquides
R	rue		
m	masse		nasales
n	nous		
ɲ	signal		
f	fer	sourdes	fricatives
s	assis		
ʃ	chou		
v	verre	sonores	
z	Asie		
ʒ	joue		
p	passe	sourdes	occlusives
t	toux		
k	cou		
b	basse	sonores	
d	doux		
g	goût		

Table 1.1 : Alphabet Phonétique International (API). La liste des symboles est restreinte aux phonèmes du français.

phonème API	phonème ARPABET	exemple en langue anglaise	classe phonétique
i	IY	b**ea**t	voyelles
I	IH	b**i**t	
ɛ	EH	b**e**t	
æ	AE	b**a**t	
a	AA	b**o**b	
ɔ	AO	b**ough**t	
ʊ	UH	b**oo**k	
u	UW	b**oo**t	
ʌ	AH	b**u**t	
ɜ	ER	b**ir**d	
ə	UR	neighb**our**	
aʊ	AX	**a**bout	
χ	IX	rose**s**	
α^{y}	AY	m**y**	diphtongues
$ɔ^{y}$	OY	b**oy**	
e^{y}	EY	b**ai**t	
o^{w}	OW	b**oa**t	
α^{w}	AW	d**ow**n	
j	Y	**y**ou	semi-voyelles
w	W	**w**it	
l	L	**l**et	liquides
r	R	**r**ent	
m	M	**m**et	nasales
n	N	**n**et	
ŋ	NX	ba**ng**	
h	HH	**h**at	fricatives
f	F	**f**at	
θ	TH	**th**in	
s	S	**s**at	
š	SH	**sh**ut	
v	V	**v**at	
ð	DH	**th**at	
z	Z	**z**oo	
ž	ZH	a**z**ure	
č	CH	**ch**urch	affriquées
ǰ	JH	**j**udge	
p	P	**p**et	occlusives
t	T	**t**en	
k	K	**k**it	
b	B	**b**et	
d	D	**d**en	
g	G	**g**et	

Table 1.2 : Correspondance entre l'API et l'ARPABET, liste restreinte aux phonèmes de l'anglais.

Symbole phonétique	Exemple en langue anglaise	Transcription	Classe phonétique
b	bay	BCL B ey	occlusives
d	day	DCL D ey	
g	gay	GCL G ey	
p	pea	PCL P iy	
t	tea	TCL T iy	
k	key	KCL K iy	
dx	muddy, dirty	m ah DX iy, dcl d er DX iy	
q	bat	bcl b ae Q	
jh	joke	DCL JH ow kcl k	affriquées
ch	choke	TCL CH ow kcl k	
s	sea	S iy	fricatives
sh	she	SH iy	
z	zone	Z ow n	
zh	azure	ae ZH er	
f	fin	F ih n	
th	thin	TH ih n	
v	van	V ae n	
dh	then	DH e n	
m	mom	M aa M	nasales
n	noon	N uw N	
ng	sing	s ih NG	
em	bottom	b aa tcl t EM	
en	button	b ah q EN	
eng	washington	w aa sh ENG tcl t ax n	
nx	winner	w ih NX axr	
l	lay	L ey	liquides et semi-consonnes
r	ray	R ey	
w	way	W ey	
y	yacht	Y aa tcl t	
hh	hay	HH ey	
hv	ahead	ax HV eh dcl d	
el	bottle	bcl b aa tcl t EL	
iy	beet	bcl b IY tcl t	voyelles
ih	bit	bcl b IH tcl t	
eh	bet	bcl b EH tcl t	
ey	bait	bcl b EY tcl t	
ae	bat	bcl b AE tcl t	
aa	bott	bcl b AA tcl t	
aw	bout	bcl b AW tcl t	
ay	bite	bcl b AY tcl t	
ah	but	bcl b AH tcl t	
ao	bought	bcl b AO tcl t	
oy	boy	bcl b OY	
ow	boat	bcl b OW tcl t	
uh	book	bcl b UH kcl k	
uw	boot	bcl b UW tcl t	
ux	toot	tcl t UX tcl t	
er	bird	bcl b ER dcl d	
ax	about	AX bcl b aw tcl t	
ix	debit	dcl d eh bcl b IX tcl t	
axr	butter	bcl b ah dx AXR	
ax-h	suspect	s AX-H s pcl p eh kcl k tcl t	

Table 1.3 : Définition et extensions de l'ARPABET dans TIMIT (d'après [timitphon90]), liste restreinte aux phonèmes de l'anglais.

1.6/ Les problèmes de variabilité de la parole

1.6.1/ Introduction

La parole est un phénomène a priori très simple à comprendre. Tout un chacun n'est-il après tout pas capable de suivre une conversation ? Mais l'homme peut rencontrer des difficultés lorsqu'il essaie de suivre une conversation dans une langue autre que sa langue maternelle, même s'il la connaît bien. Et que dire, et que comprendre surtout, lorsqu'il essaie de suivre une conversation dans une langue qui lui est inconnue ! Ce dernier cas est pourtant le plus proche du problème posé en reconnaissance automatique de la parole, la machine n'ayant aucune connaissance propre en compréhension de la parole. Tout système de RAP doit donc être défini par l'homme lui-même, qui doit ainsi découvrir son propre processus de compréhension de la langue, processus qu'il a développé inconsciemment au cours de ses plus jeunes années.

Cet apprentissage inconscient a été la cause d'une certaine naïveté lors des premières années de recherche en RAP. Depuis lors, la liste des différentes tâches qu'il faudra résoudre s'est précisée mais n'est peut-être pas encore exhaustive. Au rang des difficultés rencontrées se trouvent les problèmes de variabilité.

Le terme de variabilité, qui est assez générique, peut englober plusieurs problèmes qui sont cependant totalement indépendants du point de vue des techniques actuellement utilisées pour les résoudre. Il est ainsi possible d'isoler une variabilité du signal de parole relativement aux classes phonétiques définies (cf. tables de ce chapitre). Il est aussi possible d'isoler la variabilité de l'environnement sonore d'un système de reconnaissance. À un niveau beaucoup plus abstrait, celui de la sémantique, il est également possible de parler de variabilité, certaines phrases ne pouvant pas être comprises lorsqu'elles sont considérées hors contexte, imposant ainsi de définir des mécanismes de gestion de l'historique du dialogue.

Nous allons maintenant voir les problèmes directement liés à la parole. Ceux-ci sont relatifs à la différence innée de prononciation vis-à-vis de un ou plusieurs locuteurs.

1.6.2/ Variabilité intra-locuteur

La variabilité intra-locuteur identifie les différences dans le signal produit par une même personne. Cette variation peut résulter de l'état physique ou moral du locuteur. Une maladie des voies respiratoires peut ainsi dégrader la qualité du signal de parole de manière à ce que celui-ci devienne totalement incompréhensible, même pour un être humain. L'humeur ou l'émotion du locuteur peut également influencer son rythme d'élocution, son intonation ou sa phraséologie.

Il existe un autre type de variabilité intra-locuteur lié à la phase de production de parole ou de préparation à la production de parole. Cette variation est due aux phénomènes de coarticulation [zerling79]. Il est possible de voir la phase de production de la parole comme un compromis entre une minimisation de l'énergie consommée pour produire des sons et une maximisation des scores d'atteinte des cibles que sont les phonèmes tels qu'ils sont théoriquement définis par la phonétique. Un locuteur adoptera donc un compromis qui est généralement partagé par une vaste majorité de la communauté de langage à laquelle il appartient bien que ce compromis lui soit propre du fait de sa physionomie particulière. Ce compromis peut d'ailleurs être retrouvé à un plus haut niveau avec la notion d'idiolecte. Ce locuteur essaiera, lors d'une phase de production de parole, d'atteindre les buts qui lui sont fixés par les différents éléments de sa phrase tout en conservant un rythme naturel de production de la parole. Les cibles peuvent alors être modifiées du fait d'un certain contexte phonétique. Ce contexte peut être antérieur, lorsque le phonème provoquant une modification se trouve avant le phonème considéré, ou postérieur lorsque le phonème perturbateur se trouve après. La coarticulation peut enfin se produire à l'échelle d'un ou de plusieurs phonèmes adjacents, ce dernier cas étant cependant très rare.

La variabilité intra-locuteur est cependant beaucoup plus limitée que la variabilité inter-locuteur

que nous allons étudier maintenant. Il est en effet possible, malgré les problèmes énoncés ci-avant, de mettre en œuvre des systèmes automatiques d'identification du locuteur, à la manière d'une personne reconnaissant une voix familière. Cette capacité est la preuve qu'une certaine constance existe dans la phase de production de la parole par un même individu.

1.6.3/ Variabilité inter-locuteur

La variabilité inter-locuteur est un phénomène majeur en reconnaissance de la parole. Comme nous venons de le rappeler, un locuteur reste identifiable par le timbre de sa voix malgré une variabilité qui peut parfois être importante. La contrepartie de cette possibilité d'identification à la voix d'un individu est l'obligation de donner aux différents sons de la parole une définition assez souple pour établir une classification phonétique commune à plusieurs personnes.

La cause principale des différences inter-locuteurs est de nature physiologique. La parole est principalement produite grâce aux cordes vocales qui génèrent un son à une fréquence de base, le fondamental. Cette fréquence de base sera différente d'un individu à l'autre et plus généralement d'un genre à l'autre, une voix d'homme étant plus grave qu'une voix de femme, la fréquence du fondamental étant plus faible. Ce son est ensuite transformé par l'intermédiaire du conduit vocal, délimité à ses extrémités par le larynx et les lèvres. Cette transformation, par convolution, permet de générer des sons différents qui sont regroupés selon les classes que nous avons énoncées précédemment. Or le conduit vocal est de forme et de longueur variables selon les individus et, plus généralement, selon le genre et l'âge. Ainsi, le conduit vocal féminin adulte est, en moyenne, d'une longueur inférieure de 15% à celui d'un conduit vocal masculin adulte. Le conduit vocal d'un enfant en bas âge est bien sûr inférieur en longueur à celui d'un adulte. Les convolutions possibles seront donc différentes et, le fondamental n'étant pas constant, un même phonème pourra avoir des réalisations acoustiques très différentes.

La variabilité inter-locuteur trouve également son origine dans les différences de prononciation qui existent au sein d'une même langue et qui constituent les accents régionaux. Ces différences s'observeront d'autant plus facilement qu'une communauté de langue occupera un espace géographique très vaste, sans même tenir compte de l'éventuel rayonnement international de cette communauté et donc de la probabilité qu'a la langue d'être utilisée comme seconde ou, pire, troisième langue par un individu de langue maternelle étrangère. Là aussi, la définition phonétique tout autant qu'une définition stricte d'un vocabulaire ou d'une grammaire peuvent être mises à mal.

La variabilité inter-locuteur telle qu'elle vient d'être présentée permet de comprendre aisément pourquoi les méthodes de reconnaissance des formes fondées sur la quantification de concordances entre une forme à analyser et un ensemble de définitions strictes plus ou moins formelles ne peuvent être appliquées, avec un succès limité, qu'à des applications où le nombre de définitions est restreint, limitant ainsi le nombre des possibles. D'une manière générale, la définition assez floue des différents phonèmes ou des différents mots d'une langue est la cause de nombreuses erreurs de classification dans les systèmes de décodage acoustico-phonétique, DAP. Mais la variabilité inter-locuteur, malgré son importance évidente, n'est pas encore la variabilité la plus importante car les différences au sein des classes phonétiques sont en nombre restreint. L'environnement du locuteur est porteur d'une variabilité beaucoup plus importante, comme nous allons le voir brièvement dans le paragraphe suivant et de manière plus approfondie au chapitre 3.

1.6.4/ Variabilité due à l'environnement

La variabilité liée à l'environnement peut, parfois, être considérée comme une variabilité intra-locuteur mais les distorsions provoquées dans le signal de parole sont communes à toute personne soumise à des conditions particulières. La variabilité due à l'environnement peut également provoquer une dégradation du signal de parole sans que le locuteur ait modifié son mode d'élocution. Cette variation, considérée comme du bruit, sera étudiée ultérieurement.

La variabilité environnementale due au locuteur peut tout d'abord être de nature physiologique. Ainsi, un système mécanique provoquant une déformation du conduit vocal provoquera immanquablement une variation dans le signal de parole produit. Ces contraintes physiques sont généralement rencontrées dans les systèmes de transport où une posture particulière, ou une accélération lors du déplacement, pourront provoquer une déformation.

Les moyens de transport peuvent également entraîner d'autres déformations du signal, d'origine psychologique. Le bruit ambiant peut ainsi provoquer une déformation du signal de parole en obligeant le locuteur à accentuer son effort vocal. Enfin, le stress et l'angoisse que certaines personnes finissent par éprouver lors de longs voyages peuvent également être mis au rang des contraintes environnementales susceptibles de modifier le mode d'élocution.

1.6.5/ Spectrogrammes

Pour illustrer notre propos, nous allons maintenant présenter quelques figures présentant la variabilité du signal en parole. Ces figures utilisent une représentation graphique largement employée en phonétique : le spectrogramme.

Le spectrogramme est un outil de visualisation utilisant la technique de la transformée de Fourier et donc du calcul de spectres. Il a commencé à être largement utilisé en 1947, à l'apparition du sonagraphe [pierrel91], et est devenu l'outil incontournable des études en phonétique pendant de nombreuses années. L'apparition de l'informatique puis d'écrans graphiques de bonne qualité a permis d'abandonner tout matériel comme le sonagraphe mais la technique du spectrogramme est encore aujourd'hui largement utilisée du fait de sa simplicité de mise en œuvre et du grand nombre d'études qui ont déjà été réalisées.

Le spectrogramme permet de mettre en évidence les différentes composantes fréquentielles du signal à un instant donné, une transformée de Fourier rapide [aho74] étant régulièrement calculée à des intervalles de temps rapprochés. Avant le calcul des transformées successives, le signal doit d'abord être préaccentué par un filtre du premier ordre pour égaliser les hautes fréquences dont l'énergie est toujours plus faible que celle des basses fréquences. Cette phase de préaccentuation du signal est suivie par une phase de fenêtrage, nécessaire du fait de la théorie qui sous-tend la transformée de Fourier. Dans cette méthode d'analyse, le signal est considéré comme indéfiniment stable et constitué d'un somme invariable de fonctions sinusoïdales de fréquences différentes. Pour contourner cette contrainte théorique d'invariabilité du signal, il faut convoluer le signal avec une fenêtre temporelle qualifiée de glissante puisque chaque calcul de spectre nécessite de convoluer le signal avec la fenêtre temporelle à un instant particulier. Différentes fenêtres temporelles existent mais chacune introduit une erreur résiduelle plus ou moins importante dans le spectre obtenu du fait de la forme choisie qui peut être, dans le pire des cas, triangulaire ou carrée. Le choix de la taille de la fenêtre, en nombre de points de convolution, est également important vis-à-vis de la qualité de l'analyse fréquentielle obtenue. Ainsi, une fenêtre de petite taille (avec un nombre de 128 points, par exemple) permettra d'obtenir une bonne analyse dans le domaine temporel, du fait de son étroitesse, mais ne permettra pas d'obtenir une bonne information fréquentielle, la taille de la fenêtre étant alors trop petite pour ne pas tronquer les phénomènes de basses fréquences. À l'inverse, une fenêtre de grande taille (plus de 512 points) permettra d'obtenir une bonne information fréquentielle mais ne permettra pas d'obtenir une bonne information temporelle car tout événement, même de courte durée, est jugé présent sur l'ensemble du pas de temps analysé puisque la théorie de la transformée de Fourier considère les signaux indéfiniment stables.

Une fois la convolution effectuée, la transformée de Fourier est calculée sur la totalité de la fenêtre, le reste du "signal" étant alors égal à 0. Ce processus permet d'obtenir un spectre qui correspond à une trame, un ensemble de trames calculées à intervalles réguliers permettant d'obtenir le spectrogramme désiré.

L'ensemble du processus de calcul d'un spectrogramme est résumé dans la figure 1.6 suivante.

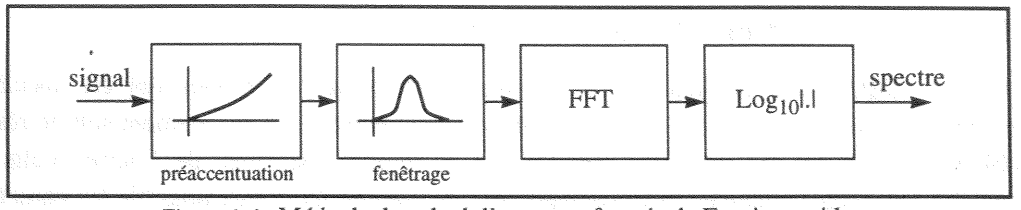


Figure 1.6 : Méthode de calcul d'une transformée de Fourier rapide
(d'après [calliope89])

Le mode de calcul du spectrogramme étant énoncé, nous allons maintenant illustrer la variabilité de la parole. La figure 1.7 présente deux signaux temporels, et les spectrogrammes qui y sont associés, d'une même phrase prononcée par deux locuteurs différents, tous deux de sexe féminin, d'âge, de taille et de culture similaires. Les signaux ayant servi à réaliser ces spectrogrammes sont issus du corpus TIMIT où nous avons choisi de prendre la phrase référencée "sa1", commune à tous les locuteurs du corpus. L'axe des abscisses du signal temporel représente le temps alors que l'axe des ordonnées représente l'amplitude du signal. L'axe des abcisses du spectrogramme représente également le temps, l'axe des ordonnées représentant la fréquence qui est, ici, comprise entre 0 et 8000 hertz (Hz). Les nuances de grisé du spectrogramme représentent l'énergie du signal pour une fréquence et à un instant donné. L'énergie minimale des spectrogrammes présentés est de 30 décibels (correspondant au gris le plus clair), l'énergie maximale étant, elle, de 100 décibels (correspondant au noir).

Une étude, même rapide, de ces deux graphiques permet de comprendre toutes les différences de bas niveau qui peuvent exister dans un message pourtant porteur de la même information. Ces différences dans le signal expliquent toute la difficulté qui peut être engendrée, et toutes les erreurs qui peuvent être provoquées, par des méthodes ne mettant en œuvre qu'une comparaison générale entre un signal de parole à interpréter et sa définition phonétique exacte, que cette définition soit enregistrée sous forme de règles ou sous la forme d'un corpus de formes de référence.

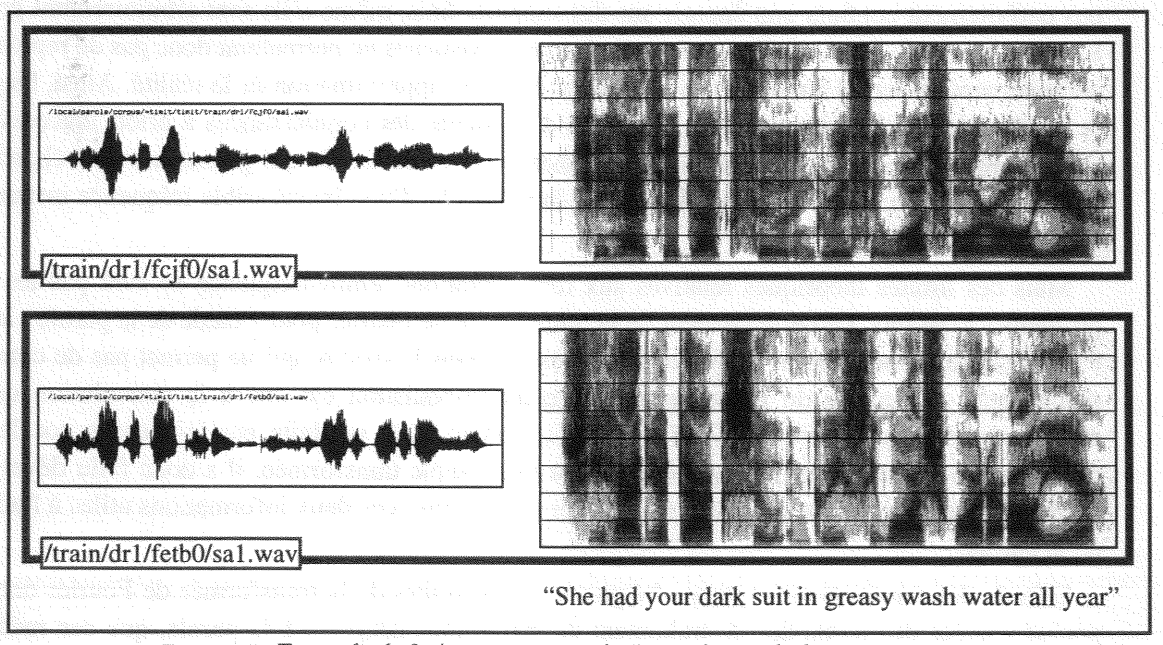


Figure 1.7 : Exemple de 2 signaux temporels (à gauche) et de 2 spectrogrammes
(à droite) d'une même phrase prononcée par deux locuteurs différents
(signal extrait du corpus TIMIT)

Le spectrogramme, fondé sur la transformée de Fourier, n'est cependant pas la seule méthode d'analyse existante, comme nous allons le voir maintenant.

1.7/ Les représentations du signal de parole

Il existe différentes méthodes de représentation du signal. Certaines ont été spécifiquement développées pour l'étude ou la compression de signaux de parole. Elles essaient soit de résoudre les problèmes posés par les méthodes fondées sur la seule transformée de Fourier, cette méthode d'analyse présentant quelques inconvénients, soit de simuler du mieux possible les caractéristiques de l'oreille humaine. De nombreux exemples de telles méthodes pourront être trouvés dans [cooke93].

1.7.1/ Problèmes posés par la transformée de Fourier

La transformée de Fourier et l'implantation algorithmique efficace qui y a été associée, la transformée de Fourier rapide, présente de nombreux avantages en tant que méthode d'analyse temps-fréquence. La rapidité de sa mise en œuvre l'a propulsé au rang d'élément incontournable des systèmes de traitement de signal. Mais, après la naissance de la notion de représentation temps-fréquence, qui fait suite à l'utilisation de représentations spectrographiques, les études théoriques du domaine ont permis de mettre à jour quelques désavantages qui sont impossibles à éliminer et qui constituent ainsi les limites d'exploitation de la transformée de Fourier [flandrin93].

Au rang de ces problèmes se trouve le compromis entre finesse d'analyse en fréquence et en temps, comme nous venons de le voir au paragraphe 1.6.5. Le fait que la transformée de Fourier ne prenne pas en compte les dépendances temporelles implique, lorsque cette méthode est adaptée aux signaux non stationnaires, de considérer l'inégalité d'Heisenberg-Gabor. Cette inégalité postule qu'un signal ne peut être concentré sur des supports temps et fréquence qui soient, simultanément, arbitrairement petits.

Une autre constatation exhibe une limitation qui dépasse le cadre de l'inégalité d'Heisenberg-Gabor et qui nous amène à nous demander ce que les transformées de tous types permettent de représenter. La théorie de Slepian-Pollack-Landau prouve en effet qu'un signal ne peut pas parfaitement confiner son énergie sur des supports finis, même s'ils sont arbitrairement grands. La transformée de Fourier et les autres transformées existantes ne permettent donc pas de représenter correctement un signal temporel discret, qui est déjà une approximation de la réalité. Ainsi, bien que la transformée de Fourier permette d'extraire d'un signal des connaissances a priori inaccessibles, l'information obtenue ne peut pas, théoriquement, être correcte. Ce qui pousse certains chercheurs du domaine à dire que nous serons toujours à la recherche d'une inaccessible fréquence instantanée (Y. Meyer dans [flandrin93]).

Mais ces limites théoriques relatives aux représentations temps-fréquence ne sont pas les seuls problèmes existants. Le défaut majeur de la transformée de Fourier pour l'étude de la parole vient de l'inévitable intermodulation source/conduit présente dans le spectre qui ne permet pas de connaître précisément la hauteur du fondamental. Cette intermodulation est due à la convolution qui est réalisée par le conduit vocal sur la fréquence fondamentale produite par les cordes vocales. La déconvolution ne pouvant pas être réalisée par une simple transformée, il a donc fallu développer une technique particulière capable de la réaliser pour fournir ces deux informations utiles à l'analyse de la parole.

L'étude des représentations temps-fréquence et les limites de la transformée de Fourier ont donc poussé à créer des méthodes de traitement de signal plus adaptées à la parole, que ces méthodes soient spécifiques à la recherche ou qu'elle soient créées pour des applications plus industrielles avec une volonté de compression maximale du signal agrémentée d'une conservation de sa qualité subjective. Nous allons maintenant voir quelques une des ces méthodes.

1.7.2/ Méthodes adaptées à la parole

Ce paragraphe présente de manière assez succincte les grands principes qui ont conduit aux différentes représentations du signal actuellement les plus utilisées. Ces grandes méthodes sont les

cepstres, le codage par prédiction linéaire, le codage par modulation et les modèles d'audition.

1.7.2.1/ Représentations cepstrales

Pour séparer les deux informations présentes dans le signal de parole que sont la fréquence fondamentale et la transformation, supposée linéaire, effectuée par le conduit vocal, il est nécessaire d'effectuer une déconvolution a posteriori du signal pour connaître la contribution des cordes vocales et du conduit vocal lors de la génération du signal qui a, par la suite, été observé en entrée du système. Cette déconvolution peut être effectuée grâce au cepstre. Il est à noter que le nom même de cepstre est défini à partir du mot spectre (cepstre = (spec)$^{-1}$tre). De même, la représentation temps-fréquence associée n'est plus qualifiée de fréquentielle mais de quéfrentielle.

Le cepstre est une méthode qui se fonde sur la transformée de Fourier mais qui, grâce à une méthode efficace, permet d'isoler la fréquence initiale du fondamental de la transformation qui a été opérée par le conduit. Comme pour le calcul du spectrogramme, le signal est préaccentué puis convolué avec une fenêtre mobile. Une première transformée de Fourier est alors calculée pour obtenir un spectre du signal, comme pour un spectrogramme. Ces coefficients sont ensuite transformés par logarithme module. La convolution étant un opérateur multiplicatif, ce passage par les logarithmes permet de passer les coefficients dans un espace additif. Une transformée de Fourier inverse permet alors d'obtenir un cepstre dont un coefficient représente le fondamental, les autres coefficients permettant d'obtenir le spectre de la convolution effectuée sur le fondamental. Cette méthode de calcul des cepstres est élémentaire [calliope89], il existe également des méthodes itératives effectuant un lissage, ce qui permet d'obtenir des cepstres de meilleure qualité.

Une extension possible des cepstres est leur passage dans un espace fréquentiel non linéaire proche de l'audition humaine. Il est ainsi possible de modifier la procédure de calcul précédente pour que les coefficients obtenus soient répartis selon une échelle Mel. Une telle procédure, proposée dans [davis80], permet d'obtenir des coefficients cepstraux à échelle Mel, *Mel Frequency Cepstral Coefficients*, *MFCC*. Ces coefficients ont été très utilisés en RAP du fait des bons résultats qu'ils ont permis d'obtenir. [davis80] avait d'ailleurs comparé cette méthode à d'autres du même ordre avec des conclusions qui, déjà, laissaient entrevoir la qualité des informations extraites par la méthode MFCC. Parmi les méthodes auxquelles les MFCC avaient été comparés se trouvaient des méthodes fondées sur la prédiction linéaire.

1.7.2.2/ Codage prédictif linéaire

Le codage prédictif linéaire (*LPC*, *Linear Predictive Coding*) est une méthode de codage et de représentation de la parole [markel76]. Elle repose principalement sur l'hypothèse que la parole peut être modélisée par un processus linéaire. Il s'agit donc de prédire le signal à un instant n à partir des p échantillons précédents (équation 1.1). La parole n'étant cependant pas un processus parfaitement linéaire, la moyenne mobile que constitue la somme pondérée du signal sur p pas de temps introduit une erreur qu'il est nécessaire de corriger par l'introduction du terme $e(n)$.

Le codage par prédiction linéaire consiste donc à déterminer les coefficients a_k qui minimisent l'erreur $e(n)$, ceci en fonction d'un ensemble de signaux constituant un corpus d'apprentissage.

$$s(n) = \sum_{k=1}^{p} a_k \cdot s(n-k) + e(n) \qquad \text{(Éq. 1.1)}$$

La méthode du codage par prédiction linéaire est tout autant utilisée en RAP qu'en compression pour le transfert de la voix par téléphone ou radio. Elle n'est cependant pas parfaite puisque l'erreur de prédiction peut être importante sans qu'il soit possible, par cette méthode, de la corriger.

La méthode RELP, *Residual Excited Linear Prediction*, permet de réduire une partie de cette erreur. Le principe consiste à comparer, lors de la prédiction linéaire, le signal obtenu avec le signal original. L'erreur, obtenue par soustraction, représente la partie du signal original que le prédicteur

n'arrive pas à modéliser. Dans la méthode RELP, l'erreur résiduelle est passée dans un filtre passe-bas permettant de conserver l'erreur effectuée dans la seule bande fréquentielle allant de 0 à 1000 hertz. La sortie du filtre est alors codée et passée au receveur qui peut alors reconstruire un signal à partir de la prédiction et de l'erreur observée.

Pour pallier le problème de l'erreur résiduelle, d'autres méthodes fondées sur la prédiction linéaire ont été développées. Ainsi la méthode *CELP, Code Excited Linear Prediction*, permet d'effectuer une compression de la parole par codage d'une trame vis-à-vis de références stockées dans un corpus. Ainsi, une trame de parole sera codée selon une combinaison linéaire de certaines trames du corpus et c'est cette combinaison linéaire qui sera considérée à la place de la trame dans les traitements ultérieurs. Cette méthode de codage de la parole est surtout employée pour la compression et la transmission de la parole à de faibles débits [dod93].

L'idée du codage prédictif linéaire n'a pas encore été abandonnée malgré son apparente simplicité et l'évident taux d'erreur introduit par l'hypothèse de linéarité de la production de la parole. Le groupe en charge de l'étude du GSM ("Groupe Spécial Mobile" devenu depuis "Global System for Mobile"), après avoir étudié différents systèmes de codage de la parole sur des critères de qualité subjective, de complexité algorithmique et de besoin en bande passante, a retenu le codage prédictif linéaire dit RPE-LPC (*Regular-Pulse Excited - Linear Predictive Coding*) agrémenté d'un système itératif de prédiction à long terme [scourias95]. Cet ensemble algorithmique permet de transmettre un signal de parole de bonne qualité à des taux de transfert de 13,2 kbps (kilobits par seconde). Ce choix va cependant à l'encontre des tendances actuelles de codage de la parole par des méthodes permettant de conserver une qualité objective au signal de parole lors de sa transmission.

1.7.2.3/ Codage dit de Modulation par Impulsion et Codage

Le codage prédictif linéaire peut provoquer des erreurs dégradant fortement la qualité du signal de parole. Il est cependant précieux car il permet de transmettre de la parole à de très faibles débits. D'autres méthodes, dites de codage par modulation (*PCM, Pulse Code Modulation*), ou plus exactement de modulation par impulsion et codage, permettent d'obtenir une bien meilleure qualité de parole mais nécessitent des débits beaucoup plus importants : l'espace nécessaire à la représentation de la parole est donc plus important que pour les méthodes présentées dans le paragraphe précédent.

Le codage par modulation n'est pas spécifique à la parole car très peu de connaissances relatives au domaine ont été prises en compte dans sa mise au point. Le principe de base consiste à quantifier le signal à représenter ou à transmettre selon un certain nombre de plages de même grandeur. Ce nombre de plages représente la qualité de la quantification. Le nombre de plages va également déterminer le nombre de bits nécessaire à la représentation binaire. Le codage par modulation est donc une méthode numérique qui suit le même principe que la conversion de l'analogique vers le numérique. Le codage par modulation peut d'ailleurs être facilement appliqué à un signal numérique.

Ce principe de base peut être raffiné. Il est tout d'abord possible de quantifier le signal selon une échelle logarithmique plutôt que linéaire, ce qui permet d'obtenir une bonne quantification de la parole. Ensuite, plutôt que de transmettre les échantillons eux-même, il est possible de coder et de simplement transmettre la différence entre deux échantillons successifs. Les échantillons successifs d'un signal de parole étant fortement corrélés, cette technique réduit l'espace des valeurs à coder. La généralisation de ce principe sur plusieurs échantillons, qui assureraient le codage de leur successeur, permet d'obtenir une prédiction linéaire de l'échantillon suivant et une mesure de l'erreur effectuée dans la prédiction. La quantification de cette différence et sa transmission permet de définir la méthode par codage de modulation différentielle (*DPCM, Differential Pulse Code Modulation*).

Enfin, la définition d'un quantificateur par prédiction linéaire dont les coefficients sont constamment adaptés, par la méthode des moindres carrés, au signal de parole transmis permet de

définir une méthode réduisant encore l'erreur de prédiction. Cette méthode, différentielle et adaptative, est connue sous le nom d'*Adaptive Differential Pulse Code Modulation* (*ADPCM*).

Les techniques que nous venons d'exposer sont très utilisées à l'heure actuelle dans le monde des télécommunications à des débits variant de 32 à 64 kbps. Elles continueront à l'être demain, notamment sur les réseaux numériques à intégration de service (RNIS), ce type de réseaux étant parfaitement adapté à des signaux définis par quantification.

1.7.2.4/ PLP

La méthode *PLP*, [hermansky85], [hermansky90] et [morgan91], *Perceptual Linear Prediction* (ou *Perceptually based Linear Prediction*), est une méthode inspirée du principe de prédiction linéaire. Elle combine ce principe à une représentation du signal qui suit l'échelle humaine de l'audition. Elle est à l'origine de toute une famille de techniques de traitement du signal de parole que nous verrons dans le paragraphe suivant.

Cette méthode peut être résumée en trois phases de traitements successifs. Le signal de parole est tout d'abord analysé pour obtenir un spectre suivant une échelle d'audition. Ce spectre est ensuite modifié par une interpolation et une transformée de Fourier inverse, le signal obtenu étant passé dans un filtre pour réduire la dimension du spectre et augmenter la résolution fréquentielle. Une troisième étape, qui peut être omise, permet de reconstruire un signal de parole par filtrage inverse, passage dans le domaine fréquentiel hertzien et désaccentuation.

La première étape est précisément constituée par :

- un analyse en bandes critiques selon une échelle Bark par un banc de filtres,
- une préaccentuation des valeurs obtenues selon une courbe suivant approximativement les même principes que les traitements effectués par l'oreille, avec accentuation des basses fréquences et atténuation des hautes fréquences,
- une application de la loi de préaccentuation de Steven.

La deuxième étape est, elle, constituée des phases suivantes :

- une interpolation des sorties des filtres du banc pour obtenir un spectre sur une échelle fréquentielle auditive,
- une transformée de Fourier inverse qui permet de ramener le spectre obtenu dans le domaine temporel,
- une résolution d'un ensemble d'équations linéaires pour obtenir les coefficients issus d'un filtre tout pôle d'ordre 5 (ce qui permet d'obtenir au moins deux sommets caractéristiques selon [hermansky85]).

Cette méthode a pour avantage de permettre une analyse et/ou un codage de la parole qui respectent le principe de la prédiction linéaire, qui suivent l'échelle fréquentielle observable dans l'oreille et, enfin, qui réduisent l'espace de représentation. Cette méthode a été, par la suite, améliorée pour résister à certaines conditions de bruit.

1.7.2.5/ Rasta PLP

La méthode PLP [hermansky90], dont l'algorithme repose sur des spectres à court terme de la parole, résiste difficilement aux contraintes qui peuvent lui être imposées par la réponse fréquentielle d'un canal de communication. Pour atténuer les effets de distorsions spectrales linéaires, [hermansky91a], [hermansky91b] propose de modifier l'algorithme PLP en remplaçant le spectre à court terme par un spectre estimé où chaque canal fréquentiel est modifié par passage à travers un filtre. Cette modification est à la base de la méthode RASTA PLP, RASTA étant l'acronyme de *RelAtive SpecTrAl* [hermansky91b]. La mise en place de ce filtrage permet, lorsqu'il est effectué dans le domaine spectral logarithmique, de supprimer les composantes spectrales constantes, supprimant ainsi les effets de convolution du canal de communication.

Différentes études réalisées avec cette méthode [hermansky91b], [hermansky92], [hermansky94] ont permis de confirmer les bonnes qualités de cette méthode relativement aux distorsions et ses moindres qualités face aux bruits qualifiés d'additifs, signe de la présence de plusieurs sources sonores dans un même environnement. Pour améliorer encore la méthode PLP, [morgan92] définit la méthode J-RASTA, plus résistante aux bruits additifs que ne l'est la méthode RASTA, par adjonction d'un filtrage passe-bas dans le domaine spectral.

1.7.2.6/ Modèles d'audition

L'oreille est un organe qui est caractérisé par des échelles de traitement particulières et non linéaires, comme nous l'avons vu dans le paragraphe 1.4. Différents modèles de traitement du signal essaient de reproduire cette non linéarité sans pour autant reproduire exactement le comportement de l'oreille interne (voir les paragraphes précédents).

D'autres modèles de représentation du signal de parole se fondent sur une modélisation beaucoup plus précise et exacte du comportement de l'oreille humaine. Ces modèles, les modèles d'audition, ne sont cependant pas tous équivalents puisqu'ils prennent en compte certaines caractéristiques de l'oreille humaine à des degrés divers.

Le modèle le plus connu est le modèle de Seneff [seneff88]. Ce modèle est composé de deux étages disjoints. Un premier niveau est constitué d'un banc de filtres qui réalisent une analyse spectrale. Le deuxième étage modélise la cochlée par l'intermédiaire de processus tels que la saturation, l'adaptation ou le masquage. Ce deuxième étage est donc un modèle des cellules ciliées qui génèrent l'influx nerveux. Les différents paramètres du modèle ont été ajustés par l'auteur en fonction d'observations expérimentales.

D'autres modèles existent. Ainsi, le modèle de Lyon [slaney88] schématise la cochlée à l'aide de filtres du second ordre. Le modèle de Patterson-Holdsworth [slaney93b] utilise un autre type de filtres et n'implante pas de mécanisme interne de contrôle du gain. Ce mécanisme est, par contre, implanté dans le modèle de Meddis [meddis86], [hewitt90], un autre modèle de cochlée.

De plus, les données issues d'un modèle de cochlée peuvent être traitées, dans chaque canal, par autocorrélation, permettant ainsi d'obtenir un corrélogramme [slaney93a].

1.7.3/ Méthodes modernes de représentation temps-fréquence

Les méthodes de représentation temps-fréquence ont grandement évolué ces dernières années [loughlin93] avec l'apparition des méthodes de transformation en ondelettes. Contrairement à la transformée de Fourier, cette méthode ne présuppose pas que le signal est infiniment stationnaire : l'étude du signal se fait grâce à une équation produisant une onde locale de durée variable car paramétrable. Cette onde est de forme sinusoïdale et conserve ainsi l'idée d'une approximation du signal par une somme infinie de sinusoïdes telle qu'elle est employée dans l'analyse de Fourier. Mais la durée variable de cette onde permet de supprimer l'étape de lissage du signal par une fenêtre mobile, fenêtre qui provoquait des distorsions dans la transformée [harris78].

La théorie des ondelettes [hubbard95] a permis de nombreux développements et, de même qu'il existe plusieurs types de fenêtres mobiles, il existe plusieurs types d'ondelettes, aussi appelées atomes temps-fréquence [meyer92]. Une ondelette peut ainsi suivre la définition donnée par Grossman-Morlet (équation 1.2),

$$\frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \qquad b \in Z \text{ et } a > 0 \qquad \text{(Éq. 1.2)}$$

ou suivre la définition de Daubechies (équation 1.3),

$$2^{j/2}\psi\left(2^{j}t-k\right) \qquad j \in Z \text{ et } k \in Z \qquad \text{(Éq. 1.3)}$$

soit, encore, suivre la définition donnée par Gabor-Malvar (équation 1.4).

$$w(t-l)\cos[\pi(k+1/2)(t-l)] \qquad l \in Z \text{ et } k \in N \qquad \text{(Éq. 1.4)}$$

Toutes ces équations ont en commun de donner une hiérarchie permettant d'analyser le signal selon différentes échelles, plus ou moins fines. Ces méthodes d'analyse du signal permettent ainsi d'avoir un bien meilleur compromis à toutes les fréquences que ne le permettait la transformée de Fourier, supprimant par conséquent la nécessité d'effectuer a priori un compromis entre la résolution en fréquence et la résolution en temps.

De telles méthodes ont déjà été appliquées à la parole [lienard90] mais la transformée de Fourier reste, aujourd'hui encore, la principale méthode d'analyse du signal du fait du nombre de connaissances accumulées au cours des recherches passées.

1.7.4/ Méthodes résistantes aux bruits

La tendance actuelle à vouloir développer des systèmes de RAP ayant une bonne résistance à des conditions environnementales variables et parfois agressives a également conduit au développement de techniques particulières.

Ces méthodes ne sont pas à proprement parler différentes des méthodes que nous venons de voir. Elles reprennent généralement des concepts existants et y ajoutent des principes mathématiques qui permettent de traiter le bruit environnemental.

Ces méthodes peuvent être classées dans trois catégories différentes :

- les méthodes paramétriques offrant une résistance intrinsèque au bruit,
- les méthodes permettant d'estimer la parole propre, c'est à dire débarrassée du bruit environnemental,
- les méthodes fondées sur des traitements pouvant s'accommoder d'un signal bruité.

Les deux premières classes ont pour but principal le débruitage du signal de parole, permettant ainsi de ne pas modifier l'étape suivante du décodage acoustico-phonétique. La troisième méthode utilise le principe opposé et n'essaie pas de débruiter le signal avant de l'analyser. La technique d'analyse est donc elle-même robuste.

Il est possible de citer, au rang des méthodes paramétriques intrinsèquement robustes, la normalisation par moyenne cepstrale, qui est très populaire [stern94]. Le cepstre n'obtenant cependant pas de bons résultats en milieu bruité, certaines recherches ont été menées pour trouver un opérateur différent du logarithme pour effectuer la déconvolution. Il est alors possible de parler d'analyse cepstrale généralisée, cette analyse pouvant être faite de différente manière [lim79a], [kobayashi84]. Il existe également des méthodes fondées sur l'analyse linéaire discriminante selon une échelle Mel [hunt89], [siohan95] ou la prédiction linéaire [hernando94], les méthodes RASTA-PLP [hermansky91b] et J-RASTA-PLP [morgan92] faisant bien sûr partie de cette dernière catégorie de techniques. Une méthode permettant de déconvoluer efficacement plusieurs sources sonores devrait également permettre d'obtenir de bons résultats dans le cadre de la reconnaissance de la parole en milieu bruité puisqu'elle permet d'effectuer une séparation sonore de manière aveugle [bell95]. Elle n'a cependant pas encore été appliquée puisqu'elle nécessite l'emploi de plusieurs sources de signal.

Les méthodes permettant d'estimer la parole propre utilisent le principe général de l'estimation d'un spectre du bruit qui est ensuite soustrait au spectre du signal brut, principe permettant d'améliorer le rapport signal-sur-bruit. Il existe des méthodes se réclamant de la soustraction dans le domaine spectral [lim79b] ou dans le domaine cepstral [acero90], [liu94]. Certaines méthodes permettent d'obtenir de meilleurs résultats en ayant une connaissance a priori des caractéristiques de la parole et du bruit, essayant ensuite de retrouver une combinaison entre ces deux signaux [gagnon92]. Il est enfin possible d'effectuer un filtrage de la parole suivant des paramètres déterminés de manière probabiliste [neumeyer94]. Ce filtrage peut également être effectué à l'aide

de réseaux connexionnistes [kuo94a]. Dans les deux cas, le filtre obtenu ne sera valable que dans des conditions de bruit connues.

Enfin, le choix de ne pas modifier le signal observé en entrée oblige à mettre en œuvre des techniques adaptées au bruit lors de la phase de classification du signal. Il est ainsi possible de mettre en place une modélisation en parallèle du bruit et de la parole propre pour recombiner correctement les deux processus lors de la phase de reconnaissance [varga90], [gales95]. Cette technique effectue une décomposition des connaissances lors de l'apprentissage, modélisant d'une part la parole propre et de l'autre le bruit, et recombine cette connaissance lors de la phase de reconnaissance. Cette recombinaison a jusqu'à présent été mise en œuvre avec les seuls réseaux de Markov (dont nous parlerons au chapitre 2, paragraphe 2.1.3) car ils permettent de modéliser un bruit avec un réseau d'architecture très simple. Les bruits modélisés jusqu'à présent n'ont cependant été que des bruits stationnaires ou quasi stationnaires [gales93], [gales95] alors que cette technique semble a priori applicable à des bruits plus complexes. À l'inverse, il est possible d'essayer de décomposer le signal d'entrée en un signal de parole propre et un signal de bruit par l'emploi de deux et non plus d'une seule fonction gaussienne. Ce choix a été fait dans les modèles *Adaptive Noise Prototypes* [nadas88] et *Speech and Noise Decomposition* [varga90]. Le filtrage de Wiener est une méthode totalement différente permettant de convertir la parole bruitée en parole propre par un processus de minimisation de la variance de la différence entre parole propre et parole bruitée [beattie92]. Les variances des modèles peuvent également être adaptées par régression linéaire [digalakis95]. Cette dernière technique est aujourd'hui plus particulièrement utilisée pour adapter le modèle à un locuteur particulier mais reste applicable aux cas de bruits additifs bien que cette technique ne fournisse pas, alors, de bons résultats. L'ajustement stochastique des modèles [sankar95] essaie, lui, de transformer les paramètres du modèle de la parole propre en paramètres adaptés au nouvel environnement par analyse d'une phrase test produite dans le nouvel environnement. Enfin, la technique de masquage du bruit, inspirée d'observations psycho-acoustiques, essaie de tenir compte de la prédominance du bruit ou de la parole dans différentes parties du spectre, que ce dernier soit linéaire ou logarithmique. Il existe différents algorithmes pour tenir compte de ce phénomène dont [klatt79] et [holmes86].

Notons qu'il est possible de combiner certaines de ces technique pour obtenir de meilleurs résultats en tirant parti des qualités de chacune des techniques choisies. Nous reviendrons sur ces techniques dans le chapitre 5. Il est possible de consulter [gong95] pour trouver une classification qui, si elle est identique dans ses grandes classes, diffère quelque peu au niveau du détail.

CHAPITRE 2 : CONNEXIONNISME

> Les dogons distinguent deux types de parole, qu'ils nomment parole sèche et parole humide. La parole humide est celle qui fut donnée aux hommes : c'est le son audible. La parole sèche, ou parole première, est l'attribut de l'Esprit Premier. C'est une parole indifférenciée, sans conscience de soi, qui existe en l'homme comme en toute chose. Mais l'homme ne la connaît pas : c'est la pensée divine, dans sa valeur potentielle, et, sur notre plan microscopique, c'est l'inconscient.
>
> M. Griaule et G. Dieterlen
> Revue de la Société des Africanistes

Résumé

Ce chapitre nous permet de présenter les trois grandes techniques de reconnaissance des formes qui sont utilisées en Reconnaissance Automatique de la Parole (RAP) : l'alignement temporel, les réseaux de Markov et les modèles connexionnistes. La présentation des modèles connexionnistes sera précédée d'une brève présentation des connaissances de la neurobiologie qui ont servi de base à l'établissement des techniques neuromimétiques.

2.1/ Le pandémonium de la reconnaissance des formes

2.1.1/ Étendue de notre étude bibliographique

Cet état de l'art des méthodes de reconnaissance des formes ne se veut pas exhaustif. Nous l'avons principalement restreint aux méthodes de reconnaissance des formes utilisées en parole et nous l'avons encore plus particulièrement restreint aux méthodes connexionnistes. Ces différentes restrictions nous ont d'ailleurs poussé à focaliser le titre sur le connexionnisme bien qu'il ne soit pas ici le seul sujet de dissertation.

Ce chapitre, nous l'espérons, permettra au lecteur de se familiariser avec les notions élémentaires du connexionnisme ainsi qu'avec les relations qu'entretient celui-ci avec d'autres méthodes de classification. Cette comparaison permettra, en outre, de dégager les avantages et les inconvénients de cette méthode particulière, décriée par certains, par rapport aux autres.

Les méthodes connexionnistes elles-même ne seront pas étudiées de manière exhaustive. Ce domaine de recherche, en pleine ébullition depuis le milieu des années 80, est porteur de nombreux développements. Ces derniers permettent, parfois, d'unifier le monde connexionniste à d'autres théories, mathématiques ou biologiques, qui ne semblaient pas, a priori, faites pour le rencontrer. Nous reportons le lecteur aux quelques grandes références du domaine pour une présentation plus approfondie des architectures, des capacités de représentation et des problèmes que pose le connexionnisme : [hkp91], [bishop95] et l'éternel [pdp86]. De bonnes présentations des méthodes de reconnaissance des formes spécifiques à la parole pourront par ailleurs être trouvées dans [calliope89], [haton91] et [robinson96].

Ces limites étant données, nous allons maintenant exposer quelques unes des grandes méthodes de reconnaissance des formes utilisées en reconnaissance automatique de la parole.

2.1.2/ Alignement temporel

L'alignement temporel, plus connu sous l'acronyme de DTW, *Dynamic Time Warping*, est une méthode fondée sur un principe de comparaison d'un signal à analyser avec un ensemble de signaux stockés dans une base de référence. Le signal à analyser est comparé avec chacune des références et est classé en fonction de sa proximité avec une des références stockées. Le DTW est en fait une application au domaine de la reconnaissance de la parole [sakoe71] de la méthode plus générale de la programmation dynamique [bellman57]. Elle peut ainsi être vue comme un problème de cheminement dans un graphe [bellman58], [bridle95].

Ce type de méthode pose deux problèmes : la taille de la base de référence, qui doit être importante, et la fonction de calcul des distances, qui doit être choisie avec soin.

La taille de la base contenant les signaux de référence est directement liée aux capacités, variables, de reconnaissance du système d'alignement temporel. Chacun des signaux de référence est en effet stocké dans son état brut, sans compression d'aucune sorte. Ce stockage permet de disposer d'un vocabulaire dont la taille correspond au nombre de mots du vocabulaire multiplié par le nombre de locuteurs et le nombre des éventuelles répétitions des mots. Cette base de référence permet d'effectuer une mise en correspondance entre le signal stocké, d'une part, et sa retranscription symbolique d'autre part.

La taille de la base de référence est importante et implique une charge de travail non négligeable puisque la classification de chaque forme à analyser impose de la comparer à chaque forme de la base de référence. Donc, si la constitution de la base de référence est assez rapide et si le processus d'apprentissage est inexistant dans la méthode de l'alignement temporel, la phase d'utilisation nécessite une puissance de calcul non négligeable pour chaque référence atomique de signal à analyser. Le schéma de principe de la méthode est présenté dans la figure 2.1.

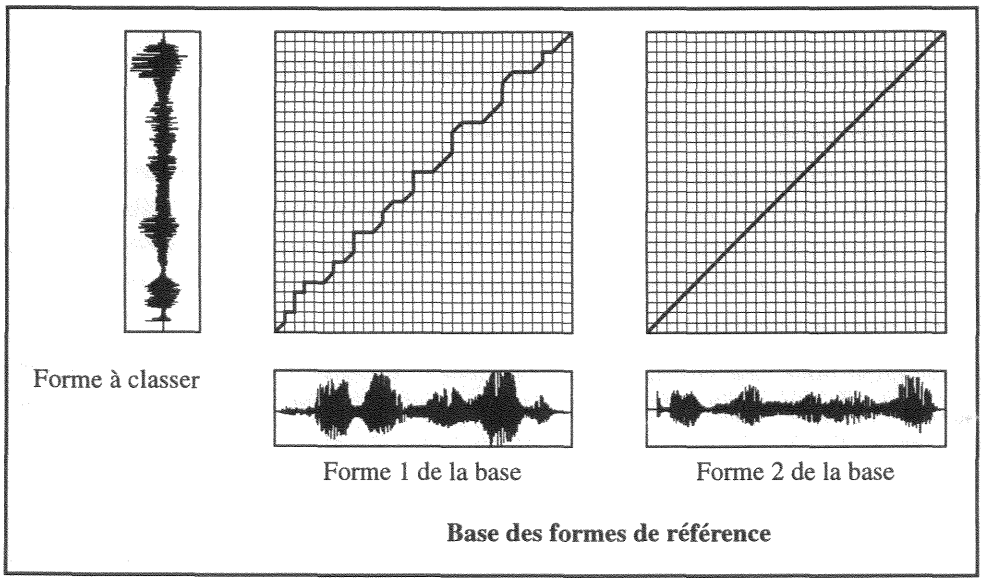


Figure 2.1 : Visualisation du cheminement de l'alignement temporel pour des formes de la base de référence.

Comme le montre le schéma de la figure 2.1, la forme choisie sera celle pour laquelle le chemin de mise en correspondance est le plus court, cette taille minimale marquant le peu de différences entre la forme à analyser et la forme de référence.

L'autre partie importante de l'alignement temporel est la définition de la fonction de recalage qui permet de calculer, selon certaines contraintes, la distance entre la forme à comparer et la forme de référence. La forme à analyser est mise en correspondance dans le plan temporel par l'algorithme

d'alignement qui essaie de trouver le plus court chemin dans le graphe ainsi constitué. Cette fonction de mise en correspondance définit une valeur pour chaque arc du graphe, ces valeurs favorisant l'axe médian qui correspond à une parfaite mise en relation de la forme à analyser et d'une forme de référence comme le montre la figure 2.1.

La fonction de recalage suit typiquement le schéma présenté dans la figure 2.2. La fonction *d(i,j)* est la fonction de calcul de la distance entre deux points successifs du graphe. Les valeurs α, β et γ permettent de définir une partie du comportement de la fonction *d* qui peut être soit symétrique ($\alpha = \gamma$) soit asymétrique ($\alpha \neq \gamma$). Ce calcul de distance entre deux nœuds successifs du graphe n'est cependant pas suffisant pour calculer la longueur totale du chemin parcouru dans le graphe. Une fonction supplémentaire, *G*, calcule une longueur totale qui permettra, après le calcul de cette longueur des chemins sur toutes les formes de la base de référence, de savoir à quel mot du vocabulaire préenregistré correspond la forme à classer. D'un point de vue mathématique, *M* et *N* étant les longueurs respectives de la forme à classer et de la forme de référence, on cherche sur l'ensemble du corpus le *G(M,N)* minimal. Le calcul de cette fonction *G* répond au même principe que le principe général énoncé par Bellman pour la programmation dynamique : toute sous-partie du chemin optimal est lui-même un chemin optimal. Des exemples de fonctions *d* et *G* de calcul de distance, qui peuvent être bien plus complexes que la fonction de recalage présentée en figure 2.2, pourront être trouvées dans [itakura75] ou [sakoe78]. Dans ces références, les fonctions présentées peuvent analyser jusqu'à 9 chemins différents pour *d*, la fonction *G* étant de complexité égale à celle de *d*.

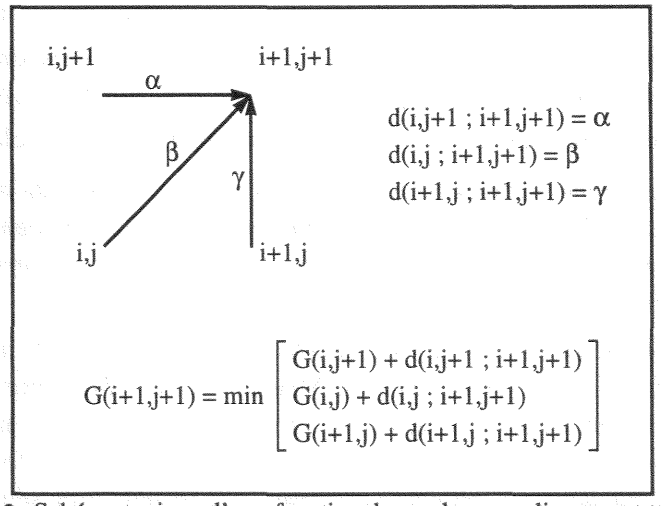


Figure 2.2 : Schéma typique d'une fonction de recalage en alignement temporel.

Cette méthode de reconnaissance des formes est, initialement, bien adaptée à la reconnaissance de mots isolés mais des extensions ont été développées pour permettre de l'appliquer à la parole continue [myers81a], [myers81b] et [sakoe79].

D'autres méthodes complémentaires ont par ailleurs été développées pour tenter de réduire la taille de la base des formes de référence par sélection optimale des formes à conserver ([rabiner77] par exemple). Ces méthodes reposent surtout sur une exploration statistique de la base des formes de référence et permettent d'obtenir une caractérisation des différents ensembles la constituant, ces ensembles correspondant aux différents symboles référencés dans la base. Une des techniques qu'il est possible d'employer pour ce faire est, par exemple, la méthode des plus proches voisins.

Certaines méthodes permettent de réduire ce temps de calcul à l'utilisation par apprentissage a priori de coefficients qui permettent de compacter la connaissance présente dans la base de référence qui devient ainsi un corpus d'apprentissage. Une première méthode mettant en œuvre ce principe de compactage de la connaissance est le modèle de Markov.

2.1.3/ Modèles de Markov et Modèles de Markov à états cachés

Les modèles de Markov et, plus particulièrement, les modèles de Markov à états cachés, plus connus sous le nom de HMM, *Hidden Markov Models*, permettent de synthétiser la connaissance contenue dans un corpus par apprentissage. Cette connaissance sera synthétisée, dans les modèles de Markov, par représentation probabiliste au sein de plusieurs graphes, chaque graphe correspondant à une classe du corpus d'apprentissage qui, en reconnaissance automatique de la parole, peut correspondre à un phonème ou à un mot [rabiner89].

D'un point de vue général, la différence de fonctionnement entre la méthode de l'alignement temporel et les modèles de Markov n'est pas fondamentale. Dans le cas des modèles de Markov, la forme à analyser est comparée à chacune des classes constituées en graphe. Dans le cas de la DTW, par contre, la forme à analyser est comparée à chacune des formes de référence dont le rattachement à une classe, et donc à une signification symbolique, ne sera utile qu'à un stade postérieur du traitement. Le modèle de Markov étant une représentation probabiliste des formes de référence, il ne s'agit plus ici de trouver un chemin de taille minimale mais de trouver une probabilité de cheminement dont la valeur est maximale. Cela se traduit mathématiquement par l'équation 2.1, [mori95].

$$\hat{W} = \arg max_W \; P\left(y_1^{\,T} \middle| W \right) \; P(W) \qquad \text{(Éq. 2.1)}$$

Au sein de cette équation, $\hat{W}$ correspond au candidat ayant la plus forte (*argmax*) probabilité de cheminement a posteriori (MAP, *maximum a-posteriori probability*), $P(y_1^{\,T}|W)$ correspond à la probabilité calculée avec le modèle acoustique tandis que P(W) correspond à la probabilité calculée par le modèle de la langue. $P(y_1^{\,T}|W)$ permet de calculer la probabilité d'observer une séquence $y_1^{\,T}$ à partir d'une séquence de vecteurs élémentaires y_t constituant la forme à classer.

Le modèle acoustique correspond à la mise en correspondance du signal à analyser avec les différents graphes de référence du modèle de Markov tandis que le modèle de la langue permettra de connaître la probabilité de génération d'une phrase vis-à-vis de l'ordonnancement des mots dans le corpus d'apprentissage.

Les modèles de Markov ne sont pas directement applicables à la parole. Tous les états des différents graphes de classe d'un modèle de Markov sont en effet porteurs d'une sémantique. Les états successifs qui peuvent être constitués à partir d'un signal de parole ne posséderont pas, eux, de sens véritable. Une autre conception des états d'un modèle de Markov doit donc être mise en œuvre pour modifier cet état de fait et permettre d'utiliser cette modélisation sémantique sur des séquences de symboles sans signification. Cette conception correspond aux réseaux de Markov à états cachés, HMM, un état caché étant un état de la chaîne d'analyse dont l'existence n'est pas rattachée par une sémantique. Grâce à ces derniers modèles, un corpus de parole peut être représenté de manière probabiliste selon un nombre de graphes correspondant au nombre de classes présentes dans le corpus. La sémantique n'est donc plus attachée aux nœuds du graphe mais au graphe lui-même.

Un HMM se définit par un couple de processus stochastiques (X, Y) [mori95]. Le processus X est une chaîne de Markov du premier ordre, qui n'est pas directement observable tandis que le processus Y correspond aux observations c'est à dire, dans le cas de la parole, à une séquence de variables aléatoires définies sur l'espace des paramètres acoustiques. Deux hypothèses formelles caractérisent les HMM tels qu'ils sont utilisés en reconnaissance automatique de la parole. La première de ces hypothèses porte sur l'ordre de la chaîne de Markov. La chaîne est supposée d'ordre 1 ce qui signifie que l'historique de la chaîne n'a aucune influence sur son évolution future si le présent est spécifié. La deuxième hypothèse est une hypothèse d'indépendance de la sortie qui spécifie que ni l'évolution de la chaîne de Markov, ni les observations passées n'influencent l'observation courante si la dernière transition de la chaîne est spécifiée.

Soit y ∈ Y, la variable représentant un vecteur d'observations et i, j ∈ X, des variables représentant les états du modèle, le modèle peut être représenté grâce aux paramètres suivants :

$$\begin{aligned} A &\equiv \{ a_{i,j} \mid (i,j \in X) \} \\ B &\equiv \{ b_{i,j} \mid (i,j \in X) \} \\ \Pi &\equiv \{ \pi_i \mid (i \in X) \} \end{aligned} \qquad \text{(Éq. 2.2)}$$

auxquels correspondent les définitions suivantes :

$$\begin{aligned} a_{i,j} &= P(X_t = j \mid (X_{t-1} = i)) \\ b_{i,j}(y) &= P(Y_t = y \mid ((X_{t-1} = i), (X_t = j))) \\ \pi_i &= P(X_0 = i) \end{aligned} \qquad \text{(Éq. 2.3)}$$

Les $a_{i,j}$ correspondent aux valeurs des transitions entre les états i de départ et les états j d'arrivée. Ces valeurs de transition correspondent aux probabilités de passer d'un état i à un état j. Les $b_{i,j}(y)$ correspondent aux probabilités d'observer le vecteur y à l'entrée du système après avoir effectué la transition de l'état i à l'état j. π_i donne, lui, la probabilité qu'a chaque état du graphe d'être un état initial. Tous ces paramètres sont définis par apprentissage et plusieurs algorithmes existent dont l'algorithme de Baum-Welch [baum67] très largement employé lors des premières mises en œuvre des réseaux de Markov, en concurrence avec l'algorithme de Viterbi [viterbi67]. Ces deux algorithmes ont aujourd'hui tendance à être abandonnés au profit d'autres plus efficaces tel que l'algorithme EM, abréviation d'*Expectation-Maximization* [dempster77].

Les réseaux de Markov, et plus précisément leurs extensions à états cachés, sont aujourd'hui très largement utilisés en RAP. Leur utilisation a cependant imposé des restrictions quand au nombre d'états de chaque graphe pour permettre un apprentissage correct. Ce nombre est ainsi réduit à trois ou cinq états par réseau lorsque la modélisation se fait respectivement au niveau du phonème ou du mot. Les nœuds du réseau représentant par définition des états stationnaires, il est aisé de comprendre qu'un nombre aussi restreint de nœuds peut être préjudiciable à la représentation de la connaissance.

D'autres extensions ont également été développées en modifiant la théorie sous-jacente des réseaux. Ainsi, l'hypothèse d'indépendance des transitions entre états peut être abandonnée au profit d'une hypothèse de dépendance entre les n dernières transitions effectuées. Cette dernière hypothèse conduit à la définition de réseaux de Markov d'ordre n. De tels réseaux d'ordre 2 sont aujourd'hui utilisés en parole [mari96] où ils permettent d'améliorer les résultats des réseaux d'ordre 1, l'hypothèse d'ordre 2 contraignant statistiquement plus le parcours du graphe.

Certains des derniers développements des réseaux de Markov font aujourd'hui également appel au paradigme connexionniste que nous verrons plus loin dans ce chapitre. Ces modèles de Markov ont été baptisés modèles hybrides [bourlard90], [bourlard93] par leurs créateurs et permettent de supprimer une partie de l'indépendance existant entre plusieurs réseaux de Markov. En effet, un seul et unique réseau connexionniste est utilisé pour l'émission des probabilités servant au parcours des différents graphes. Ce seul réseau connexionniste résoud ainsi le problème d'un apprentissage indépendant des formes du corpus de référence. La connaissance est donc représentée de manière plus synthétique et, surtout, de manière concurrente. Le parcours des différents graphes se fait cependant toujours de manière séparée sans que les différents parcours ne puissent être confrontés en cours de processus.

Les modèles de Markov sont aujourd'hui décriés au sein même de la communauté qui les soutient [bourlard96a], [bourlard96b]. Les causes de ces critiques, qui sont fondées et dont les effets sont compris, n'ont cependant pas encore été totalement circonscrites et aucune solution reconnue pour

son efficacité n'est apparue à ce jour. Les réseaux de Markov constituent cependant une étape importante dans le développement des méthodes de reconnaissance des formes.

2.1.4/ Évolutions de la modélisation

Le rapide survol de deux grandes méthodes de reconnaissance des formes que nous venons d'effectuer nous pousse à faire quelques constatations simples. Tout d'abord, l'apparition des réseaux de Markov à états cachés a permis de mettre en place une méthode d'apprentissage qui synthétise les données contenues dans la base de référence. Cette synthèse permet, par exemple, d'obtenir, pour une base de N classes possédant chacune M instances, une réduction de N×M à N du nombre de graphes à parcourir. Cette synthèse permet donc de faire une compression de la connaissance et, par conséquent, permet de réduire le temps de calcul nécessaire à la reconnaissance d'une forme.

Cependant, la compression effectuée sur les classes par les modèles de Markov n'est pas encore maximale. En effet, bien que les références communes à une classe soient regroupées dans un même graphe, le nombre de graphes se doit d'être égal au nombre de classes. En outre, la procédure d'apprentissage utilisée pour définir les modèles de Markov à partir des différents sous-ensembles du corpus d'apprentissage ne permet pas de prendre en compte les différences entre classes. Ainsi, un réseau de Markov ayant appris à reconnaître une classe particulière de formes d'entrée pourra tout à fait fournir une bonne probabilité de reconnaissance a posteriori pour un événement phonétique assez proche appartenant cependant à une classe différente. Ce type d'apprentissage, fondé sur l'estimation de la probabilité maximale (MLE, *maximum likelihood estimate*, [rabiner89]) est, à notre avis, le principal inconvénient de ce type de méthodes, qui apprend à reconnaître les similarités mais est incapable de prendre en compte les dissimilarités entre la classe de formes apprises et la forme analysée. Cet inconvénient est, en partie, dû à la répartition d'une connaissance aux bornes assez floues en plusieurs unités sémantiques la représentant. Il est aujourd'hui possible de modifier la méthode d'apprentissage de manière à ce que celle-ci soit capable d'extraire, d'une partie d'un corpus d'apprentissage, les caractéristiques remarquables d'une classe particulière par rapport à l'ensemble des données de tout le corpus tout en tenant compte, également, des différences entre classes (méthode *MMI, Maximisation of Mutual Information* [bahl86]). Le problème posé à l'utilisation par la répartition de la connaissance dans plusieurs réseaux de Markov n'en reste pas moins effectif.

Une technique de classification de données a été mise en œuvre ces dernières années de manière intensive dans de nombreux domaines de recherche. Cette méthode permet d'effectuer un pas supplémentaire et important sur la voie de la synthèse des connaissances grâce à l'utilisation d'un seul réseau pour représenter l'ensemble des classes présentes dans le corpus de référence. Cette méthode, le connexionnisme, s'inspire très largement d'une modélisation assez fine du cerveau humain et se veut donc être une méthode neurobiologiquement plausible.

2.2/ Neurobiologie

L'étude du cerveau humain, et d'une manière plus générale, de tout forme de vie sur Terre possédant dans son anatomie un organe dédié à la collecte d'informations et à la prise de décisions, est une science relativement jeune dans l'histoire de l'Humanité. L'étude d'une telle partie de l'anatomie s'est, en effet, longtemps heurtée à de nombreuses considérations philosophiques ou religieuses. Elle a, malgré tout, vu le jour et les outils modernes de la médecine permettent aujourd'hui de faire reculer à grands pas les frontières de nos connaissances au sujet de cet organe, siège présumé de notre pensée et de notre âme [simon94].

Le cerveau se caractérise par une organisation très complexe à analyser du fait du grand nombre de cellules, les neurones, et de liens entre cellules, les connexions synaptiques, qui le compose. Il est couramment admis que le chiffre moyen de neurones dans le cerveau est de l'ordre de 10^{12} tandis que

le nombre de connexions est estimé à 10^{15}. Ce grand nombre de neurones et de connexions conduit à un enchevêtrement qui est, aujourd'hui encore, très difficile à appréhender (figure 2.3). De plus, tous les neurones du cerveau n'ont pas la même architecture ni le même rôle au sein de l'organisation générale.

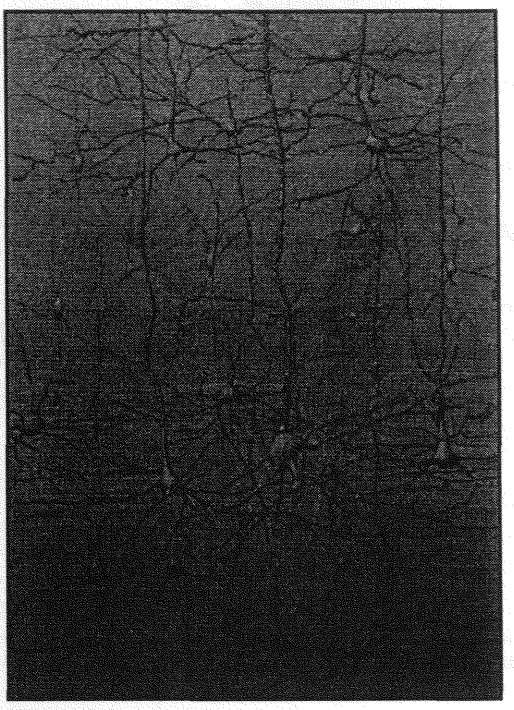


Figure 2.3 : Vue d'artiste de neurones et de leurs connexions synaptiques.

L'étude anatomique du cerveau ne se limite pas seulement à l'étude de son constituant de base qu'est le neurone. Le cerveau est en effet constitué en zones qui sont autant de sièges de traitements différents. Ces zones peuvent être étudiées suivant plusieurs granularités qui permettent d'observer soit des aires cérébrales, soit des colonnes corticales. Nous allons maintenant présenter les différents constituants du cerveau non d'un point de vue anatomique mais d'un point de vue mathématique.

2.2.1/ Modélisation du neurone

Le neurone, en tant qu'unité constituante du cerveau, mérite un regard particulier. Les premières études véritables remontent au début du siècle, avec les travaux de Ramon y Cajal qui établit les premiers croquis de neurones de types différents en 1909. Un neurone biologique est constitué de trois parties distinctes (figure 2.4) qui assurent 1) une collecte d'information, 2) l'intégration de cette information et 3) la restitution de l'information vers d'autres cellules. La collecte de l'information est effectuée par les dendrites du neurone qui réceptionnent l'information des unités afférentes par l'intermédiaire des connexions synaptiques. Cette information est alors acheminée, grâce à un processus électro-chimique, vers la cellule elle-même qui intègre cette information au sein de son noyau, également appelé soma. Cette information, une fois traitée, est répercutée en sortie de la cellule vers l'axone qui propage cette information vers d'autres cellules via les axones terminaux et les connexions synaptiques.

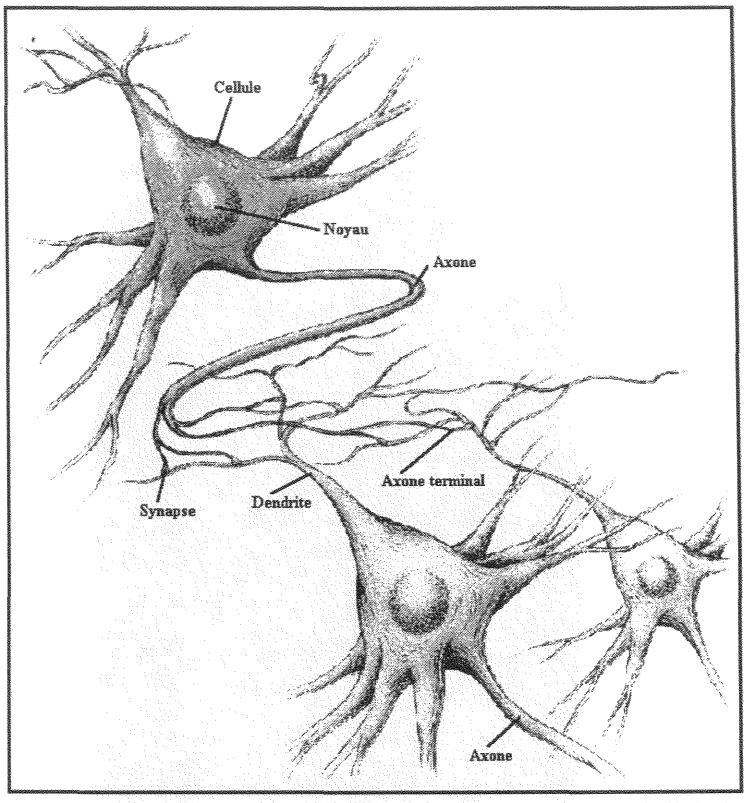


Figure 2.4 : Vue d'artiste des connexions neuronales dans le cerveau. Cellule émettrice en haut et cellules réceptrices en bas.

L'axone, qui permet à une cellule de propager son activité, peut être très long (figure 2.5). Cette longueur variable permet à une cellule d'être en contact avec d'autres qui ne sont pas forcément dans son voisinage proche, de manière à répercuter une information locale dans une autre région du cerveau.

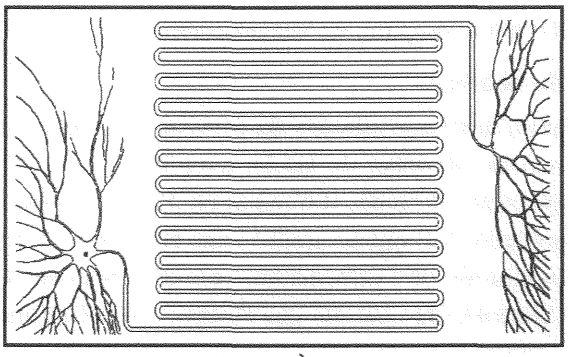


Figure 2.5 : Schématisation d'un neurone. À gauche, les dendrites et le corps de la cellule ; au centre, l'axone ; à droite, les axones terminaux.

Après ces premières études descriptives du neurone vinrent les premiers modèles de neurone formel. Le modèle de neurone formel le plus couramment utilisé est celui de McCulloch et Pitts [mcculloch43]. Cette modélisation (figure 2.6) caractérise le comportement du cerveau par l'agrégation de cellules élémentaires, chacune effectuant une sommation pondérée des entrées, le résultat de cette sommation étant ensuite transformé par une fonction de transfert non linéaire. Il faut noter que cette fonction non linéaire est indispensable à tout système de décision et permet, ici, de distinguer le neurone d'un simple système de classification linéaire.

La modélisation de McCulloch et Pitts est critiquable par sa trop grande simplicité et par son unicité vis-à-vis de cellules biologiques aux comportements parfois bien peu semblables. Cette

modélisation n'en a pas moins l'avantage d'exister et d'avoir servi de base à une grande majorité des études formelles effectuées en connexionnisme.

Un neurone formel effectue donc tout d'abord une somme des valeurs d'entrée, ces valeurs étant pondérées par des poids synaptiques qui sont définis lors d'un processus d'apprentissage et qui permettent au neurone de se spécialiser en fonction d'une erreur calculée, généralement, en couche de sortie d'un réseau de telles cellules. Cette étape de sommation correspond à la collecte et à l'intégration de l'information. Cette valeur d'activation, interne, est ensuite modifiée par la fonction de transfert non linéaire qui permet d'obtenir la valeur effective de l'activation de la cellule, valeur qui sera répercutée en sortie.

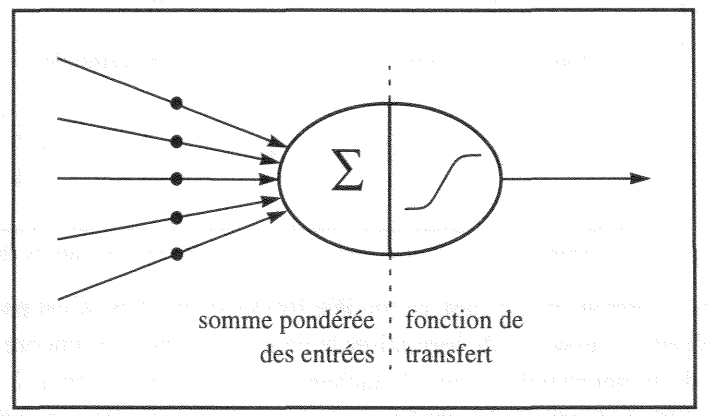


Figure 2.6 : Le neurone formel de McCulloch et Pitts (d'après [mcculloch43]).

La fonction de transfert se doit d'être non linéaire mais cette non linéarité peut être exprimée de différentes façons. Nous présentons ci-dessous un ensemble de telles fonctions non linéaires utilisées dans le domaine du connexionnisme.

La figure 2.7 présente des fonctions binaires à seuil qui correspondent à un mécanisme tout ou rien. Parmi celles-ci se trouve la fonction utilisée par McCulloch et Pitts.

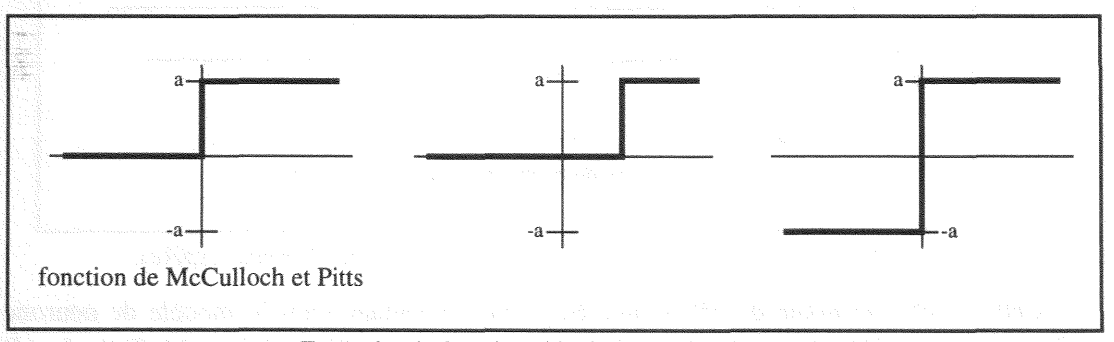


Figure 2.7 : Exemples de fonctions binaires à seuil (d'après [buniet91]).

La figure 2.8 présente elle une généralisation de ces fonctions : les fonctions à saturation ou linéaires à seuil.

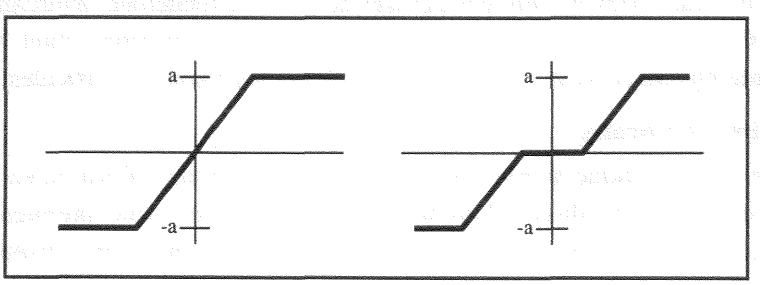


Figure 2.8 : Exemples de fonctions à saturation (d'après [buniet91]).

La figure 2.9 présente, enfin, les fonctions non linéaires dérivables qui sont actuellement utilisées puisque la dérivabilité de ce type de fonctions est aujourd'hui une condition nécessaire au processus d'apprentissage.

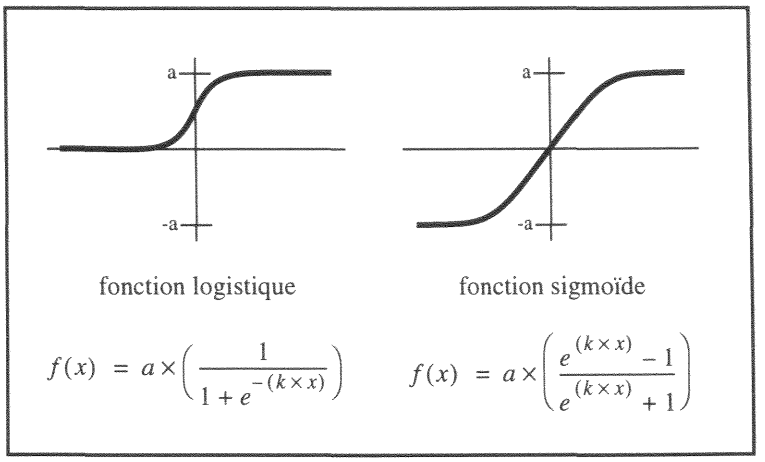


Figure 2.9 : Exemples de fonctions non linéaires dérivables (d'après [buniet91]).

Comme nous le verrons plus avant, ce modèle formel de neurone n'est pas le seul modèle existant. D'autres modélisations essaient de formaliser le neurone de manière encore plus exacte vis-à-vis des connaissances de la neurobiologie en s'attachant, tout particulièrement, à modéliser de manière la plus exacte possible la forme de la courbe produite par la propagation sous forme d'impulsion, *spike*, de l'information au sein des cellules. Cette courbe ne présente pas un comportement de type tout ou rien. Le potentiel d'une cellule est transmis par une décharge rapide de l'impulsion suivie d'un retour plus lent à la norme. La puissance d'un *spike* s'exprime en millivolts (figure 2.10).

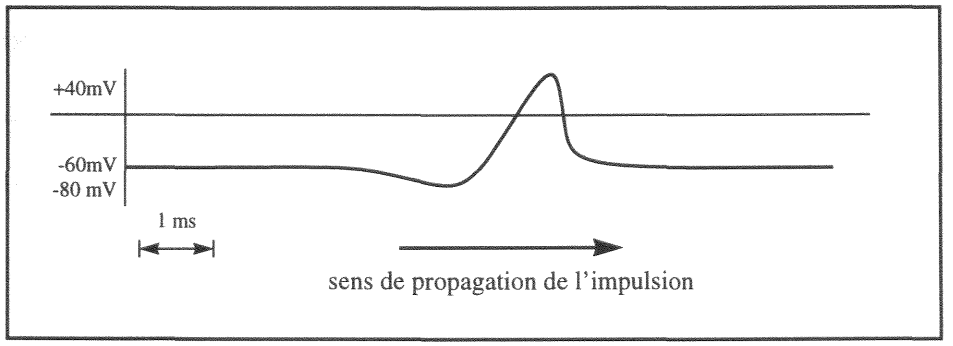


Figure 2.10 : Courbe de propagation d'une impulsion neuronale (*spike*).

Cette courbe nécessite de définir une étape supplémentaire dans le modèle de neurone formel. Cette étape supplémentaire rend bien sûr plus complexe le modèle défini par McCulloch et Pitts mais permet d'obtenir une capacité supplémentaire de représentation dans l'échelle temporelle [usher95]. Nous reviendrons sur ces capacités lors du chapitre 5.

Les neurones du cortex ne sont pas agrégés de manière totalement anarchique dans le cerveau. Ils sont organisés selon un certain nombre de fonctions, ces fonctions étant elles-même associées à différentes régions du cerveau : les aires cérébrales et les colonnes corticales.

2.2.2/ Les aires cérébrales

Une autre caractéristique architecturale du cerveau est sa spécialisation en aires qui sont autant de régions du cerveau spécialisées dans un type de traitement. Une première répartition est due à Brodmann qui, en 1909, a divisé le cerveau en une cinquantaine d'aires disparates.

Une autre classification, fonctionnellement plus homogène, a été définie dans [burnod88] où 24 aires ont été isolées. Elles peuvent être regroupées en trois catégories [alexandre90] que sont les aires

sensorimotrices, les aires associatives et les aires frontales.

Les aires sensorimotrices sont responsables des interactions avec l'environnement et permettent donc de recevoir de l'information et d'agir sur le milieu. La réception de l'information se fait de manière isotopique dans chacune de ses aires qui sont ainsi responsables de la compréhension immédiate du monde. Il est ainsi question de sonotopie pour l'aire auditive et de rétinotopie pour l'aire visuelle. L'isotopie permet d'identifier une caractéristique intéressante du cerveau : deux stimuli de même nature et de forme proche seront codés dans deux régions qui sont différentes mais pourtant spatialement proches. Certaines des aires sensorielles sont également responsables de la réception des informations hormonales. Les aires motrices agissent, elles, sur les muscles et permettent donc d'adapter notre attitude physique à notre volonté mentale, consciente ou non.

Les aires associatives permettent d'effectuer le lien entre les aires sensorimotrices et de relier, par exemple, la connaissance visuelle à la connaissance auditive, engendrant ainsi une connaissance distribuée et "multimédia".

Les dernières aires, apparues le plus tardivement dans l'histoire de l'évolution, sont les aires frontales qui permettent des traitements de haut niveau tels que l'élaboration des plans d'action, le raisonnement et la hiérarchisation.

Les aires corticales sont reliées entre elles par des connexions qui constituent un réseau. Ce réseau peut être présenté de deux manières : sous une forme fermée [burnod88] ou sous une forme ouverte [alexandre90]. Ces modélisations permettent de visualiser les voies de circulation de l'information entre les aires corticales.

Ces différentes aires sont très importantes et même si la connaissance de la topographie du cerveau permet de savoir où s'effectuent certains types de traitement, cette connaissance ne permet pas, en revanche, de savoir comment peuvent s'effectuer ces traitements. Une granularité supplémentaire existe qui permet de mieux appréhender les moyens mis en œuvre pour ces traitements : la colonne corticale.

2.2.3/ La colonne corticale

Une colonne corticale correspond à un niveau de granularité intermédiaire entre les aires corticales et les neurones. Ce modèle a été proposé pour la première fois de manière formelle par Lorento de No en 1938 après que celui-ci ait effectué des recherches sur la connectivité dans le cortex. Cette "unité de mesure" est intéressante car elle permet de comprendre comment peuvent s'organiser les neurones, à un niveau local du cortex, pour effectuer des traitements.

Ce type d'architecture a, par la suite, été étudié de manière plus approfondie. Ainsi, [szentagothai73] en fournit une étude détaillée et [mountcastle78] généralise le concept après l'avoir étudié dans le cortex sensoriel.

Un schéma de colonne corticale, reproduit d'après [szentagothai73], est fourni à la figure 2.11. Dans ce schéma sont regroupés plusieurs types de neurones. Ces types et leur répartition caractérisent chacun des 6 différents niveaux d'une colonne corticale. Ces couches, d'épaisseur variable [bullier83], assurent chacune une partie de la chaîne des traitements de la colonne. Cette épaisseur n'en est pas moins très faible puisqu'elle est estimée à $2{,}5\ 10^3$ µm.

Toutes les modélisations citées jusqu'à présent restent cependant d'un haut niveau conceptuel et restent floues sur certaines des caractéristiques. Cette constatation justifie le choix de [alexendre90] et [guyot90] pour le choix du modèle exposé dans [burnod88] qui, seul, permet une implantation calculatoire. D'autres modèles se rapprochant expressément ou non de ce concept de colonne ont par ailleurs été développé [ingber81], [lumer92]. Nous reviendrons plus tard sur ce modèle (cf. chapitre 6, paragraphe 6.2.6).

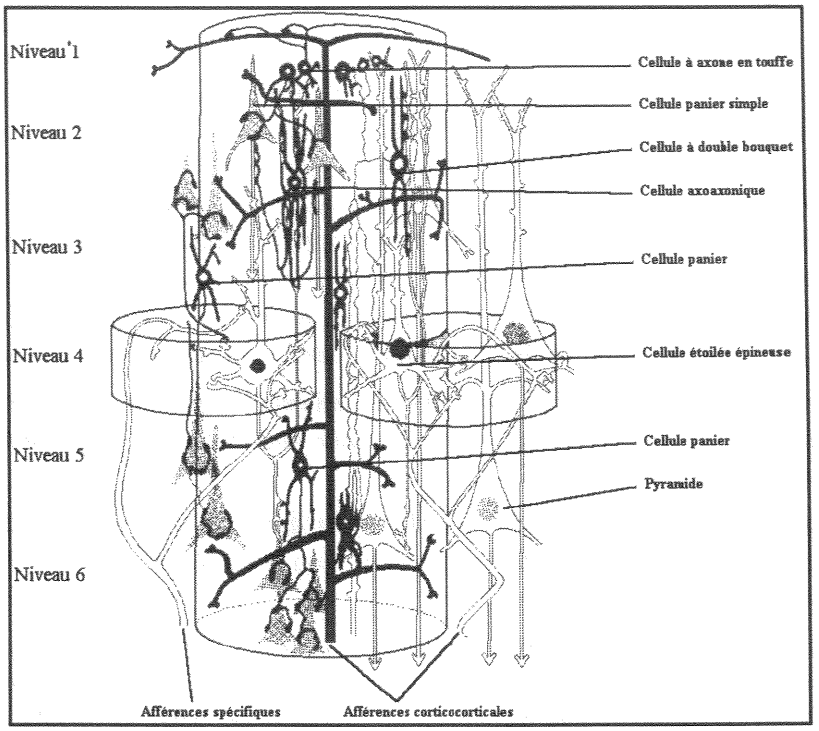


Figure 2.11 : Schéma d'une colonne corticale (d'après [szentagothai73]).

2.3/ Modélisation connexionniste

Même si les grands principes descriptifs du fonctionnement du cerveau sont aujourd'hui partagés, cette description n'est pas encore figée. La science continue en effet à progresser dans le domaine de la neurobiologie au rythme des recherches et de l'arrivée de matériels permettant d'explorer toujours plus l'infiniment petit.

Cette progression continue permet de découvrir de nouveaux principes qui sont, ou ne sont pas, répercutés dans le monde de la modélisation formelle qu'est le connexionnisme. Cette prise en compte, à des degrés variables, de la connaissance neurobiologique et la volonté de développer ou non des modèles dans un souci de mimétisme biologique ou selon un besoin plus en rapport avec l'ingénierie provoque la mise en place d'architectures formelles diverses. L'exploration du domaine des possibles provoque par ailleurs une explosion du nombre de modèles, l'intérêt ou non d'une technique particulière pouvant ainsi être mesuré.

Cette grande diversité des modèles connexionnistes n'en permet pas moins d'effectuer une taxonomie en grandes classes selon des principes de base reconnus. Il est ainsi possible de distinguer les modèles statiques des modèles dynamiques en fonction du principe de récurrence. Il est également possible de distinguer les modèles supervisés des modèles non supervisés en fonction de la méthode d'apprentissage.

Nous allons maintenant donner une liste d'architectures connexionnistes en fonctions de critères que nous énoncerons au fur et à mesure de l'exposé.

2.4/ Modèles connexionnistes statiques

Les modèles connexionnistes statiques sont actuellement les plus utilisés. Cette présence majoritaire dans le domaine est justifiée par l'existence d'algorithmes d'apprentissage efficaces dont la convergence est presque totalement assurée, à condition toutefois de respecter certaines règles et d'utiliser certaines heuristiques développées au cours des recherches effectuées.

La principale caractéristique d'un modèle statique est qu'il permet de classer des formes indépendantes du temps et d'une quelconque évolution. La forme à classer à un instant t est donc

jugée totalement indépendante des formes classées lors d'instants précédents. Les réseaux statiques se caractérisent donc avant tout par l'absence totale de récurrence au sein de leur architecture. Un réseau statique effectue une fonction d'approximation non linéaire qui lui permet de classer des données dans des espaces multidimensionnels aux frontières non linéaires. Cette dernière caractéristique doit cependant être considérée avec circonspection car un neurone seul ne peut pas séparer un espace autrement que linéairement et, ce, de manière plus ou moins fine en fonction des caractéristiques de la fonction non linéaire choisie. Seule une architecture complexe permet d'obtenir des séparations dans l'espace aptes à résoudre des tâches complexes. Les réseaux connexionnistes statiques seront donc définis sur deux ou trois couches successives, permettant d'effectuer des séparations non linéaires.

C'est ce besoin de complexité qui a d'ailleurs été à l'origine de l'abandon du paradigme connexionniste pendant plus d'une décennie. L'algorithme d'apprentissage initialement utilisé avec ces réseaux ne permettait pas d'ajuster plus d'une couche de poids. Or un réseau ayant cette architecture est dans l'incapacité de résoudre des problèmes non linéairement séparables. Un exemple très simple d'un tel problème est la fonction logique ou exclusif, XOR en jargon anglo-informatique. La découverte de ce problème simple et pourtant insoluble [minsky69] avec la théorie alors en place a stoppé net les recherches dans le domaine.

a	b	a XOR b
0	0	1
0	1	0
1	0	0
1	1	1

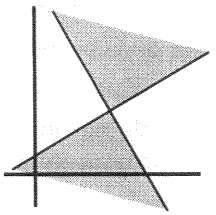


Figure 2.12 : La fonction XOR et la présentation schématique du graphe des régions d'une fonction non linéairement séparable.

Il a alors fallu attendre une quinzaine d'années pour voir apparaître une méthode d'apprentissage capable de circonvenir ce problème [lecun85]. Ce type de réseau a alors connu d'importants développements et de nombreuses architectures ont été définies pour résoudre des problèmes variés.

2.4.1/ Les perceptrons multicouches

Les perceptrons multicouches sont les réseaux à la base des méthodes connexionnistes. Ils sont, en effet, les plus employés et les plus étudiés [lippmann87], [hush93], [jodouin90]. Deux abréviations anglaises sont utilisées dans la littérature pour les nommer : *MLP* pour *Multi Layer Perceptrons* et, de manière un peu abusive, *ANN*, *Artificial Neural Networks*.

2.4.1.1/ Architecture

Un perceptron multicouche est composé de plusieurs couches de neurones et de connexions (figure 2.13). Ce nombre est au moins égal à deux, signifiant ainsi que le réseau possède deux couches de poids connexionnistes, une couche de sortie et une couche cachée. Le nombre de couches cachées détermine la complexité des frontières des différents sous-espaces que le réseau pourra représenter [lippmann87]. La complexité de l'approximation est également déterminée par le nombre de neurones de chaque couche puisque ce nombre détermine le nombre maximal d'informations que le réseau peut extraire du signal traité [makhoul89], [murata92], [priel93]. La couche d'entrée, correspondant le plus souvent à un vecteur de données issu d'une phase de prétraitement, n'est pas véritablement considérée comme appartenant au réseau.

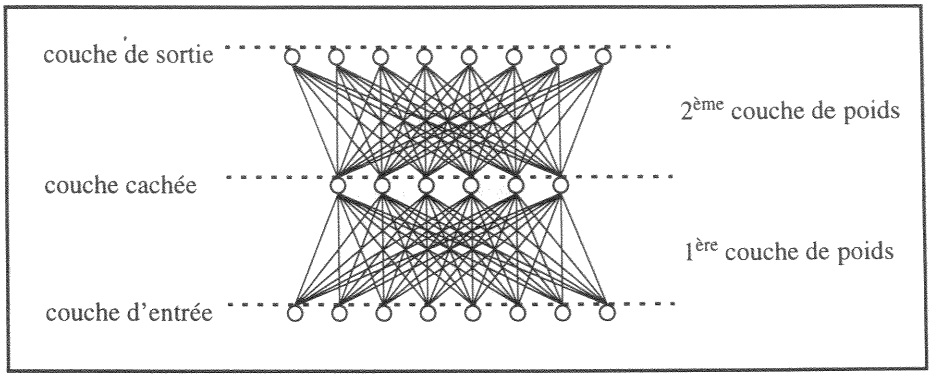


Figure 2.13 : Schéma d'un réseau connexionniste statique à deux couches.

La propagation de l'activité au sein de ces réseaux est très simple à calculer : il suffit d'effectuer un parcours itératif du réseau. Ainsi, pour chaque couche du réseau, de l'entrée vers la sortie, et pour chaque neurone de la couche considérée, l'activité est calculée en fonction de l'équation :

$$x_j = f\left(\sum_{i=1}^{N} w_{ji} x_i\right) \qquad \text{(Éq. 2.4)}$$

Dans cette équation, qui correspond à la figure 2.6, x_i représente l'activité du neurone i tandis que w_{ji} représente la valeur de la connexion synaptique entre les neurones i et j, N est le nombre d'unités afférentes au neurone x_j. Toute la connaissance acquise par le réseau lors de la phase d'apprentissage de la tâche est représentée par les valeurs des connexions synaptiques. Ces dernières représentent une solution possible permettant d'effectuer une transformation des données observées en entrée selon les valeurs qui ont été imposées par le concepteur sur les différents neurones de la couche de sortie lors de la phase de mise au point.

Cependant, les connexions ne sont pas uniques vis-à-vis d'une tâche et un réseau connexionniste n'utilisera que très rarement un ensemble de poids optimal. Un réseau connexionniste comme le perceptron est, cependant, un approximateur universel [lippmann87] qui permet, en théorie, d'implanter toute fonction mathématique non récurrente définie sur l'ensemble des réels.

2.4.1.2/ Applications possibles des perceptrons multicouches

Comme nous l'avons dit précédemment, les perceptrons multicouches sont les architectures connexionnistes les plus utilisées. De nombreux problèmes peuvent être résolus par leur intermédiaire.

Les applications qui nous intéressent le plus sont, bien sûr, celles en relation avec la reconnaissance de la parole. Le domaine n'est cependant pas réducteur car la très grande majorité des principes d'application des réseaux de neurones peuvent y être retrouvés.

Certaines applications des réseaux connexionnistes sont évidemment spécifiques à la parole. Les perceptrons peuvent ainsi servir à la segmentation de la parole [bendiksen90], [depuydt90], [cohn91], [ghiselli91], à la reconnaissance de phonèmes [harrison89], [franzini89], [leung90], [anderson91] ou de reconnaissance de mots [zhu90], [morgan91b]. Les perceptrons peuvent également être mis en œuvre dans des problèmes de reconnaissance du locuteur [hattori92] ou d'amélioration de la qualité d'un signal bruité [tamura90], [ohkura91], [sorensen91a], [sorensen91b], [trompf92]. Les perceptrons peuvent même apprendre à parler, la synthèse de parole s'effectuant à partir d'une séquences de lettres [sejnowski87] ou à partir du langage des sourds et muets par décodage des signes de l'alphabet captés par un gant désignateur [fels94].

Les neurones permettent également de faire de la prédiction. Il est possible de produire le rayonnement solaire [fessant94] tout autant que la consommation électrique journalière [muller94]. Certains essaient même de prévoir les cours de la bourse... Il est également possible de faire de la

prédiction en parole. La prédiction d'une trame de parole à partir d'autres qui lui sont antérieures permet de comparer le résultat de la prédiction à la trame effectivement observée, l'erreur pouvant être traitée lors d'une étape postérieure d'alignement temporel. Cette technique est aussi bien utilisée pour la prédiction de phonèmes avec les *Linked Predictive Neural Network* [tebelskis90], [tebelskis91] que pour la prédiction de mots avec les *Neural Prediction Model* [iso90], [iso91].

Nous avons, lors du paragraphe 1.7, parlé des différentes méthodes permettant de représenter ou de synthétiser l'information présente dans le signal de parole. Cette synthèse est également possible à réaliser avec des réseaux connexionnistes. Dans ce cas, l'idée est de réaliser un réseau connexionniste possédant autant d'unités en entrée qu'en sortie. La ou les couches cachées sont par contre constituées de moins de neurones que ne le sont l'entrée et la sortie. Ceci oblige le réseau de neurones à synthétiser l'information de manière plus ou moins optimale selon le nombre de neurones alloués. Mais le processus n'est pas totalement symétrique : alors que la première couche de poids synthétise l'information de manière approximative, la dernière couche de poids utilise les informations en couche cachée pour recréer une des formes apprises dans un ensemble restreint. Cette technique [nakamura91] permet donc d'obtenir une trame qui respecte beaucoup plus une forme phonétique choisie par le concepteur que la trame qui était disponible en entrée, permettant ainsi de réduire le nombre des erreurs de classification.

Le domaine de la reconnaissance d'images est également un domaine d'application des réseaux connexionnistes comme le perceptron multicouche. Si de nombreuses tâches peuvent être résolues [wurtz94], l'application qui nous semble le plus en rapport avec le domaine de la reconnaissance de la parole est celle de la reconnaissance des chiffres manuscrits [lecun89a] [lecun89b] [doré90]. Les architectures employées donnent de très bons résultats dans le domaine industriel et sont très proches de celles employées en reconnaissance de phonèmes. Il existe par ailleurs un domaine du traitement de l'image très proche de celui de la parole : les modèles articulatoires dont nous avons déjà parlé au paragraphe 1.3.2. Ceux-ci peuvent être modélisés à l'aide de perceptrons donnant les différents coefficients du modèle articulatoire, permettant de reconstruire une image du conduit vocal, en fonction d'un spectre de signal de parole [kobayashi91].

Enfin, signalons qu'il existe tout un domaine du connexionnisme se fondant sur des données artificielles ou pseudo-réelles. Ces données sont principalement utilisées pour étudier les capacités d'apprentissage des réseaux et constituent les cas d'école (*toy problems*) du connexionnisme.

Un état de l'art des applications des réseaux de neurones en reconnaissance automatique de la parole pourra être trouvé dans [lippmann87] ou [waibel91].

2.4.1.3/ Extensions des perceptrons avec la notion de poids partagés

Les poids au sein d'un perceptron multicouche sont normalement totalement indépendants les uns des autres pendant et après l'apprentissage. Chaque connexion est donc associée à une valeur qui lui est propre et l'information représentée par un neurone lui est spécifique.

Cette liberté peut être contrainte pour aboutir à des réseaux dont certaines unités d'un même niveau (d'une même couche) représentent la même information. Ceci permet de rechercher une même caractéristique tout au long du flux d'information disponible, dans le cas d'un signal bidimensionnel, ou dans tout l'espace de représentation si le signal a plus de deux dimensions. Cette technique a été largement utilisée après sa présentation dans [waibel88] et [waibel89]. La principale application qui en a été faite est la reconnaissance de la parole et plus particulièrement des phonèmes.

Pour la mise en place du partage des poids, il est nécessaire d'effectuer une moyenne sur les poids arrivant à chaque ligne ou colonne de neurones des différentes couches du réseau après chaque phase de rétropropagation de l'erreur. Ce principe a tout d'abord été utilisé pour la reconnaissance de phonèmes [waibel89], [hataoka91] mais peut également être mis en œuvre pour l'extraction des caractéristiques phonétiques [bimbot90], [bimbot90] ou pour la reconnaissance du locuteur

[bennani91].

Ce principe de partage des poids du réseau peut être poussé très loin et le nombre de couches intermédiaires peut être augmenté pour obtenir diverses spécialisations tout autant que peuvent être augmentés le nombre et la forme des partages de poids. Ainsi, alors que le TDNN, *Time Delay Neural Network*, effectue le partage des poids sur l'ensemble d'une ligne ou d'une colonne, il est possible de mettre en place un partage sur des sous parties de ces lignes ou colonnes. Différents types de partages ont été proposés dans [altosaar91], [komori91a], [sawai91a] où les neurones sont regroupés par sous-plaques dans chaque couche, permettant de coder localement des informations de nature géométrique. Dans le même élan, il est possible de multiplier le nombre de couches cachées. Le modèle présenté dans [waibel89] est constitué de deux couches cachées et le flux des informations à travers le réseau suit un seul chemin. Certains études ont été menées suivant un principe d'augmentation du nombre des couches cachées, en divisant le flux de données en plusieurs directions parcourant chacune un sous-ensemble des couches cachées. Ce principe permet de subdiviser les couches cachées en fonction du locuteur ou de la classe phonétique. Ce principe a tout d'abord été mis en œuvre dans [sawai91b], avant que n'apparaisse un "monstre" connexionniste cherchant à traiter tous les cas séparément, créant ainsi de nombreuses couches cachées organisées en flux distincts [nakamura92].

D'autres développements ont également été réalisés en relation avec d'autres techniques mathématiques ou de reconnaissance des formes. Il est ainsi possible d'utiliser un TDNN comme générateur d'une séquence descriptive du signal, cette séquence étant ensuite examinée par analyseur de grammaires à contexte libre. Cet analyseur permet de contraindre les réponses et permet ainsi d'obtenir de meilleurs résultats [miyatake90], [sawai91c]. Il est également possible d'utiliser un TDNN en conjonction avec un système expert, les informations extraites venant se fondre au milieu d'autres obtenues par des méthodes de traitement du signal [komori90], [komori91b]. À un niveau architectural plus fin, il est possible de modifier l'architecture TDNN de nombreuses manières. La procédure d'apprentissage peut être modifiée pour prendre en compte des informations floues [komori92], la fonction de transfert non linéaire peut être modifiée [takami91] ou l'architecture du neurone peut se voir mise en accord avec la technique des fonctions à base radiale [berthold94], plus proches de la théorie des probabilités.

L'apprentissage dans les réseaux comme le TDNN mérite une attention particulière. L'avantage de ce réseau est sa capacité à prendre en compte des informations dans un plan temporel fini qui peut cependant être assez vaste. Les poids peuvent être contraints par la simple calcul de la moyenne des poids, qui permet un apprentissage somme toute assez grossier. Plusieurs techniques d'apprentissage essaient de définir la position des délais de manière plus fine, de manière à obtenir une meilleure représentation temporelle dans les couches internes. Parmi ces algorithmes se trouvent l'algorithme Tempo 2 [bodenhausen91a], [bodenhausen91b] et une amélioration possible de celui-ci développée dans [rander92].

Les applications du TDNN ne se limitent pas à la seule reconnaissance des phonèmes. Une extension de ce réseau permet également de faire de la reconnaissance de mots. L'architecture du TDNN n'est pas intrinsèquement modifiée mais plutôt étendue par l'ajout de deux couches supplémentaires ce qui permet d'obtenir une nouvelle architecture de réseau baptisée *MS-TDNN*, *Multi-State Time-Delay Neural Network*. Le TDNN conserve son rôle initial de classification de phonèmes, l'ajout de la structure de reconnaissance de mots se faisant après que le TDNN ait appris à correctement faire sa tâche. La couche de sortie de la nouvelle structure permet de classer les mots tandis que la couche cachée, située entre la sortie originale du TDNN et la sortie du MS-TDNN, assure, après apprentissage, la liaison entre la classification au niveau du phonème et la classification au niveau du mot. Cette architecture a été exposée de manière étendue dans [lang90], [haffner91a], [haffner91b], [haffner92a], [haffner92b].

La parole, comme nous venons de le voir tout au long de ce paragraphe, est l'application qui a le plus mis en œuvre le concept de partage des poids. Il est cependant possible de l'appliquer à la reconnaissance de caractères manuscrits avec un réseau où les poids partagés permettent de représenter des contraintes géométriques communes à l'ensemble de la plaque d'entrée, plaque où sont fournies les représentations discrètes des caractères manuscrits à reconnaître [lecun89b].

Le concept de poids partagés peut également être mis en œuvre avec les modèles à auto-organisation que nous allons voir maintenant [webber94].

2.4.2/ Modèles à auto-organisation

Les modèles à auto-organisation ne suivent pas les même principes que les perceptrons multicouches. Ils permettent d'effectuer une classification en suivant une règle générale de représentation des connaissances dans le cerveau : l'isotopie. Ils peuvent être rapprochés des principes de quantification vectorielle et de matrices de corrélation [kohonen72].

L'auto-organisation permet d'obtenir un réseau qui répond aux stimuli qui lui sont présentés en entrée suivant une classification qui n'est pas donnée a priori par le concepteur. Ainsi, à l'inverse des perceptrons, la répartition des activations en fonction de l'entrée ne peut pas être connue avant la fin de l'apprentissage. La répartition des stimuli sur la couche de sortie, qui est généralement une carte, un espace à deux dimensions, permet d'obtenir une répartition des formes par proximité de leurs caractéristiques. Tous les stimuli sont donc organisés suivant une sorte de continuum perceptif, deux formes voisines sur la carte auto-organisée pouvant être considérées comme proches dans l'espace des paramètres d'entrée suivant la distance qui a été utilisée lors du processus d'apprentissage. Cette proximité de formes semblables constitue le principe d'isotopie et peut s'appliquer, entre autres, à la vision ou à l'audition.

Le modèle de carte auto-organisatrice le plus célèbre est la carte de Kohonen [kohonen84], [kohonen87]. Sa principale application est la reconnaissance de la parole [kohonen88], [brauer89], [torkkola91], [kangas92]. Signalons qu'il semble difficile d'appliquer le principe d'isotopie à la vision grâce à ce type de méthodes mais le problème est plus matériel, du fait du nombre de connexions à mettre en œuvre, que théorique [ritter89], [webber94].

Une carte auto-organisée se présente sous la forme qui est donnée à la figure 2.14. Le réseau est constitué de trois parties : la couche d'entrée, qui reçoit les valeurs issues de la phase de prétraitement, la couche des connexions, qui permettra de calculer la distance entre une forme d'entrée et un jeu de poids considéré, et la couche de sortie, qui permet de faire ressortir l'unité vainqueur de la plaque.

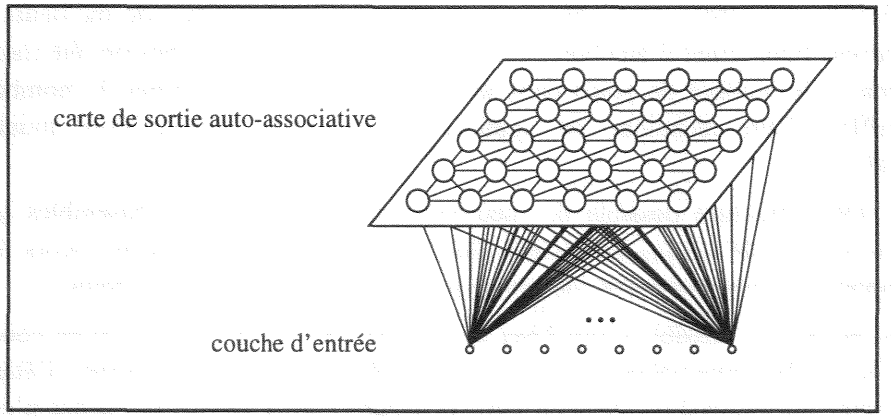


Figure 2.14 : L'architecture générale d'une carte auto-organisatrice (d'après [kohonen88]).

Le principal défaut des modèles auto-organisés comme la carte de Kohonen est, justement, la non supervision de la phase d'apprentissage. La répartition des formes sur la plaque de sortie se fait selon

un processus non-supervisé qui ne tient absolument pas compte des différentes classes existantes et donc de la symbolique associée au signal d'entrée. La prise en compte de la variabilité du signal en fonction de la classe associée peut donc rapidement devenir un problème. La phase postérieure à l'apprentissage est la détermination de la symbolique associée à chaque unité de la couche de sortie. Le corpus d'apprentissage est donc parcouru à nouveau pour savoir à quel(s) symbole(s) est associé une unité. Ce processus de réassociation peut permettre d'établir une carte ne possédant pas trop d'unités polysémantique [kohonen88]. Mais l'apprentissage d'une classification de sons très proches, tels que les sonantes ou les occlusives, fournit parfois des résultats presque inexploitables [buniet91]. Il n'y a alors plus d'autre choix que de réaliser un nouvel apprentissage en espérant obtenir une meilleure qualité.

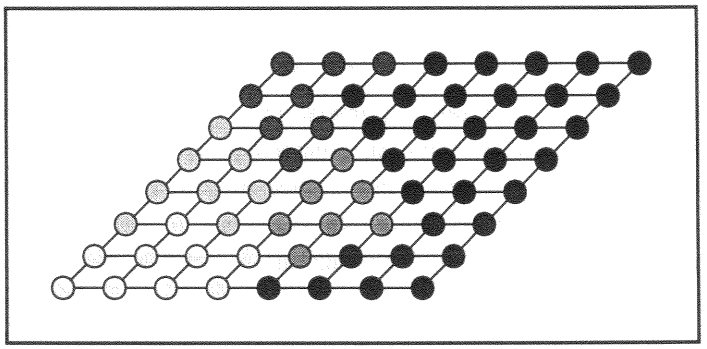


Figure 2.15 : Exemple de répartition des classes sur la couche de sortie d'une carte auto-organisatrice.

Le modèle de la carte de Kohonen peut être étendu dans la dimension temporelle pour la reconnaissance de séquences de signaux de parole [kangas91], [kangas94] ou de musique [carpinteiro96]. Ce réseau de Kohonen peut également recevoir une extension architecturale par l'adjonction d'un mécanisme de récurrence local et d'un mécanisme d'inhibition latérale dans la couche de sortie [kohonen93].

2.4.3/ Autres architectures

D'autres architectures connexionnistes statiques peuvent être définies. Le perceptron multicouche n'est en effet pas le seul arrangement possible pour les éléments simples que sont les neurones.

Il est, par exemple, possible d'organiser les neurones en structures arborescentes pour combiner la méthode connexionniste aux techniques de classifications en arbre [stromberg91]. Cette structure permet de diminuer la complexité des architectures connexionnistes et d'augmenter les capacités de classification des arbres. La technique n'en obtient pas pour autant de meilleurs résultats que les perceptrons. Pour tenter d'améliorer ces résultats, certaines recherches ont été menées pour réduire plus encore le nombre de poids dans les arbres pour rendre minimal le nombre de connexions [sankar91a]. Cette technique est applicable à la parole comme l'ont montré [rahim92] et [sankar91b].

Une autre structure possible des neurones est la formation en ensembles [hansen90]. Cette structure permet de minimiser l'erreur globale du réseau en utilisant plusieurs réseaux similaires ayant appris la même tâche avec des poids initialisés à des valeurs différentes.

Il est également possible d'organiser les neurones en cascade plutôt qu'en couche de manière à utiliser le nombre minimal de neurones [fahlman90]. Dans cette architecture, l'état initial du réseau est restreint aux unités d'entrée et de sortie et aucun neurone en couche cachée n'est défini. La mise en place de l'architecture du réseau est faite pendant la phase d'apprentissage, phase pendant laquelle des unités cachées sont ajoutées, une à une, au réseau. Cette procédure ajoute donc le nombre minimal de neurones nécessaire à l'apprentissage de la tâche demandée. Cette procédure peut, par ailleurs, être réalisée avec une précision finie [hoehfeld91].

L'activité d'un neurone peut également être contrainte par la mise en place, à l'entrée du neurone, d'afférences contrôlant l'entrée du flux d'information dans le neurone. Ce type d'architecture a conduit à la mise en place des architectures connexionnistes à propagation guidée puis à détection de coïncidences [escande92], [laine94], [nioche94].

Il est possible d'employer le concept de réseau de taille minimale en mixant les paradigmes d'apprentissage supervisé et d'apprentissage non supervisé. Ainsi, [alpaydin91] propose un réseau dont la première couche est définie par apprentissage non supervisé tandis que la deuxième est définie par apprentissage supervisé. Une troisième couche, la couche de sortie, permet de définir l'ensemble des classes à apprendre. De plus, alors que la première couche a une architecture fixe, la deuxième est définie selon une architecture variable qui permet d'ajuster la représentation effectuée par la première couche à la représentation qui est fixée en couche de sortie.

Les cartes auto-organisées tirent leur origine des mémoires associatives et des représentations distribuées [kohonen72]. Ces notions peuvent être grandement étendues par l'emploi d'algèbres matricielles et convolutionnelles [plate91], [plate94]. Les opérateurs mis en œuvre dans ce cas sont de bien plus haut niveau que les simples sommes pondérées.

Les réseaux de neurones sont des structures effectuant une analyse répartie de la connaissance qui leur est fournie en entrée. Cette analyse répartie est ensuite synthétisée pour fournir une réponse appropriée au problème posé. Il n'y a pas de structure a priori pour l'analyse et les poids synaptiques, initialisés aléatoirement, définissent cette structure par apprentissage. D'autres méthodes d'analyse du signal sont également à la recherche des "bons coefficients". La méthode de la transformée en ondelettes impose ainsi de rechercher les coefficients qui permettent un bon compactage approximatif d'un signal jusqu'à une échelle donnée [tewfik91]. L'idée est donc apparue de mixer les principes du connexionnisme avec la théorie des ondelettes [zhang92], [zhang93], [mousset94]. Dans de tels réseaux, les neurones représentent une partie de la transformée en ondelettes tandis que les poids connexionnistes représentent les coefficients de ces ondelettes "élémentaires". L'apprentissage permet de déterminer une bonne approximation des coefficients par itération, permettant d'obtenir un réseau d'ondelettes.

Notons qu'il est également possible de totalement redéfinir la théorie associée à un neurone, la redéfinition de cette unité allant au delà des modifications qui sont faites lors de la mise en place des réseaux d'ondelettes. C'est par exemple le cas avec les réseaux de Clifford [pearson94] qui utilisent des neurones définis selon l'algèbre de Clifford. Ceci permet de passer d'espaces réels ou complexes [vaucher96] à des espaces d'entités multidimensionnelles.

Tous ces modèles connexionnistes pourraient très bien ne servir à rien si aucune méthode d'apprentissage ne permettait de définir correctement les poids des connexions.

2.4.4/ Apprentissage dans les modèles statiques

L'apprentissage est la pierre d'achoppement de tout modèle connexionniste comme de tout système de classification en générale. Cet apprentissage prend cependant ici un sens particulier puisqu'il a été la cause, pendant de longues années, de la mise en sommeil des recherches dans ce domaine.

2.4.4.1/ Apprentissage supervisé

L'apprentissage dans les réseaux connexionnistes du type des perceptrons multicouches se fait grâce à la méthode de la rétropropagation du gradient d'erreur [lecun85]. Cette méthode permet de rétropropager une erreur exacte sur plusieurs couches de connexions et d'adapter ces connexions par ajustement pour minimiser l'erreur. Le calcul de ce gradient se fait de deux manières en fonction de la position de la connexion dans le réseau. Soit cette connexion appartient à la dernière couche de connexions du réseau, auquel cas l'erreur utilisée est celle qui est calculée en sortie, soit cette connexion appartient à une couche cachée de connexions et l'erreur utilisée est une synthèse des

erreurs de la couche de neurones en aval.

La formule générale du gradient d'erreur est donnée par la formule :

$$\Delta w_{i,j} = -\alpha \cdot \frac{\partial E}{\partial w_{i,j}} = \alpha \cdot \left(-\frac{\partial E}{\partial y_i}\right) \cdot \frac{\partial y_i}{\partial w_{i,j}} \qquad \text{(Éq. 2.5)}$$

Dans cette équation, w_{ij} représente la connexion synaptique allant du neurone j au neurone i, y_i représente l'activité du neurone i et E représente l'erreur commise en sortie du réseau. La formule générale permet d'obtenir deux équations. Dans le cas où le gradient est calculé par rapport à la couche de sortie du réseau, l'équation 2.5 devient :

$$\Delta w_{i,j} = \alpha \cdot y'_i \cdot y_j \cdot (y_i - d_i) \qquad \text{(Éq. 2.6)}$$

Dans cette équation, d_i représente la valeur désirée en sortie du neurone i et y' représente la dérivée de y. Dans le cas où le gradient d'erreur est calculé dans une couche interne au réseau et où aucune erreur n'est directement disponible, il faut resynthétiser celle-ci à partir de l'erreur des couches supérieures. L'équation 2.5 devient donc :

$$\Delta w_{i,j} = \alpha \cdot y'_i \cdot y_j \cdot \left(\sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot y'_k \cdot w_{k,i} \right) \qquad \text{(Éq. 2.7)}$$

Le δ_k représente ici l'erreur partielle commise dans un neurone k se trouvant en aval de i. Une fois le gradient d'erreur calculé pour toutes les connexions du réseau, celles-ci peuvent être mises à jour.

Une présentation approfondie de la théorie associée à cette méthode d'apprentissage pourra être trouvée dans [minoux89]. La dérivation complète des équations d'apprentissage pour les perceptrons multicouches est fournie en annexe 1.

Il est important de se rappeler que la simple équation de rétropropagation du gradient d'erreur ne suffit pas à assurer une bonne convergence du réseau connexionniste. La présence dans l'équation 2.5 d'un coefficient d'apprentissage, α, est une première heuristique mise en œuvre. D'autres sont possibles comme l'apprentissage en ligne [biehl94], l'adaptation en cours de processus du coefficient d'apprentissage ou l'utilisation d'un gradient conjugué. Ces dernières méthodes, et d'autres, sont détaillées dans [schiffmann92] et [jervis93]. Il est également possible de profondément modifier l'algorithme pour mettre en œuvre la technique de maximisation de l'information commune (*MMI, Maximum Mutual Information*) [niles90], [fry95].

Cette recherche d'heuristiques toujours plus efficaces peut cependant conduire à des méthodes dont la capacité de convergence est loin d'être prouvée [jurik91].

2.4.4.2/ Apprentissage non supervisé

L'apprentissage dans les modèles non supervisés, comme le réseau de Kohonen, se fait grâce à l'emploi d'une fonction de voisinage. Après initialisation aléatoire des poids, comme précédemment, une forme est présentée en entrée du réseau. L'apprentissage entre alors dans une phase de compétition : c'est la cellule dont le potentiel d'activation est le plus fort en fonction de l'entrée qui est choisie comme vainqueur. Cette activation est calculée en fonction d'une distance et sera d'autant plus forte que cette distance entre les poids synaptiques de la cellule et les valeurs du vecteur d'entrée sera faible. Le choix d'une cellule particulière permet alors d'ajuster les poids localement, en minimisant la différence qui existe encore entre les poids et le vecteur d'entrée. Cet ajustement se fait suivant une forme de voisinage qui peut être carrée, ronde ou hexagonale. La taille du voisinage décroît de manière progressive lors de l'apprentissage et les valeurs des connexions sont ajustées selon une fonction qui suit l'allure de celle présentée à la figure 2.16. Cette figure, utilisée par Kohonen, a été baptisée "chapeau mexicain" et correspond à la dérivée seconde de la fonction

gaussienne.

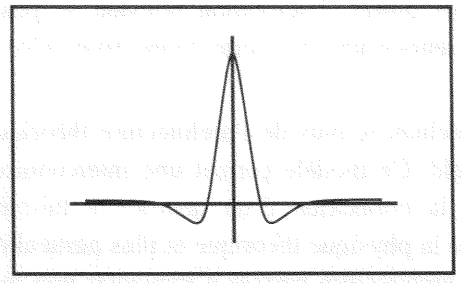


Figure 2.16 : Forme schématique de la fonction d'apprentissage utilisée dans les cartes auto-organisatrices de Kohonen (d'après [kohonen82]).

Formellement, ce processus d'apprentissage se traduit par les 5 phases suivantes qui sont itérées jusqu'à la minimisation d'une erreur globale calculée sur l'ensemble du corpus :

- présentation d'une forme en entrée du réseau, le vecteur d'entrée étant de taille N : (e_1, e_2, ..., e_N),
- calcul de la distance entre cette forme et chaque neurone de la couche de sortie. La couche de sortie possède N neurones et la distance est ici euclidienne, c'est à dire calculée par la méthode des moindres carrés :

$$d_j^2 = \sum_{i=1}^{N} (e_i - w_{ij})^2 \qquad \text{(Éq. 2.8)}$$

- recherche de la distance minimale :

$$d_k^2 = min_{1 \le j \le M}\left(d_j^2\right) \qquad \text{(Éq. 2.9)}$$

- mise à jour des connexions synaptiques de la carte. Cette mise à jour se fait pour tous les neurones j se trouvant dans le voisinage $V(k,P)$ de taille P du neurone vainqueur k.

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(t) \cdot (e_i - w_{ij}(t)) \qquad \text{(Éq. 2.10)}$$

- diminution du coefficient d'apprentissage $\alpha(t)$ et diminution de la taille du voisinage $V(k,P)$

Notons que la fonction de voisinage peut avoir d'autres définitions que celle donnée par Kohonen, voir, par exemple, [carpinteiro96].

2.5/ Modèles connexionnistes dynamiques

Ce paragraphe présente quelques unes des architectures connexionnistes récurrentes existantes. Nous ne attacherons cependant pas à citer toutes les architectures dont les principes nous ont intéressé. Nous reviendrons en effet sur les réseaux connexionnistes dynamiques de manière plus approfondie dans le chapitre 6.

La caractéristique distinctive des réseaux dynamiques par rapport aux réseaux statiques est la mise en place de connexions récurrentes dans l'architecture. Ce mécanisme peut cependant être mis en place à divers degrés d'importance. Nous avons donc distinguer ici trois niveaux d'architecture, que nous reprendrons plus tard au chapitre 6. Nous allons donc maintenant énoncer brièvement les caractéristiques des réseaux totalement récurrents, des réseaux à récurrence par plaque et des réseaux à récurrence locale.

2.5.1/ Modèles connexionnistes totalement récurrents

Nous qualifions ici de réseau totalement récurrent un réseau donc les connexions permettent de relier les neurones en eux sans référence à un strict flot des données comme il en existe un pour les

perceptrons ou les cartes auto-organisées. Ainsi, de manière théorique, tous les neurones peuvent être connectés les uns aux autres, l'activation pouvant se propager dans les deux sens. Cette architecture théorique présentant une très forte connectivité n'est cependant implantée dans aucun modèle connexionniste.

L'architecture se rapprochant le plus de l'architecture théorique dont nous venons de parler est celle du réseau de Hopfield. Ce modèle permet une interconnexion totale des neurones avec des connexions symétriques, la connexion d'un neurone à lui-même n'étant pas autorisée. Cette architecture est inspirée de la physique théorique et plus particulièrement de la théorie des verres de spins [hopfield82]. Cette architecture permet d'implanter une mémoire associative. Le modèle de Hopfield peut être complexifié avec des récurrences locales aux neurones comme cela a été fait dans les neurones pulsés [derou94]. Les récurrences locales ne sont cependant pas, dans ce dernier cas, au même niveau que les connexions intercellulaires puisque la récurrence est effectuée avec la transformation non linéaire.

Une architecture comportant moins de connexions entre les neurones est l'architecture appelée machine de Boltzmann. Dans ce cas, le réseau ressemble à un perceptron multicouche, avec une couche d'entrée et une couche de sortie mais la connectivité à l'intérieure de la "couche cachée" est beaucoup plus anarchique que pour le perceptron, incluant la possibilité de connexions récurrentes [hinton84]. L'algorithme d'apprentissage est, par ailleurs, assez particulier puisqu'il fait appel à la théorie du recuit simulé [ackley85]. Quelques applications de la machine de Boltzmann en reconnaissance de la parole ont été tentées [prager86] avec des résultats cependant mitigés, les machines de Boltzmann étant assez peu stables [azencott92a].

D'autres architectures dynamiques peuvent être définies. Il existe par exemple un modèle connexionniste architecturalement proche mais fonctionnellement éloigné de la machine de Boltzmann : la mémoire à court terme de Zipser [zipser91]. D'autres notions peuvent être utilisées telle que la création de nœuds dans le réseau en fonction de la définition des classes. Les modèles de la famille des réseaux se rattachant à la théorie de la résonnance adaptative [carpenter88] mettent en œuvre cette dernière notion. Il est même possible de l'étendre en fonction de la théorie du chaos [dogaru95].

La structure de la colonne corticale que nous avons présenté dans le paragraphe 2.2.3 peut également être implantée sous la forme d'un réseau connexionniste à récurrence forte. Quelques implantations de la colonne corticale ont été réalisées avec une référence forte à la neurobiologie [alexandre90], [ingber81], [ingber82], [massone94] et les modèles obtenus partagent parfois des concepts avec une extension particulière des machines de Boltzmann [azencott94].

2.5.2/ Modèles connexionnistes à récurrence par plaque

À la différence des réseaux totalement récurrents que nous venons de voir, les réseaux à récurrence par plaque permettent de distinguer des structures en couches identiques à celles présentes dans les perceptrons multicouches. Mais à la différence des perceptrons, où le flot des données passe obligatoirement de la couche d'entrée vers la couche de sortie, certaines couches sont ici rebouclées à l'intérieur du réseau.

Il est ainsi possible de reboucler la couche de sortie sur la couche d'entrée, ce qui permet d'obtenir un réseau de Jordan, présenté dans [jordan86]. Plutôt que de reboucler la couche de sortie, il est possible de reboucler la couche cachée vers la couche d'entrée. L'architecture correspond alors au modèle de Elman [elman88]. Il est bien sûr possible de reboucler la couche de sortie et la couche cachée sur la couche d'entrée, ce qui permet de mixer les avantages des modèles de Jordan et d'Elman. Ce modèle a été présenté dans [hanson96]. Enfin, plutôt que d'effectuer un rebouclage d'une couche quelconque vers l'entrée, il est possible d'effectuer le rebouclage sur la couche cachée. Ainsi, [robinson89] propose une architecture où la couche cachée est rebouclée sur elle-même. Cette dernière architecture peut cependant être considéré comme une variante de celle de Elman.

2.5.3/ Modèles connexionnistes à récurrence locale

Un dernier type d'architecture récurrente peut être envisagé. Il s'agit des réseaux où les neurones possèdent des mécanismes de récurrence locale. L'architecture globale du réseau n'est pas déterminée par ces récurrences locales et cette architecture peut aussi bien être similaire à un perceptron [frasconi92], [tsoi94] qu'à un réseau de Hopfield [derou94].

Le mécanisme de récurrence locale au neurone est lui-même un concept recouvrant toute une série de choix qui définiront l'architecture de ce mécanisme :

- la récurrence doit-elle rebouclée en entrée du neurone, avec les activations afférentes, ou après la somme pondérée ?
- la récurrence doit-elle être placée avant, après ou autour de la non-linéarité ?
- la récurrence doit-elle être associée à un coefficient pondérateur ou doit-elle être prise en compte telle que ? Au cas où un coefficient est utilisé, faut-il contraindre celui-ci sur un espace restreint de valeurs ?
- la récurrence doit-elle mettre en œuvre un ou plusieurs délais ?
- la récurrence doit-elle être mettre en œuvre un mécanisme de délai simple ou un mécanisme plus complexe ?

Toutes ces questions permettent de faire un choix sur la complexité de la récurrence qui sera utilisée avant même de définir l'architecture du réseau lui-même. Il est aisé de comprendre que le concept de récurrence locale a conduit à la définition de nombreuses architectures connexionnistes. Plutôt que de commencer à en faire un liste ici, nous renvoyons le lecteur aux chapitres 6 et 7 où cette liste est d'ailleurs encore loin d'être complète.

2.5.4/ L'apprentissage dans les modèles dynamiques

Les modèles dynamiques ont, eux-aussi, besoin d'une méthode capable de déterminer correctement les valeurs des poids synaptiques. Les premières méthodes mises au point n'étaient, en fin de compte, qu'une réécriture de la rétropropagation adaptée aux réseaux récurrents [almeida87], [pineda87]. Dans ces méthodes, le gradient d'erreur n'est pas déterminé complètement comme pour les réseaux statiques (paragraphe 2.4.4) mais il est déterminé partiellement, une partie de l'erreur devant être calculée récursivement.

Cette réécriture de la rétropropagation a conduit à l'écriture de l'algorithme d'apprentissage récurrent en temps réel, *Real-Time Recurrent Learning* [williams89a], [williams89b]. Ce dernier algorithme n'est pas différent des algorithmes précédents bien qu'il soit présenté d'une autre manière et constitue en fait une généralisation de la prise en compte de la récurrence pour tout type d'architecture, les récurrences locales constituant cependant une exception à cette généralité. Il est intéressant de noter qu'une version particulière de cet algorithme a été présentée dans [robinson89] pour une architecture particulière, cet algorithme ayant été complété par de nombreuses heuristiques par la suite [robinson91]. Enfin, signalons qu'une méthode d'amélioration de cet algorithme de rétropropagation du gradient implante de manière expérimentale [catfolis93] une partie de l'algorithme de détermination des caractéristiques des systèmes dynamiques non linéaires tel qu'il est proposé dans [takens81], le taux d'échantillonnage de l'information étant dans les deux cas laissé au concepteur du système.

Toutes les techniques d'apprentissage que nous venons d'exposer relèvent d'une même utilisation de la rétropropagation. Le gradient est déterminé de manière approximative à un pas de temps par rapport à l'approximation qui en avait été faite au pas de temps précédent. Cette technique peut être abandonnée au profit d'une autre permettant de réaliser des calculs exacts mais beaucoup plus gourmande en ressource une fois venue l'implantation machine. Cette méthode permet de faire de la rétropropagation dans le temps, *Back-Propagation Through Time*, en dupliquant le réseau à chaque pas de temps avant de rétropropager le gradient à travers toutes les couches architecturales et

temporelles [werbos90]. Cette méthode constitue en fait la duale de celle que nous avons présenté dans le paragraphe précédent [beaufays94].

D'autres méthodes existent encore mais sont cependant moins utilisées car les résultats qu'elles permettent d'obtenir sont de moins bonne qualité que ceux obtenus par l'apprentissage récurrent en temps réel et la rétropropagation dans le temps. Parmi celles-ci, il est possible de citer les algorithmes d'apprentissage de Pearlmutter [pearlmutter90] et de Green [sun91] qui ont été étudiées et comparées à l'apprentissage récurrent en temps réel dans [logar93]. Il est également possible de rapprocher l'apprentissage dans les réseaux dynamiques des espaces d'états [pearlmutter89] ou des espaces de phases [tsung93]. Il a par ailleurs été montré que l'apprentissage dans les modèles dynamiques était sensible au codage des entrées [omlin94].

Il existe également des méthodes spécifiques à certains types d'architecture. Une dérivation de la rétropropagation pour les réseaux à récurrence locale a, par exemple, été présentée dans [leigthon91].

L'apprentissage dans les modèles dynamiques peut être étudié à un niveau plus abstrait que celui de l'application. De telles études pourront être trouvées dans [baldi94], [baldi95] et [nerrand94]. Ces études permettent de comprendre, en partie, les difficultés que posent les réseaux dynamiques lors de la phase d'apprentissage [bengio94a].

Les capacités de différentes architectures statiques et dynamiques ont été étudiées et comparées dans [horne95] sur des tâches nécessitant une capacité de mémorisation.

PARTIE 2

CONTRIBUTION

CHAPITRE 3 : PROBLÉMATIQUE DU BRUIT EN RAP

Loi de Hofstadter

"Ça prend toujours plus de temps qu'on ne le croit, même si l'on tient compte de la loi de Hofstadter"

Loi de Byrd

"Ça prend toujours plus de temps qu'on ne le croit, même si l'on tient compte de la loi de Hofstadter"

Résumé

Ce chapitre permettra au lecteur d'avancer dans la compréhension du problème étudié dans cette thèse : la reconnaissance de petits vocabulaires prononcés de manière continue en environnements bruités. Après avoir exposé notre sujet et le cadre plus général auquel il se rapporte, nous présenterons les problèmes que le bruit pose à l'être humain et à la machine. Nous donnerons ensuite un bref aperçu des premiers travaux que nous avons effectués sur des méthodes d'énergie.

3.1/ Objectif

3.1.1/ Mise en œuvre d'un système de Reconnaissance Automatique de la Parole

Un système de reconnaissance de la parole fonctionnant efficacement hors du domaine de la recherche est en soit prometteur. Un tel système, s'il existait, aurait de nombreuses applications dans toutes les activités humaines [carre91]. La parole étant le principal vecteur de l'information entre les êtres humains, l'acquisition du don de la parole par les machines permettrait des développements extraordinaires. Un véritable dialogue homme-machine multimodal dont le naturel se rapprocherait du dialogue homme-homme pourrait alors émerger. Le dialogue homme-machine ne serait plus alors limité par les capacités restreintes des seuls claviers et souris. Mais la Reconnaissance Automatique de la Parole, RAP, n'est encore aujourd'hui qu'un vaste sujet d'étude.

Le domaine du dialogue oral homme-machine et, plus généralement, le domaine des interfaces homme-machine fait intervenir des sciences et des techniques encore mal maîtrisées. Ainsi les notions de lexique et de syntaxe sont-elles encore mal comprises. Les langues pratiquées couramment par diverses populations doivent être qualifiées de vivantes et une théorie s'accommode mal d'un phénomène aussi évolutif. Au delà de la forme de la langue, la sémantique et la pragmatique sont bien plus complexes que la simple logique du premier ordre [moeschler95]. Tous les paradigmes qui viennent d'être cités sont généraux à la langue et concernent tout autant la parole que l'écrit. Par rapport à la RAP, ces paradigmes peuvent être qualifiés de "haut niveau", par

opposition à la phase de décodage acoustico-phonétique, DAP, souvent qualifiée de "bas niveau". Les paradigmes de haut niveau ne sont cependant pas strictement postérieurs aux phases de bas niveau puisque des interactions peuvent intervenir d'un haut vers un bas niveau.

Le langage humain, qu'il soit parlé ou écrit, est donc soit restreint à un sous-vocabulaire, ce qui permet d'en garder le contrôle (cf figure 3.1), soit laissé libre, cette liberté étant génératrice d'une complexité qui est encore très mal maîtrisée.

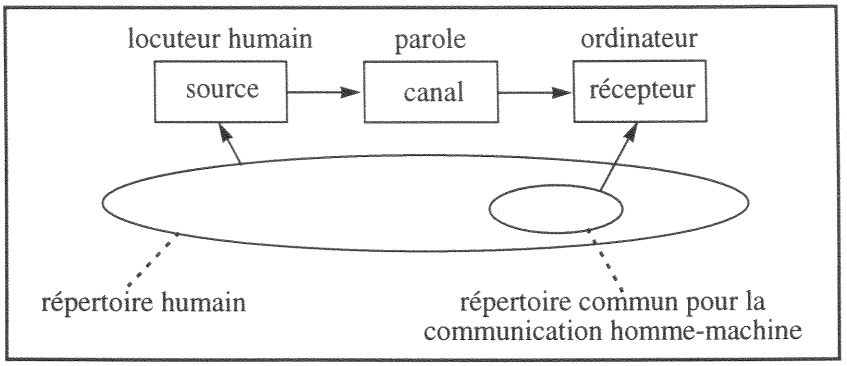


Figure 3.1 : Un système de communication homme-machine (d'après [pierrel87]).

Au delà du simple système de retranscription phonétique qui, aujourd'hui encore, peut être considéré comme un sujet de recherche, il faudra résoudre des problèmes aux niveaux "supérieurs" à celui de la retranscription [haton91].

3.1.2/ Existence du besoin d'un système fiable

Les premières recherches menées sur les systèmes de RAP ont été conduites en conditions de laboratoire. L'important était alors de comprendre le phénomène qu'est la parole. La parole étudiée était parfaite dans le sens où son enregistrement était fait dans des conditions optimales. Les locuteurs avaient une prononciation claire et étaient généralement isolés dans des locaux très peu bruyants. Ces conditions imposées à juste titre par la recherche ont cependant eu comme conséquence d'annihiler la réalité du monde sonore tel qu'il est. Les premiers systèmes, bien que capables de fonctionner correctement sur les corpus de parole qui avaient été étudiés, ne pouvaient plus fonctionner convenablement lorsque la parole analysée se trouvait confondue à un bruit de fond ou lorsque la parole avait été modifiée lors de sa transmission.

Savoir appréhender ces types de contraintes est donc un besoin impérieux pour une utilisation effective de la communication orale homme-machine dans la société. Il faut donc revoir les recherches déjà menées en tenant compte du fait que le signal à traiter effectivement n'a pas la pureté des corpus initialement enregistrés en chambre anéchoïde.

L'étendue des problèmes dont il faut tenir compte est vaste. Traiter de la parole suppose que tout son qui n'appartient pas au message à analyser doit être considéré comme du bruit. Il faut même en fait aller plus loin et estimer que tout son autre que la parole du locuteur est à considérer comme du bruit. Ceci suppose que des systèmes de RAP présents dans une même pièce fréquentée par plusieurs personnes devront avoir des réactions différentes en fonction du locuteur, le système devant ou non prendre en compte un message qu'il entend selon qu'il vient de son utilisateur ou non. Un système placé dans de telles conditions devra également reconnaître les phrases prononcées par son utilisateur alors que l'onde de parole de ce dernier pourra être partiellement masquée par la parole des autres personnes présentes dans la pièce.

3.1.3/ Ambitions relatives au système à développer

Notre ambition lors de cette thèse est de développer un système qui soit capable de faire de la reconnaissance de petits vocabulaires en environnements bruités pour plusieurs locuteurs.

Les petits vocabulaires sont restreints à des corpus de mots tels que les 10 chiffres ou les lettres de

l'alphabet prononcées isolément les unes des autres. Ce type de petits vocabulaires est employé dans des tâches de composition de numéros téléphoniques [lockwood93] ou d'épellation de noms [anglade94]. De telles tâches ne nécessitent quasiment aucun niveau supérieur car seule une éventuelle contrainte syntaxique peut être ajoutée de manière à vérifier la concordance de la suite de chiffres ou de lettres avec les mots d'un dictionnaire. Ce dictionnaire peut correspondre, dans le cas des chiffres, à un annuaire ou à la liste des postes téléphoniques d'une entreprise ou bien, dans le cas de lettres, à une liste de noms de personnes répertoriés de différentes manières.

Le système à développer doit également pouvoir fonctionner quel que soit le locuteur. Notre étude se place donc dans le cas d'un système indépendant du locuteur. Cette contrainte est assez importante puisque nous avons vu (cf chapitre 1, paragraphes 1.6.2 et 1.6.3) que la parole variait d'un individu à l'autre et pouvait varier pour un même individu d'un jour à l'autre. Nous ne faisons pas ici la distinction entre une tâche en mode multi-locuteur et une tâche en mode locuteur indépendant puisque les corpus de parole pré-enregistrés qui ont été utilisés pendant cette thèse correspondent à ces différentes acceptions. Cette distinction n'est d'ailleurs pas très forte et de bons taux de reconnaissance ont déjà été obtenus par ailleurs en mode locuteur indépendant sur des corpus de grande taille [robinson94].

3.1.4/ Contraintes imposées

Les contraintes imposées au système à définir ne sont pas nombreuses. Elles sont cependant assez générales pour devenir très fortes.

Nous avons tout d'abord l'intention de développer un système qui soit directement applicable et donc immédiatement utilisable par une application de plus haut niveau. Pour que ce système soit directement applicable, il faut qu'il puisse être capable de fonctionner sans avoir besoin d'une quelconque refonte. Il est en effet possible, dans une certaine mesure, de développer un système viable adapté à un environnement sonore particulier, il faut pour cela le bâtir en tenant compte des contraintes très particulières de l'environnement cible. Le système ainsi développé devient totalement ad hoc et est inapplicable à tout nouvel environnement différent de l'environnement cible d'origine. Une adaptation à un environnement différent nécessitera donc une phase de développement supplémentaire qui donnera lieu à un nouveau système, lui aussi spécifique à son environnement cible. Le système que nous voulons développer doit donc être applicable directement et, par là même, être capable de fonctionner correctement dans un grand nombre d'environnements. Nous ne pouvons pas prétendre obtenir un système parfait qui soit capable d'aller au delà des capacités humaines. Nous pouvons cependant essayer de faire un système qui ne soit pas trop sensible à des conditions de bruit différentes. Les études sur les capacités humaines de la compréhension de la parole dans le bruit sont, en outre, limitées et les capacités considérées sont donc difficiles à quantifier.

Une autre contrainte importante est la vitesse d'adaptation du système à des environnements de bruit différents. En effet, pour être applicable à différents environnements, un système pourrait nécessiter un temps d'adaptation plus ou moins long. Cette phase d'adaptation devrait être effectuée lors de la mise en route du système, constituant une sorte de démarrage à froid de l'application. Mais il peut aussi être nécessaire d'effectuer cette adaptation en cours d'utilisation, une sorte de redémarrage à chaud, dans le cas où les conditions courantes de bruit sont estimées trop éloignées des conditions initiales pour ne pas dégrader le fonctionnement. La meilleure solution possible serait dans ce cas que le système s'adapte en permanence aux conditions de bruit courantes sans que cette adaptation ne dégrade pour autant la vitesse de réponse du système. L'ultime possibilité serait de mettre au point un système qui n'ait pas besoin de s'adapter en permanence au bruit. Un tel système serait donc directement applicable et la contrainte d'adaptation rejoint alors la contrainte d'applicabilité.

3.2/ Résistance de la parole au bruit

3.2.1/ L'influence du bruit dans la communication

Définir le bruit dans l'absolu est impossible car cette notion est assez subjective. Pourraient être définis comme étant du bruit tous les phénomènes qui empêchent la transmission d'un message d'une source à sa destination ou tout ce qui détériore la qualité et l'intelligibilité du message transmis.

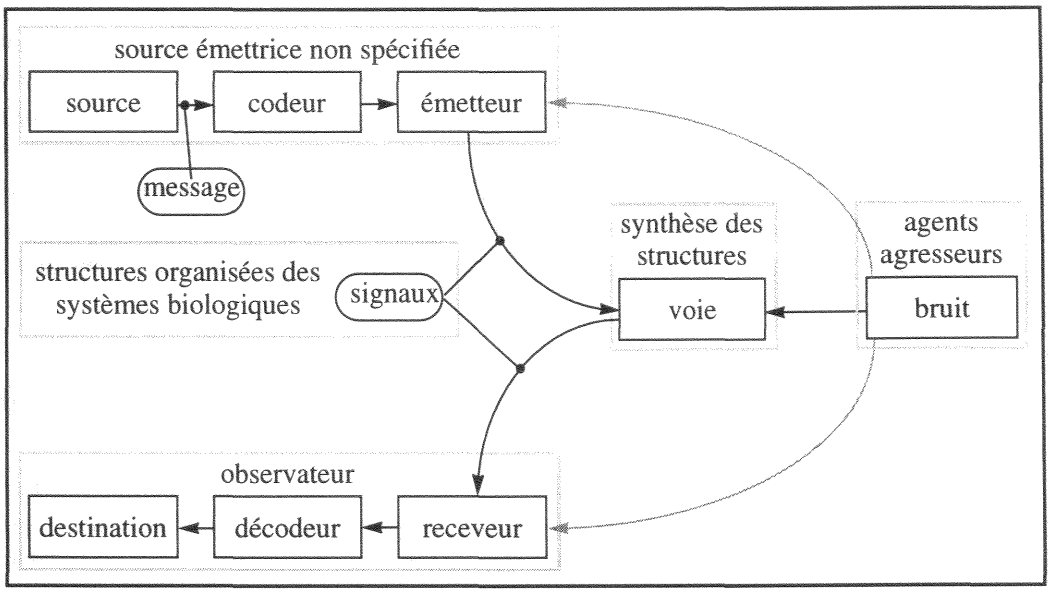


Figure 3.2 : Schéma d'un système de communication et identification de ses éléments avec ceux d'un système biologique (d'après [atlan92] et [quastler58]).

Le bruit intervient donc sur le ou les média de transmission et amoindrit les capacités de la voie de communication comme cela est décrit dans la figure 3.2 [atlan92]. Le schéma original que nous avons repris n'est cependant pas complet et nous avons ajouté deux interactions supplémentaires, l'une de la source de bruit vers l'émetteur et l'autre de la source de bruit vers le receveur. Le bruit peut en effet perturber l'émetteur lors de sa production, celui-ci modifiant le signal qu'il envoie. Il est en particulier possible d'observer, dans ce cas, l'effet Lombard qui est un signe de l'élévation de la voix [lombard11]. Le receveur peut, lui, voir ses capacités de réception amoindries par une exposition trop longue au bruit ou par une exposition à un bruit hors des limites de sa résistance. Cependant, ces dernières influences du bruit doivent être vues comme étant des modifications à plus long terme que la simple détérioration du message.

3.2.2/ Qualité d'un message

3.2.2.1/ Critères de qualité

Un message est très généralement porteur de sens et la parole n'échappe pas à ce principe. Mais ce sens peut être partiellement ou totalement caché par du bruit. La qualité d'un message permet de définir un critère de bonne réception du message vis-à-vis des agressions de l'environnement. Mais la qualité est, en général, une notion assez subjective pour qu'il faille rappeler les méthodes qui permettent de la quantifier.

Les critères de qualité d'un message peuvent être divisés en deux groupes que sont les critères objectifs et les critères subjectifs. Les critères subjectifs sont des jugements portés par l'auditeur sur le message qu'il reçoit. Ces critères sont dits subjectifs car ils ne font intervenir qu'un seul des deux intervenants de la communication, en l'occurrence le receveur, qui n'a, a priori, aucune connaissance sur le message qui lui est envoyé. Ces tests de qualité ont été, entre autres, employés pour l'évaluation des méthodes de numérisation de la parole sur des lignes à plus ou moins haut débit et

donc avec des compressions plus ou moins fortes [potage90c]. Les critères objectifs sont des mesures qui tiennent compte du message tel qu'il est fourni par l'émetteur et tel qu'il est compris par le receveur. Les critères objectifs permettent donc des calculs exacts bien que certaines erreurs, dues à la méthode de calcul employée ou à de trop fortes approximations, puissent cependant apparaître.

3.2.2.2/ Critères objectifs de qualité

Ces critères font appel à des mesures de dissemblance sur le signal et font intervenir le message au niveau de l'émetteur et au niveau du récepteur.

Le critère de qualité objectif le plus couramment utilisé pour quantifier la qualité d'une transmission est le rapport signal-sur-bruit, RSSB ou RSB, en anglais *signal-to-noise ratio*, SNR. Le rapport signal-sur-bruit est défini par :

$$RSSB\,(dB) \;=\; 10\log_{10}\left[\frac{\sum_{n=1}^{N} s^{2}(n)}{\sum_{n=1}^{N} [\,s(n) - s'(n)\,]^{2}}\right] \qquad \text{(Éq. 3.1)}$$

Dans cette équation du rapport signal-sur-bruit, s représente le signal reçu à l'issue de la transmission et s' représente le signal d'origine, par définition non bruité.

Le rapport signal-sur-bruit permet, en fait, de connaître la valeur de l'inverse de l'erreur quadratique moyenne $\frac{1}{N}\sum_{n=1}^{N} [\,s(n) - s'(n)\,]^{2}$ normalisée par la puissance du signal $\frac{1}{N}\sum_{n=1}^{N} s^{2}(n)$. Le calcul, faisant intervenir le signal original et le signal finalement perçu, donne une indication de l'agressivité des agents extérieurs sur la voie de communication. Mais ce critère est sensible. Il faut par exemple parfaitement aligner le message émis et le message reçu avant d'effectuer le calcul car la méthode de calcul est très sensible aux déphasages. De plus, le RSSB ne tient pas compte de la répartition spectrale de l'énergie de l'erreur et, à énergie totale égale, une distorsion additive sera moins perceptible qu'une autre si son spectre suit le spectre du signal original de manière forte. Enfin, l'intervalle de calcul du RSSB n'est pas sans importance. Le RSSB est généralement calculé sur une phrase entière et cette méthode ne permet pas d'avoir une bonne idée de ce que seront les qualités subjectives du message. Pour pallier ce dernier inconvénient, il convient de calculer la moyenne du RSSB sur un ensemble de blocs pour obtenir le RSSB segmental.

Le RSSB segmental est calculé sur un ensemble de N blocs de longueur M (M ayant généralement une durée de 16 millisecondes, durée en relation avec l'inertie de l'appareil phonatoire, et $N \times M$ étant la durée de la phrase à analyser) :

$$RSSB \text{ segmental } (dB) \;=\; \frac{1}{N}\sum_{i=0}^{N-1} 10\log_{10}\left[\frac{\sum_{n} s^{2}(n+iM)}{\sum_{n} [\,s(n+iM) - s'(n+iM)\,]^{2}}\right] \qquad \text{(Éq. 3.2)}$$

Cette mesure présente l'avantage de tenir compte de l'évolution du RSSB au cours du temps. La parole étant composée de juxtaposition de segments non stationnaires (et non prédictibles) et de segments quasi-stationnaires, le RSSB segmental permet d'avoir une mesure qui est plus correllée avec les mesures subjectives.

Il faut également signaler qu'il existe d'autres méthodes de calcul du RSSB qui font intervenir des notions fréquentielles. Les RSSB sont alors calculés dans un certain nombre de plages fréquentielles respectant un modèle très simplifié de l'oreille. Certains critères subjectifs sont basés sur ces méthodes (cf. paragraphe 3.2.2.3).

Les critères objectifs permettent d'avoir des indications utiles sur les performances d'un système. Mais dans le cas de la parole, le récepteur ultime étant un être humain, les propriétés subjectives sont bien plus pertinentes bien qu'elles soient plus difficiles à collecter.

3.2.2.3/ Critères subjectifs de qualité

Les critères subjectifs ne font intervenir que le receveur du message et sont donc en partie liés à ses capacités intrinsèques. Ils ne prennent donc un sens que lorsque les tests statistiques sont effectués de manière à avoir un intervalle de confiance le plus réduit possible.

Il existe plusieurs critères pour juger de la qualité subjective d'un message mais toutes les études s'accordent à mettre en avant quatre notions importantes dans le domaine de la parole :

- l'intelligibilité, c'est l'évaluation de la capacité d'un système de communication à fournir de l'information parlée qui soit compréhensible à un locuteur. Cette mesure peut être faite au niveau de la phrase, du mot ou du phonème,
- l'agrément, c'est la mesure de préférence d'un type de sons de parole par rapport à un autre, cette mesure est donnée par l'auditeur de manière totalement arbitraire,
- le niveau sonore, qui est un paramètre purement physique,
- la reconnaissance du locuteur, qui est un paramètre psychologique, dans la communication homme-homme tout du moins.

L'intelligibilité et l'agrément sont les deux critères les plus importants au niveau de la transmission puisqu'ils mesurent le confort de l'écoute ou l'effort à produire et la compréhensibilité de ce qui est transmis.

D'autres tests subjectifs ont été mis en place pour juger la qualité des sons transmis ou synthétisés. Ainsi le test de rime, en anglais *Diagnostic Rhyme Test*, DRT, permet d'évaluer la netteté d'un vocodeur [potage90c]. Ce test est basé sur une série de tests statistiques à réponse fermée, effectués sur différents auditeurs, et qui permettent d'obtenir une mesure de netteté globale. Ce test permet de mesurer le degré de dégradation des caractéristiques élémentaires des consonnes lorsque celles-ci se trouvent en début de mot [miller55], [peckles73]. Une version plus générale du test DRT, le *Modified Rhyme Test*, MRT, a par la suite été mis en place pour tester tout aussi bien les voyelles que les consonnes et, ce, quelle que soit leur position dans un mot [fairbanks58]. Le DRT peut également être étendu à la production de matrices de confusion en autorisant les réponses ouvertes [steenek86].

Bien que les critères d'évaluation subjectifs soient difficiles à mettre en œuvre, une comparaison entre les résultats de tels tests et les capacités de systèmes de RAP à fonctionner dans des milieux bruités variés pourrait être intéressante, tant du point de vue des résultats corrélés que des résultats décorrélés [steenek92b].

Il faut noter qu'il existe des recherches visant à automatiser l'évaluation des critères subjectifs. Ainsi [french47] définit l'index d'articulation (*Articulation Index*, AI) qui repose sur le calcul d'un RSSB fréquentiel réparti sur 20 bandes de fréquences. Cet index tend à déterminer la qualité subjective de la parole en effectuant de simples mesures physiques et, donc, en supprimant les auditeurs humains. L'*Articulation Index* a fait l'objet d'une standardisation ANSI [ansis35].

D'autres méthodes d'évaluation automatique ont été définies. Citons, entre autres, le SIL (*Speech Interference Level*) [beranek54] qui est une version simplifiée de l'*Articulation Index* [allen94] et le STI (*Speech Transmission Index*) [steenek80].

3.2.3/ Les différents types de bruit

Les différents bruits pouvant influer sur un message peuvent être divisés en deux grandes catégories : les bruits additifs et les bruits convolutionnels. La distinction entre les deux peut être faite par le nombre d'agents agresseurs extérieurs à la transmission du message. Les bruits additifs sont causés par des agents extérieurs au trinôme source-voie-destinataire alors que les bruits

convolutionnels sont causés par la moindre qualité de la voie de communication, celle-ci ayant alors un rôle ambigu, du point de vue du message, de médium et d'agresseur.

3.2.3.1/ Les bruits additifs

Les bruits additifs sont dûs à la multiplicité des systèmes de communication dans un même environnement. Plusieurs émetteurs et plusieurs receveurs pouvant être confinés dans un même espace, les messages de tous les émetteurs peuvent donc se trouver en concurrence sur une même voie sans que les récepteurs possèdent un mécanisme infaillible pour isoler le message qui leur est destiné. L'émetteur et le récepteur peuvent aussi se trouver en présence d'un ou de plusieurs équipements générant un bruit de fond de force variable.

Les bruits additifs peuvent être subdivisés en trois groupes en fonction des lieux où ils peuvent être rencontrés :

- bruits des systèmes industriels : ils peuvent être très intenses et sont, par nature, non stationnaires. Ils correspondent aux bruits émis par des machines possédant une faible isolation phonique.
- bruits des moyens de transport : ils correspondent aux bruits qui peuvent être observés dans diverses véhicules tels que les voitures, les trains ou les avions.
- bruits des milieux administratifs et urbains : ce sont les bruits présents dans les bureaux, les domiciles ou dans les concentrations urbaines. Ces bruits peuvent être très variés (climatisation ou bruit de parole) mais sont peu intenses.

Les bruits produits par les systèmes industriels sont très souvent des bruits rythmiques, ou périodiques, correspondant à la répétition d'une tâche de nature productive. Cette définition doit cependant être nuancée car elle correspondrait à une usine totalement automatisée où l'homme n'aurait plus sa place. L'automatisation totale des sites de production n'étant pas encore atteinte, il faut également considérer les bruits produits directement ou indirectement par l'homme. Au titre de ceux-ci peut se retrouver le bruit de parole (cf. annexe 3, figures A3.2 et A3.7) qui est le fait immédiat de l'homme. Il est également possible de classer, dans ce type des bruits produits par les systèmes industriels, le bruit des outils de travail des ouvriers présents sur un site, tel que le bruit du petit matériel électrique (cf. annexe 3, figure A3.8).

Les bruits produits par les moyens de transports se caractérisent généralement par une très forte stationnarité qui correspond à la vitesse de fonctionnement des organes moteurs. Le bruit observé est ainsi constitué d'un ou de plusieurs harmoniques et ne comporte que de micro-fluctuations. Ces remarques générales doivent cependant être nuancées par l'observation du bruit de certains moyens de transport tels que le train ou le bateau. Dans le cas du train, un bruit non-stationnaire et rythmique est présent tout au long du déplacement. Dans le cas du bateau, le moteur peut fonctionner de manière très lente, surtout sur de grosses unités, et produire lui aussi un bruit rythmique. Ces bruits peuvent également varier en fonction des conditions de déplacement.

Les bruits produits dans les milieux administratifs et urbains sont les bruits qu'il est possible de rencontrer dans la vie de tous les jours. Ce sont les bruits des systèmes de ventilation, des machines à écrire ou des ordinateurs, voire même des systèmes d'éclairage. À cette liste peuvent être ajoutés les bruits de mobiles par rapports à l'auditeur tels que les voitures ou les avions. Ces bruits peuvent être relativement intenses dans de rares cas et sont toujours momentanés au contraire des bruits de moyens de transport où l'auditeur est un passager.

3.2.3.2/ Les bruits convolutionnels

Les bruits convolutionnels (ou multiplicatifs) sont dûs à la distorsion induite par la voie de communication. Ils résultent de la mauvaise qualité d'un ou de plusieurs éléments de support du message ou, tout simplement, de son étroitesse en bande passante.

Les sociétés modernes utilisent de plus en plus de moyens de communication à longue distance

tels que le téléphone, les moyens radiophoniques et, récemment, radiotéléphoniques. Ces moyens de communication à longue distance ont été élaborés à partir d'un compromis coût/efficacité. La parole, lorsqu'elle est transmise par un tel moyen, est forcément dégradée tout en gardant une grande intelligibilité.

Un des champs possibles d'application de la RAP sont les serveurs vocaux accessibles par les lignes téléphoniques. Mais la parole transmise par téléphone souffre de déformations variables induites par la qualité de la connexion. Une transmission peut ainsi souffrir de l'étroitesse de la bande passante, de la mauvaise qualité des microphones de certains terminaux téléphoniques, de bruits additifs stationnaires et de porteuses basse fréquence [moreno94]. La qualité de la transmission varie cependant très peu au cours d'une même communication [mokbel93].

De manière plus générale, le bruit convolutionnel est présent dans toute application de RAP par l'intermédiaire du microphone utilisé pour la saisie de la voix. Un système de RAP mis au point avec certain microphone pourra voir ses capacités diminuer de manière conséquente lorsqu'un autre microphone sera employé [acero90]. La parole enregistrée dans tous les corpus utilisés pour la recherche est en effet toujours bruitée puisque le microphone utilisé effectue toujours un filtrage linéaire.

Enfin, certains milieux d'enregistrement sont de mauvaise qualité et peuvent provoquer des phénomènes de réverbération. C'est notamment le cas des pièces possédant de grandes surfaces faites d'un matériau dur ou lorsque le microphone utilisé pour l'enregistrement est placé assez loin du locuteur. Pour résoudre ce problème, on utilise généralement un ensemble de microphones pour trouver le filtre inverse [wang91], [comper90].

3.2.3.3/ Les bruits physiologiques

D'autres bruits peuvent également être considérés dans le domaine de la RAP mais ils n'ont pas la généralité des bruits de type additif ou convolutionnel car ils sont spécifiques à l'être humain lors de sa phase de production de parole.

Les plupart des systèmes de RAP fonctionnent mal en milieu bruité car les contraintes posées par de tels environnements n'ont pas été prises en compte dès le départ. L'homme essaie, lui, de s'adapter aux conditions sonores rencontrées en modifiant sa méthode de production de parole

Un des phénomènes les plus remarquables de modification de production de la parole par l'homme est l'effet Lombard [lombard11]. Lorsqu'un locuteur est placé dans un environnement bruité, il modifie sa voix, et son effort vocal, en "haussant le ton" de manière à ce que la parole produite conserve un bon RSSB par rapport à l'environnement. Cette accentuation de la voix pose cependant un problème majeur aux systèmes de RAP car les spectres de tous les phonèmes peuvent être modifiés [junqua92] ce qui a pour effet de nettement amoindrir les taux de reconnaissance [rajasekaran86]. Certaines études montrent a contrario que l'homme arrive à avoir de meilleures capacités de compréhension dans le cas de la parole Lombard que pour de la parole normale lorsqu'il lui est demandé de reconnaître des mots isolés ou de la parole continue masqués par du bruit ([dreher57], [summers88]).

Il faut enfin noter qu'il existe des situations où la parole est modifiée sans que l'homme ne modifie sa façon de parler de manière volontaire. Ceci peut arriver lorsqu'un pilote d'avion oblige son avion à entrer dans une phase d'accélération verticale positive ou négative ou lorsque la parole est produite par une personne se trouvant en contact avec un appareil en phase vibratoire très prononcée.

3.2.4/ Capacités humaines

3.2.4.1/ Robustesse de la perception humaine

Comme il a été dit précédemment, les capacités de reconnaissance de la parole en milieu bruité ont été assez peu étudiées. Il est cependant assez évident que l'homme a de très bonnes capacités dans ce domaine. L'environnement humain actuel est très souvent urbain et, par conséquent, très

souvent bruité. À tel point que l'amélioration de ce cadre de vie passe également, aujourd'hui, par une limitation du niveau des nuisances sonores. Mais quel que soit son environnement, l'homme a été capable de s'adapter à des contraintes très différentes et c'est justement cette capacité d'adaptation qui rend les systèmes de RAP si difficiles à mettre en œuvre à grande échelle et dans de nombreux champs d'application.

L'être humain peut cependant se retrouver confronté à des difficultés. Certains phénomènes sonores peuvent ainsi être dangereux pour les organes auditifs. Ce risque et la limite qui lui est associée sont de bons indicateurs de ce à quoi l'homme ne peut pas s'adapter. Mais ces difficultés peuvent être considérées comme naturelles car elles ne sont que le résultat d'évènements plus ou moins courants. Ces limites humaines pourraient donc être considérées, temporairement peut-être, comme les limites effectives de fonctionnement pour les systèmes de RAP. Pourquoi en effet bâtir des systèmes de dialogue capables d'évoluer dans des environnements trop agressifs pour qu'un homme y évolue ? Cette limite reste cependant floue car certains environnements à risque, tels que certains ateliers mécaniques ou même les boites de nuit (!), restent fréquentés par des gens qui peuvent éprouver le besoin de communiquer entre eux par la parole...

D'autres difficultés peuvent apparaître, celles-ci étant liées à des recherches scientifiques. Des études ont confronté l'homme à des tâches qui sont également effectuées par la machine dans la chaîne des traitements de la reconnaissance automatique de la parole. L'homme a en effet quelques difficultés à résoudre les problèmes posés lorsqu'on le met devant des tâches telles que la reconnaissance des voyelles ou des occlusives dans des contextes totalement artificiels [steenek92b]. Ainsi, bien que l'homme puisse communiquer en plusieurs langues, il est prouvé que des auditeurs de parole bruitée auront de meilleurs résultats si la langue utilisée est leur langue natale que si cette langue leur est étrangère bien que le résultat des deux types d'auditeurs (langue natale VS langue étrangère) soient les mêmes pour de la parole non bruitée [gat78]. Ceci tend à montrer l'importance des connaissances de haut niveau sur la langue et également l'importance de l'intégration de ces règles de haut niveau à un stade non verbal.

Une autre étude intéressante [miller55] tente de mesurer la capacité de l'être humain pour une tâche de discrimination de différentes consonnes de l'anglais. Le graphe de la figure 3.3 montre les résultats d'auditeurs sur un test de rime effectué en condition de bruit blanc. Ce graphe présente les confusions progressives entre les différentes consonnes de l'anglais, deux lignes accolées signifiant qu'aucune distinction entre deux consonnes ne peut être effectuée en deçà du seuil de RSSB qui vient d'être franchi. Cette étude tend à prouver les faibles capacités de l'homme sur des tâches de reconnaissance de la parole hors de tout contexte et de tout ancrage lexical et/ou syntaxique, ce type de reconnaissance étant pourtant à la base des systèmes de RAP.

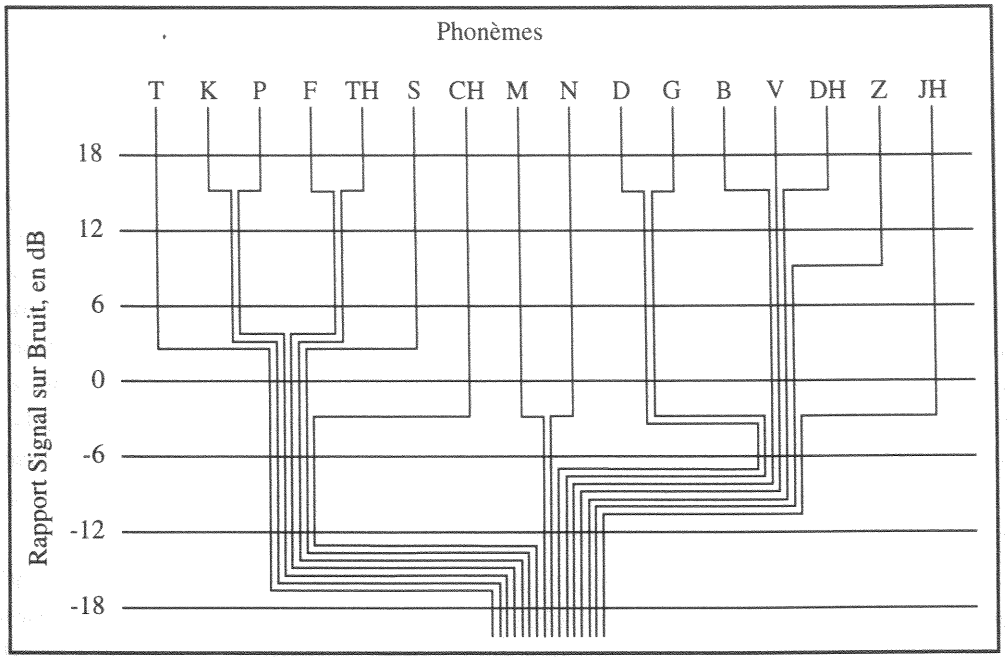


Figure 3.3 : Graphe des confusions progressives entre les consonnes de l'anglais américain en fonction des rapports signal sur bruit en condition de bruit blanc (consonne placée devant la voyelle /a/, d'après [miller55]).

La perception de différents types de parole dans le bruit, faite par [pickett56] (figure 3.4), tend à prouver la bonne résistance d'une parole non déformée dans du bruit pour la communication homme-homme. La parole non déformée doit être entendue comme étant de la parole normale (entre 50 et 80 dB) qui n'est donc ni murmurée, ni trop amplifiée. Cette étude va à l'encontre d'autres ([dreher57], [summers88]) et montre toute la difficulté qu'il y a à traiter de la parole subissant l'effet Lombard.

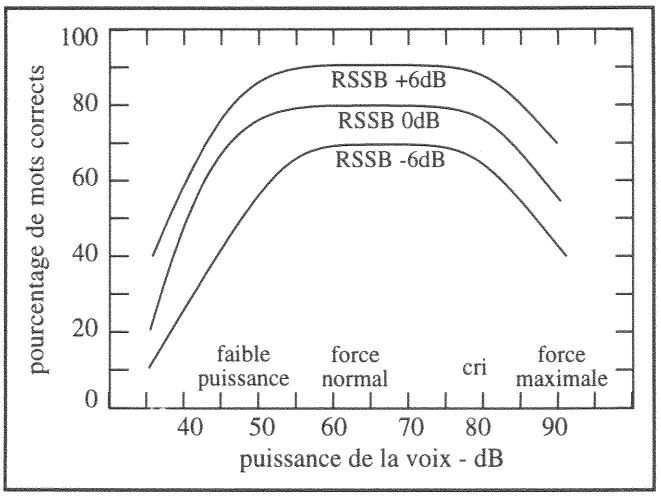


Figure 3.4 : Relation entre l'intelligibilité et la puissance de la voix. La puissance est mesurée à un mètre de l'orateur. Le bruit original a un spectre plat à une puissance de 70 dB. La parole préenregistrée est modifiée en puissance lorsqu'elle est ajoutée au bruit pour obtenir le RSSB donné sur chaque courbe (d'après [pickett56]).

L'homme possède, en outre, de très bonnes capacités discriminatoires lors de tâches des plus complexes telles que peuvent l'être les suivis de conversation au milieu de nombreux locuteurs (bruit connu sous le patronyme *cocktail party* en RAP). La figure 3.5 montre la capacité moyenne des auditeurs de l'expérience menée dans [miller47]. Un auditeur est en moyenne capable de reconnaître 80 pour cent des mots prononcés par son interlocuteur alors qu'un troisième interlocuteur parle aussi

fort que l'auditeur cible, ces 80% de mots étant compris en dehors de tout contexte sémantique.

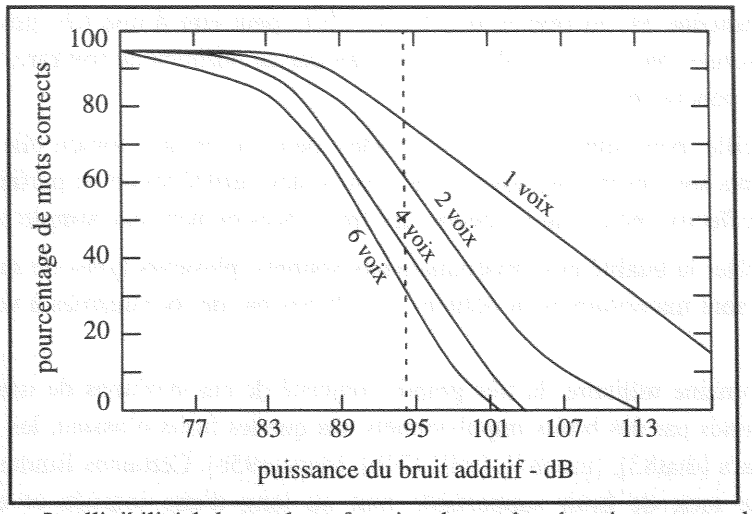


Figure 3.5 : Intelligibilité de la parole en fonction du nombre de voix masquant la voix cible. La voix cible a été maintenue à un niveau constant de 94 dB (d'après [miller47]).

Enfin, l'homme ne perçoit pas le bruit de la même manière en fonction de son activité quotidienne. Ses critères subjectifs de qualité ainsi que ses capacités de discrimination pourraient être grandement liés à sa profession. Dans une étude, [spieth56] a demandé à des auditeurs de juger le niveau de gêne provoqué par des bruits produits dans six plages de fréquences différentes. Les auditeurs ont ainsi, pour chaque plage, défini le seuil inférieur de puissance du bruit à partir duquel commençait la gêne. Spieth a ensuite divisé l'ensemble des auditeurs en deux groupes : il a donc calculé pour chaque plage de fréquences la moyenne du niveau de gêne pour les auditeurs travaillant en atelier ou en usine (milieu industriel) d'une part ainsi que la moyenne du niveau de gêne pour les auditeurs travaillant en milieu tertiaire. La différence entre ces deux moyennes n'est pas négligeable puisqu'elle varie approximativement entre 15 et 20 dB (figure 3.6).

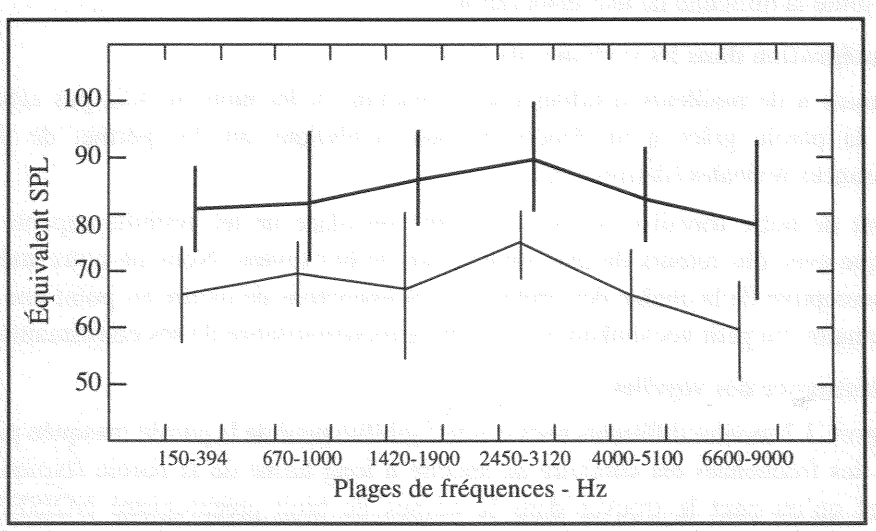


Figure 3.6 : Gène provoquée par des bruits de chacune des plages de fréquences listées en fonction de leur puissance (*Sound Pressure Level*). Le trait fort correspond à la moyenne obtenue sur un groupe de test travaillant dans un atelier alors que le trait fin correspond à la moyenne obtenue sur un groupe de test travaillant en bureau. Les barres verticales donnent une indication de l'intervalle de confiance à 95% (d'après [spieth56]).

Cette différence peut être vue comme le résultat d'une accoutumance qui pourrait elle-même être vue comme un apprentissage.

3.2.4.2/ Limites des capacités auditives humaines

L'oreille humaine est un organe très fragile. Elle peut être d'une très grande sensibilité. Ainsi certaines personnes ont l'«oreille absolue» et bien que ce phénomène soit rare, il prouve jusqu'à quel point l'oreille peut devenir efficace.

Mais l'oreille peut également se dégrader dans le temps lorsqu'elle est soumise à des environnements sonores trop agressifs. Il en résulte une surdité totale ou partielle, la surdité partielle pouvant être effective pour tout le spectre des fréquences ou pour une simple plage de fréquences.

Pour contrôler la qualité des environnements sonores, plusieurs types de mesures ont été définis. Ces mesures sont majoritairement définies et utilisées par des organismes à vocation industrielle ou militaire.

Dans le domaine militaire, la très grande majorité de ces systèmes de mesure s'intéressent aux dégâts provoqués par des bruits impulsionnels tels que les fusils d'assaut, les obusiers lourds ou les armes antichars [dtat83], [nato87], [mil1474b], [dancer95b]. Certaines limites ont ainsi été définies concernant le taux de bruit supportable tout au long d'une journée et d'autres pour le taux supportable pendant 8 heures d'exposition continue [dancer92].

Certaines des mesures dernièrement développées [dancer95a] se fondent sur un calcul de l'énergie acoustique exprimée en décibels suivant une loi A supportée par un sujet pendant 8 heures. Ceci se note L_{Aeq8}, la loi L_{AeqN} s'exprimant sous la forme [cammarata95] :

$$L_{AeqN} = 10\log\frac{1}{N}\sum_{i} 10^{(L_i/10)} \qquad \text{(Éq. 3.3)}$$

où N est le temps d'exposition en secondes et L_i est le niveau du son, en décibels, suivant la loi A mesurée sur une période de 1 seconde

Ce type de loi est équivalente aux standards définis pour l'industrie [iso1999]. Certains de ces types de mesures ont d'ailleurs eu une naissance difficile ([iso3741], [iso3742] et [iso3745]) ce qui montre toute la difficulté de leur élaboration.

3.2.5/ Intégration dans les systèmes de RAP

L'homme a de meilleurs résultats que la machine si les mots ne sont pas aléatoires. L'homme décode la parole grâce à un fonctionnement analytique qui lui permet de tirer parti de ses connaissances lexicales [dermody92].

Le but de notre travail n'est pas de mettre en place un tel système capable d'une recherche analytique avec des retours du haut niveau vers le bas niveau. Nous ne nous intéresserons qu'à la partie perceptive de la chaîne des traitements et essaierons de mettre au point une méthode capable de reconnaître un petit vocabulaire sur la simple reconnaissance de ses constituants phonétiques.

3.2.6/ Résistance des voyelles

La figure 3.7 montre différents tests d'intelligibilité pour de la parole masquée par un bruit dont le spectre des fréquences est similaire au spectre à long terme de la parole (typiquement du *babble noise* tel qu'on peut le trouver dans le corpus de bruit préenregistré NOISEX [varga92]). Ce graphique montre la très bonne résistance des lettres et des chiffres au bruit. Ce type de tâche provoque en effet une saturation de l'intelligibilité aux alentours du RSSB de -5 décibels (la saturation de l'intelligibilité signifie que l'intelligibilité sera toujours de 100 pour cent à des RSSB supérieurs à -5dB). D'après [steenek92b], ceci est dû au fait que :

- le nombre de mots est limité,
- la reconnaissance de ces mots dépend surtout des voyelles et peu des consonnes.

Dans ce même article, [steenek92b] note que les voyelles ont un SPL (*Sound Pressure Level*)

moyen plus élevé de 5dB que le SPL moyen des consonnes ce qui leur permet de mieux résister au bruit. La faible résistance des consonnes au bruit a d'ailleurs été prouvée (cf figure 3.3).

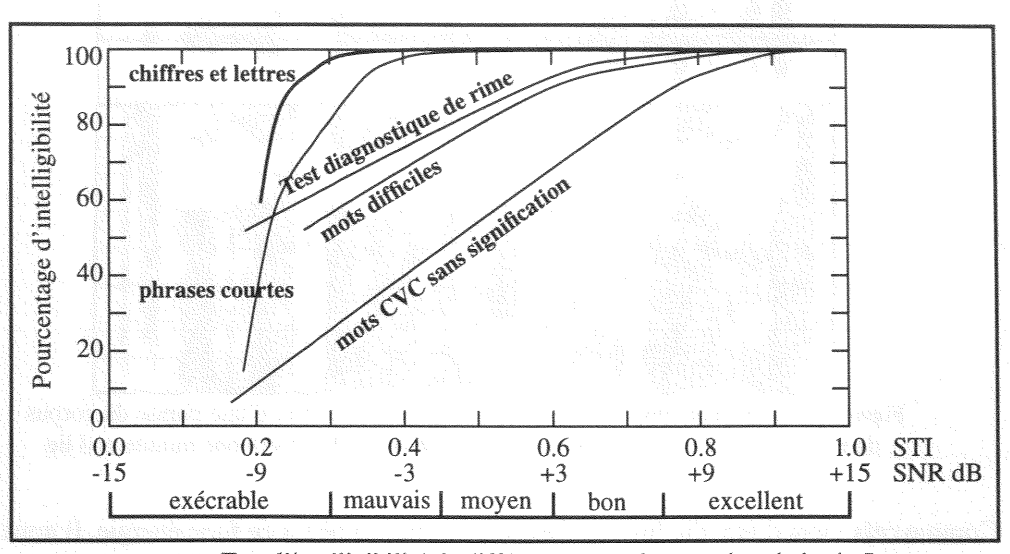


Figure 3.7 : Test d'intelligibilité de différents types de mots dans le bruit. Les pourcentages sont obtenus en demandant aux auditeurs d'estimer le nombre de mots correctement entendus (d'après [steenek92b]).

Cependant, la meilleure résistance des voyelles par rapport aux consonnes doit être relativisée. Certains bruits convolutionnels peuvent avoir des effets beaucoup plus forts sur les voyelles que sur les consonnes et donc interdire les traitements fondés sur le voisement. Une limitation de la bande passante pourra provoquer une dégradation très forte des taux de reconnaissance des consonnes sans pour autant affecter le taux de reconnaissance des voyelles [steenek92b]. Certaines consonnes, telles les fricatives, sont des phénomènes de haute fréquence et la limitation du nombre maximum de passages par zéro peut très fortement réduire leur signature. Mais d'autres distorsions non linéaires peuvent par contre provoquer l'effet inverse en dégradant très fortement les voyelles sans pour autant affecter les consonnes. C'est le cas lors d'une limitation de l'amplitude dans le signal (*peak clipping*). Ceci peut se comprendre en regardant un signal temporel et le spectrogramme associé dans la figure 3.8. L'énergie des formants des voyelles se remarque principalement dans le signal temporel par l'éloignement de certains pics par rapport à l'axe temporel qui représente également le zéro de l'amplitude. Le *peak clipping* correspond à la limitation de la hauteur de ces pics et donc, spectrographiquement parlant, à la disparition des plages de fréquences très énergétiques que sont les formants. Un *peak clipping* trop prononcé peut donc faire disparaître les voyelles du fait de l'élimination de ce qu'elles ont de plus remarquable.

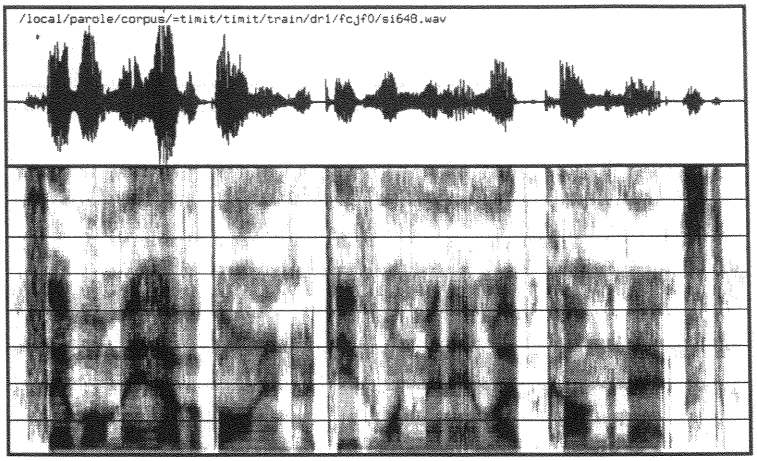


Figure 3.8 : Un signal temporel et son spectrogramme (extrait d'une phrase du corpus de parole TIMIT : «***A sailboat may have a bone in her teeth*** one minute and lie becalmed the next»).

Comme cela vient d'être dit, les voyelles sont des phénomènes de forte énergie. Il nous a donc tout d'abord semblé intéressant d'étudier les méthodes fondées sur des calculs d'énergie dans le signal pour essayer de localiser les voyelles avant toute phase de reconnaissance de manière à obtenir une liste d'îlots de confiance dans le signal de parole à analyser.

3.3/ Méthodes fondées sur des calculs d'énergie

3.3.1/ Présentation

Les travaux initialement entrepris lors de notre thèse nous ont porté à étudier des méthodes fondées sur le calcul d'une énergie de manière à isoler les îlots de voisement dans le signal de parole, ces îlots devant permettre de localiser les voyelles avant de les reconnaître. Les études ont été faites sur le corpus NOISEX pour des rapports signal-sur-bruit (RSSB) variant de -6 à 18 décibels, par incrément de 6 décibels, et sur le signal de parole non bruité fourni par NOISEX (le RSSB est alors infini, voir l'équation 3.1). Ces études nous ont conduit à toujours obtenir le même type de résultats, présentés de manière synthétique dans la figure 3.9, quelle que soit la méthode de calcul utilisée pour calculer l'énergie présente dans le signal de parole.

3.3.2/ Algorithme et résultats

Notre recherche heuristique des meilleurs paramètres de la fonction de calcul de l'énergie nous a posé un problème aux limites de l'intervalle des RSSB. En effet, si une méthode de calcul de l'énergie dans un cas moyen peut générer de bons résultats, l'application de cette même méthode aux cas extrêmes que sont les très faibles RSSB ne pourra permettre d'obtenir qu'une mauvaise segmentation du signal. Il est à noter qu'un RSSB est considéré comme faible à partir de 10 ou 6 décibels, selon les cas trouvés dans la littérature.

Il aurait donc fallu déterminer une méthode possédant des paramètres variables en fonction du RSSB. Mais ce rapport ne peut être déterminé de manière correcte que lorsque le message émis et le message reçu sont tous deux connus. Toute détermination du RSSB à partir du seul message reçu ne peut qu'entraîner des erreurs plus ou moins graves, qui sont fonction de la variabilité du bruit ambiant.

Nous avons donc essayé d'implanter une méthode se fondant sur la seule énergie. Cette méthode était, en outre, totalement indépendante du RSSB et du bruit lui-même pour les raisons ci-avant exposées. Ainsi, aucune connaissance relative à la phonétique n'était prise en compte. En particulier, la forme spectrale des voyelles n'était pas considérée alors que cette forme, composée de plusieurs formants, est la caractéristique spectrographique majeure de cette classe d'événements acoustiques (cf. figure 3.8).

L'algorithme générique effectue une sommation de l'énergie à partir du signal temporel, sommation immédiatement suivie de la soustraction de valeur minimale attendue de l'énergie, valeur déterminée heuristiquement ou par moyenne sur une partie du signal. Cette soustraction permet d'obtenir une courbe composée de méplats et de pics, les pics indiquant les endroits de forte énergie et donc de voisement probable. Enfin, une phase de lissage est effectuée pour éliminer les pics de très courte durée, ces pics ne pouvant pas correspondre à des voyelles puisque ces dernières sont des phénomènes de relativement longue durée. Le lissage est effectué soit par la méthode de la moyenne, soit par la méthode de la médiane.

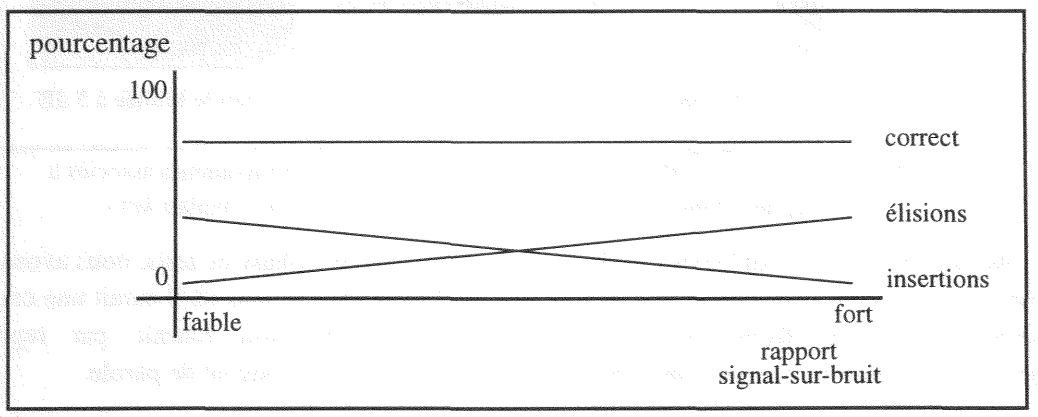


Figure 3.9 : Graphique type des résultats de reconnaissance des îlots de voisement dans la parole bruitée en fonction du rapport signal-sur-bruit.

3.3.3/ Inconvénients de la méthode

L'algorithme de segmentation basé sur le calcul de l'énergie possède plusieurs inconvénients qui ne peuvent être contournés :

- aucune connaissance phonétique n'est incorporée. Il est tout au plus possible de voir ce type de connaissance dans la phase de lissage puisque cette phase élimine les pics de faible durée, durée qui est en contradiction avec la durée des voyelles,
- il est quasiment inapplicable dans le cas où le RSSB est nul ou négatif puisque le signal du bruit est alors au moins aussi énergétique que le signal de parole, par définition de ce type de conditions,
- il ne permet pas d'obtenir de bons résultats dans le cas de bruits non stationnaires puisque l'énergie du bruit est alors changeante. Ce phénomène est particulièrement remarquable dans le cas des bruits de salle des machines ou de rafales de mitrailleuse de NOISEX (voir l'annexe 3, figures A3.6 et A3.9), ces bruits provoquant l'apparition de pics énergétiques.

Les deux dernières remarques méritent quelques explications supplémentaires qui peuvent être facilement comprises par rapport à la figure 3.10. Cette figure montre l'évolution d'un même signal temporel dans trois conditions de bruit différentes, le bruit utilisé étant un bruit blanc, signal parfaitement stationnaire puisque généré à partir d'un processus gaussien. Les trois conditions de bruit correspondent à un signal non bruité tout d'abord puis à un signal bruité ensuite, à 15 puis 5 décibels de rapport signal sur bruit. L'évolution du signal temporel montre l'accroissement de la force du bruit à mesure que le RSSB diminue. À un RSSB nul ou négatif, le signal de parole disparaîtra totalement. Il pourra cependant encore être visible dans la partie basse du spectrogramme pour le seule raison que le bruit blanc est moins agressif en basses qu'en hautes fréquences. Des bruits différents auraient cependant conduits à des remarques différentes. Certains bruits, moins stationnaires que le bruit blanc, auraient permis de continuer à distinguer la parole (quoique nous ayons ici une connaissance a priori de sa position dans le plan temporel) tandis que d'autres bruits, plus agressifs que le bruit blanc en basses fréquences, auraient par contre permis de distinguer la parole dans le signal temporel tout en le dégradant beaucoup plus fortement dans le spectrogramme.

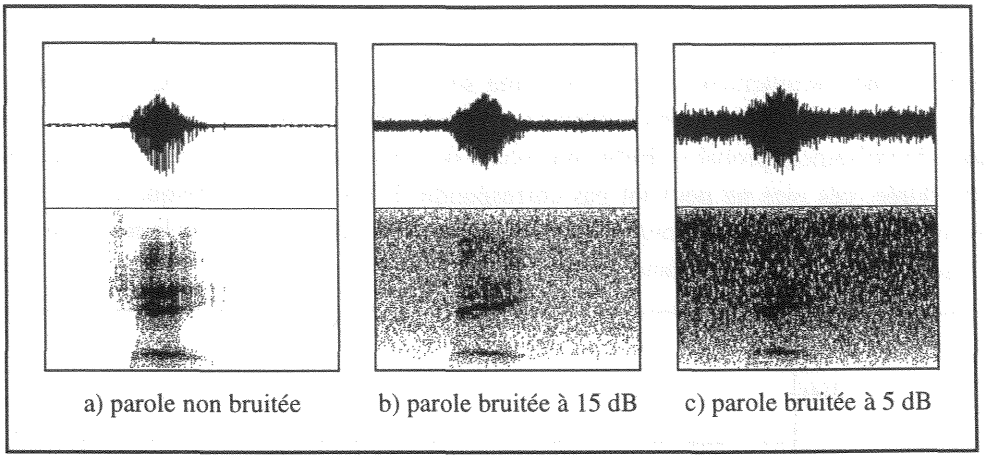


Figure 3.10 : Vue de différents signaux temporels et des spectrogrammes associés à ces signaux dans différentes conditions de bruit blanc (mot anglais *key*).

Ayant constaté ces problèmes lors de cette première et courte phase de tests, nous avons essayé de mettre en place un système qui soit un peu plus intelligent dans le sens où il aurait une connaissance phonétique de la forme des voyelles, que cette forme soit fournie par représentation spectrographique ou par toute autre représentation plus adaptée au signal de parole.

CHAPITRE 4 : DÉVELOPPEMENT AVEC LES PERCEPTRONS MULTICOUCHES

"Parler, parler, il en restera toujours quelque chose"
L. P. Beria

Résumé

Ce chapitre présente un système de reconnaissance automatique de petits vocabulaires prononcés de manière continue en milieu bruité. Ce système est fondé sur l'utilisation séquentielle et hiérarchique de plusieurs réseaux connexionnistes statiques tels que les perceptrons multicouches ou les Selectively Trained Neural Networks. Trois étapes successives ont été mises en place : elles correspondent à une première segmentation du signal, puis à la reconnaissance des voyelles et, enfin, à la reconnaissance des mots.

4.1/ Exposé du problème

4.1.1/ Besoin d'une connaissance phonétique

Comme nous venons de le voir au chapitre 3, paragraphe 3.3, un système de segmentation n'utilisant que la seule énergie présente dans le signal temporel est inapplicable au domaine de la reconnaissance automatique de la parole en milieu bruité. Cette inapplicabilité est principalement due au manque de connaissances phonétiques qui caractérise cette méthode, uniquement fondée sur un calcul mathématique. Il est, dans ce cas, impossible de distinguer, dans l'absolu, l'énergie du bruit ambiant de l'énergie du signal de parole. Cette distinction pourrait cependant être faite dans certains cas de bruits particuliers grâce à des méthodes ad hoc.

Pour pallier ce problème, il semble nécessaire de mettre en place une méthode de segmentation qui possède un minimum de connaissances en phonétique, ces connaissances devant surtout être fondées sur la forme que peuvent prendre les différents événements de la parole qui nous intéressent dans notre tâche : les voyelles (cf. chapitre 3, paragraphe 3.2.6). Avec de telles connaissances, le réseau de segmentation devrait être capable de distinguer les phonèmes par rapport au bruit où ces formes peuvent être noyées. Les figures 3.10.b et 3.10.c du chapitre 3 permettent de comprendre aisément ce fait puisqu'il est possible de distinguer, dans les spectrogrammes, les traces des formants de la voyelle.

La nécessité d'introduire des connaissances phonétiques se retrouve également dans les phases de reconnaissance des voyelles et des mots auxquelles nous ne nous sommes pas encore intéressés. Cette connaissance est nécessaire dans ces tâches de classification, directement en prise avec notre interprétation, perceptive et symbolique, du signal [harnad90].

4.1.2/ Critique de l'existant

Différentes recherches ont déjà été menées dans le domaine de la reconnaissance de petits vocabulaires. Ces études permettent généralement de valider, sur des tâches d'envergure limitée, de nouvelles architectures de reconnaissance des formes ou de nouvelles méthodes de prétraitement du signal. Elles sont également utiles pour des tâches limitées dans le cadre d'applications restreintes du concept général d'interface vocale homme-machine.

Avant de nous intéresser à différents travaux très proches des nôtres, nous allons rapporter ici quelques remarques faites lors de recherches s'intéressant à la phonétique et à son utilisation lors de l'étiquetage de corpus de parole préenregistrée. Une première étude aux résultats très intéressants a été présentée dans [phillips87]. Cette étude mesure la concordance entre l'étiquetage effectué par des experts en phonétique sur un corpus du DARPA, *Defense Advance Research and Project Agency*, et l'étiquetage effectué par deux auditeurs après écoute du même corpus. [phillips87] note ainsi que si les deux locuteurs étiquettent le corpus avec une concordance de 67%, la concordance entre l'expert phonétique et les auditeurs n'est que de 62%. La cause mise en avant pour expliquer cette différence est la coarticulation mais la définition phonétique des segments manuels semble également poser problème puisqu'un système automatique, développé vis-à-vis du corpus du DARPA, classe correctement les phonèmes 48% du temps avec l'étiquetage recueilli auprès des locuteurs alors que ce taux n'est que de 46% lorsque l'étiquetage original est utilisé. Une étude restreinte du corpus TIMIT effectuée dans [keating92] présente d'autres constatations relatives aux variabilités contextuelles et confirme ainsi toute la difficulté d'obtenir un étiquetage manuel correct et le problème posé par l'application de règles phonétiques strictes. Une autre étude intéressante a été effectuée par [cole92] et porte sur la capacité de perception par l'homme lors de tâches équivalentes à celles qui sont posées aux systèmes de RAP. Ainsi, des tests de perception de voyelles présentées hors contexte à des auditeurs quelconques n'ont permis d'obtenir que 54,8% de classification correcte, ce type de tâches est pourtant fréquemment réalisé par des méthodes mathématiques dont certaines ont réussi à obtenir jusqu'à 80% de classification correcte. Le même test de perception réalisé avec une présentation des voyelles au milieu d'un contexte restreint a permis aux auditeurs de reconnaître ces voyelles à hauteur de 65,8%. [cole92] précise enfin que la présentation d'un locuteur à un auditeur par écoute préalable d'une phrase complète permet d'obtenir de meilleurs taux de classification puisque l'auditeur peut déterminer les caractéristiques de la voix écoutée. Aucun résultat synthétique n'est cependant présenté dans ce cas. Enfin, plus près de nos préoccupations, [daly88] a effectué une étude de la tâche d'épellation au niveau acoustico-phonétique et au niveau lexical. Cette étude montre que la cause principale d'erreurs est le regroupement de plusieurs consonnes avec une même voyelle c'est à dire le rôle trop important joué par certaines voyelles dans l'épellation de nombreuses consonnes, comme nous le verrons au paragraphe 4.1.3.2. [daly88] note cependant que le nombre d'erreurs peut être réduit par l'emploi de la notion de perplexité, c'est à dire du facteur de branchement, qui impose des contraintes salvatrices même si l'implantation d'une grammaire est alors obligatoire.

La segmentation de la parole a été étudiée dans de nombreuses recherches. Ainsi, [aktas90] présente une comparaison de plusieurs méthodes de reconnaissance des formes appliquées à une tâche de segmentation en grandes classes qui sont les occlusives non voisées, les fricatives non voisées, les occlusives voisées, les nasales, les sonantes, les voyelles et le silence, l'étude étant réalisée en milieu non bruité. Les systèmes utilisés pour la segmentation sont deux réseaux de Markov, un d'ordre 1 et l'autre d'ordre 2, un système de classification fondé sur le maximum de vraisemblance, que l'auteur considère comme un modèle de Markov d'ordre 0, et un réseau connexionniste baptisé *Temporal Flow Model*, équivalent à un TDNN mais dont les contraintes d'égalités sur les poids auraient été abandonnées. Les meilleurs résultats de cette étude ont été obtenus par le HMM d'ordre 2, juste devant le TFM, ces deux modèles surclassant, dans l'ordre, le HMM d'ordre 1 et le système de classification fondé sur le maximum de vraisemblance. D'autres

systèmes de segmentation mettant en œuvre des perceptrons multicouches pourront être trouvés dans [bendiksen90], [ghiselli91] et [cohn91], ces systèmes ayant des résultats variables. [depuydt90] présente lui un système fondé sur l'emploi conjoint d'un perceptron et d'un modèle d'audition particulier et obtient des résultats moyens. [galiano91] présente un système de segmentation utilisant des unités infra-lexicales, c'est à dire plus petites que le mot, pour segmenter le signal de parole en grandes classes que sont les voyelles, occlusives, nasales, affriquées, fricatives, latérales et consonnes roulées. La méthode utilisée repose sur l'emploi d'un automate d'états fini stochastique et obtient des résultats plus moyens que ceux présentés dans [aktas90]. S'éloignant un peu des modèles présentés jusqu'à présent, [feng91] définit un système de segmentation utilisant des modèles adaptatifs mettant en place une mémoire à court terme et une mémoire à long terme. La détection des segments se fait par observations des changements dans le signal, le système reposant sur le principe que la parole est un phénomène localement stationnaire. Aucun véritable résultat n'est cependant présenté. Un autre axe des recherches en segmentation se fonde sur les méthodes de calcul de l'énergie incluant des mécanismes supplémentaires. [mak92] présente ainsi une méthode de segmentation des mots isolés, EPD-TFF. Cette méthode utilise la valeur de l'énergie du signal tout en confirmant ou infirmant les choix par l'utilisation d'indices temporels et fréquentiels. Cette méthode a été présentée de manière plus approfondie dans [junqua94b] où elle est favorablement comparée à d'autres méthodes telles que celle calculant l'énergie avec ajustement automatique du seuil, celle effectuant l'extraction du *pitch* en tenant compte des variations d'énergie, celle calculant l'énergie avec ajustement automatique au bruit ou encore celle calculant l'énergie avec un mécanisme de détection de la voix par *zero-crossing*. Cependant, tous les bruits étudiés dans [junqua94b] sont stationnaires. [hunt95] présente lui aussi une méthode de segmentation robuste au bruit se fondant sur la détection du voisement par calcul d'une autocorrélation spectrale d'ordre limité. Le peu de résultats ne permet cependant pas de juger de la qualité de la méthode.

Un affinement de la segmentation en classes comprenant chacune moins d'éléments que les "grandes classes" permet de s'orienter vers les systèmes de classification des phonèmes. [elenius91] présente ainsi un système de classification des phonèmes utilisant trois réseaux connexionnistes différents dont les résultats sont fusionnés logiquement. Un premier réseau permet de détecter les caractéristiques du signal. Ces caractéristiques sont des indices grossiers tels que le voisement, la nasalité ou la position avant, centrale ou arrière de la voyelle. Un deuxième réseau permet d'identifier la voyelle à l'échelle de la trame tandis qu'un troisième permet de détecter les frontières entre voyelles dans une fenêtre de 150 millisecondes. Les résultats sont variables en fonction de la nature du réseau. Ainsi, si la reconnaissance des voyelles est correcte et la détection des caractéristiques assez bonne, l'étape de segmentation est, elle, tout à fait moyenne. L'utilisation de plusieurs réseaux connexionnistes a également été étudiée par [pratt91] dans une optique différente pour la classification des phonèmes. Trois réseaux connexionnistes sont entraînés séparément à reconnaître les voyelles avec des couvertures différentes sur le signal. Cette couverture peut être de 35, 65 ou 135 millisecondes. Ces trois réseaux, ayant des couvertures différentes, sont ensuite regroupés au sein d'un unique réseau qui se voit adjoindre quelques unités supplémentaires pour assurer la cohérence de l'ensemble. Cette méthode obtient de bons résultats. D'autres systèmes d'identification de phonèmes employant plusieurs réseaux connexionnistes en parallèle pourront être trouvés dans [buhrke91], [zeng92] ou [koizumi94]. Un modèle connexionniste spécifique utilisant ce même paradigme de décisions prises en parallèle, le *Neural Tree Network*, a également été appliqué au problème de l'identification de phonèmes [sankar91b], [rahim92]. [leung90] présente également une méthode d'identification des phonèmes utilisant un réseau connexionniste et des indices particuliers mais n'obtient pas de très bons résultats bien que les indices semblent intéressants. [bennani91b] montre, quand à lui, la supériorité d'un modèle connexionniste mixant les paradigmes du TDNN et des cartes de Kohonen pour ce type de tâche. Toutes les méthodes que nous venons de voir reposent sur des méthodes standard de prétraitement du signal. Il est également possible

d'utiliser des modèles d'audition comme, par exemple, [muthusamy90] qui compare la représentation spectrographique au cochléagramme de Lyon [slaney88] pour la reconnaissance des 12 voyelles de l'anglais. L'utilisation de l'une ou l'autre des représentations complétée par des indices acoustiques tels que le *pitch* moyen, la durée de l'étiquette manuelle de TIMIT et l'amplitude relative, ne permet cependant pas d'obtenir de très bons résultats. Un autre modèle d'audition a été utilisé par [anderson91] avec de moins bons résultats. [richards92] présente, lui, un réseau connexionniste utilisant une toute autre philosophie. Son réseau se voit en effet imposer de redonner en sortie l'ensemble des informations mises à disposition en entrée. Ces données sont tout autant des indices articulatoires que des paramètres issus du modèle d'audition défini par Libermann et Mattingly et sont fournis sur une ou trois trames. En plus de devoir redonner ces données en sortie, le réseau se voit demander de classer un total de douze phonèmes observables dans le corpus d'apprentissage. Les résultats obtenus en mode locuteur indépendant sont assez moyens, de l'ordre de 70%. Cette idée de compression de l'espace des paramètres d'entrée a également été appliqué à l'identification des voyelles par [nakamura90]. Cet article présente un modèle connexionniste particulier, le *Phoneme Filter Neural Network*, qui permet lui aussi de réduire l'espace des paramètres. Ce réseau est divisé en deux parties, la première assurant une compression-décompression des paramètres tandis que la deuxième assure la classification. La première partie du réseau permet de transformer une trame en une autre plus proche d'un modèle faisant partie d'un corpus restreint. Toute trame présentée en entrée sera donc modifiée pour être rapprochée d'un élément d'un dictionnaire de formes standard. La deuxième partie du réseau assure l'identification à partir des formes du dictionnaire. [nakamura90] présente des résultats d'identification de voyelles. Les résultats obtenus sont cependant de plus mauvaise qualité que ceux obtenus avec un simple perceptron. Ces résultats ont été améliorés par la suite sur une tâche d'identification de phonèmes [nakamura91]. La compression des informations n'est pas la seule technique envisageable. [gong91] présente une méthode de classification de phonèmes fondée sur l'interpolation vectorielle non linéaire puis la comparaison d'une trame résultat à un corpus de références. Cette méthode regroupe deux notions très utilisées : les systèmes de prédiction non linéaire d'une part, l'interpolation vectorielle non linéaire subsumant des systèmes tels que les *Linked Predictive Neural Network* [tebelskis90], [tebelskis91], les *Neural Prediction Model* [iso90], [iso91] et les *Hidden Control Neural Network* et, d'autre part, les systèmes fondés sur la quantification vectorielle. Cette méthode obtient de bons résultats. [harrison89] présente, lui, un système de classification de phonèmes fondé sur l'emploi de deux réseaux connexionnistes consécutifs. Le premier réseau permet l'identification des unités infra-phonémiques tandis qu'un deuxième réseau est chargé de la reconnaissance des unités zonales. Ce deuxième réseau utilise un mécanisme de récurrence locale proche de celui exposé, ensuite, dans [vries90]. Ce réseau obtient de bons taux de reconnaissance sur les voyelles mais la reconnaissance des consonnes est de moins bonne qualité. Un autre réseau connexionniste récurrent, d'architecture proche de celle d'[elman90], a été présenté dans [robinson89]. Ce réseau a été appliqué au problème de l'identification des voyelles où il a obtenu d'excellents résultats puisque 80% des voyelles du corpus TIMIT peuvent être correctement reconnues [robinson94]. D'autres architectures connexionnistes ont également été appliquées au problème de la classification des phonèmes. [torkkola91b] utilise ainsi un modèle de Kohonen pour l'identification des phonèmes, les résultats issus de ce réseau étant ensuite traités par un perceptron. Ce réseau a obtenu de bons résultats sur le corpus étudié de même que celui présenté dans [mcdermott92] où le réseau de Kohonen utilisé s'est vu adjoindre une étape de DTW, permettant de classer correctement 80% d'un corpus *E-set*, type de corpus dont nous reparlerons. Le modèle connexionniste des *Radial Basis Functions* a également été appliqué à une tâche de classification de phonèmes et comparés aux résultats obtenus par un perceptron. [renals89] note que ces deux modèles connexionnistes ont des résultats équivalents mais que les perceptrons ont de meilleures capacités de généralisation. Des résultats similaires sont présentés par [burr92]. [renals89] note par ailleurs que ces deux types de réseaux connexionnistes ont, dans tous les cas, de

meilleures capacités que les réseaux de Markov d'ordre 1. Les réseaux de Markov ont été utilisés par [niles92] pour l'identification "temps réel", ou *spotting*, des trois consonnes /R/, /s/ et /t/. Les résultats obtenus sont assez bons mais l'intérêt de [niles92] réside surtout dans la présentation d'un algorithme d'apprentissage par correction des erreurs qui est assez proche des algorithmes utilisés pour l'apprentissage dans les réseaux connexionnistes récurrents. [chiba90] met lui aussi en place un *spotting* des consonnes par observation du signal temporel et des indices acoustiques par un perceptron où il obtient 85% de taux de reconnaissance sur huit consonnes. [shirai91] utilise une méthode de quantification vectorielle alliée à un dictionnaire de références pour effectuer une classification de phonèmes. Il repose sur l'observation de trois types d'indices : l'énergie, le ratio d'énergie et les coefficients issus d'une étape de codage par prédiction linéaire du signal. Ceci lui permet d'atteindre des taux de reconnaissance de 98% pour les voyelles et de 82% pour les consonnes sur un corpus de taille cependant limitée. Les méthodes employées dans cette dernière étude sont néanmoins d'un niveau différent de toutes celles que nous venons de citer puisqu'habituellement réservée à l'identification des mots.

[bourlard96a] note qu'une tâche de reconnaissance de petits vocabulaires se résoud le plus souvent par un modèle de mots et donc par une méthode globale. Le paradigme des méthodes globales prône l'analyse d'un phénomène dans son ensemble plutôt que par l'analyse de certains de ses constituants, cette deuxième option constituant la base du paradigme des méthodes analytiques. Rapporté au domaine de la RAP, ces deux paradigmes opposent donc l'analyse des mots à l'analyse des phonèmes ou de toute autre unité phonétique. Comme nous le verrons tout au long de ce chapitre, notre choix s'est, a contrario, porté sur une méthode analytique, s'intéressant à plusieurs caractéristiques des mots plutôt qu'au mot lui-même. Le paradigme des méthodes globales n'en est pas moins intéressant à étudier puisque certaines recherches ont été menées à partir d'excellentes idées. [english92] a ainsi étudié le problème de la reconnaissance des mots en temps réel, ou *word spotting*, en utilisant trois modèles connexionnistes différents. La première étape de son système utilise une carte de Kohonen, les sorties de cette carte sont ensuite traitées par une couche d'unités connexionnistes récurrentes avant que ces activations ne soient finalement classées par un perceptron multicouche possédant des unités gaussiennes et synaptiques. Le but de ce réseau était d'identifier les dix chiffres. La couche de sortie comprenait elle-même onze sorties, la onzième sortie, baptisée "*don't care*", permettant de classer tout phénomène acoustique ne correspondant pas à un des dix chiffres. Avec cette architecture, l'auteur a pu obtenir un taux de reconnaissance en monolocuteur de 97,5% en test, l'apprentissage ayant permis d'obtenir 100% de reconnaissance correcte. L'utilisation intégrée de plusieurs réseaux connexionnistes a également été étudiée par [yamaguchi90] avec une méthode beaucoup plus simple. Le système est dédié à l'identification de mots isolés indépendamment du locuteur et utilise plusieurs réseaux de neurones successifs, du premier niveau composé d'*event-nets* au dernier baptisé *super-net* en passant par les *word-nets*. L'auteur obtient ainsi un taux d'identification correcte de 97% sur un ensemble de 10 chiffres et de 18 mots de contrôle. [franzini89] utilise par contre un seul et même réseau, celui-ci réalisant deux tâches de manière concurrente : l'identification des phonèmes d'une part et l'identification des 11 chiffres de l'anglais d'autre part. Toutes les informations de sortie sont exploitées par un processus de niveau supérieur. Entre autre particularité, le réseau connexionniste exploite l'information à partir d'une couche cachée de premier niveau synthétisant l'information en provenance de l'entrée mais également à partir de la copie de cette couche cachée sur les 10 pas de temps précédents. Une deuxième couche cachée permet de synthétiser encore toutes les activations avant que celles-ci ne permettent de calculer les deux types de réponse souhaités. Un développement de cette architecture a été présenté dans [franzini92]. Cette dernière architecture essaie de réaliser, à un même niveau, une identification de phonèmes et une identification de mots. L'idée de reprendre un système d'identification de phonèmes par réseau connexionniste pour réaliser l'identification des mots a été étudiée par [lucke92] qui a, pour ce faire, mis en place des représentations compositionnelles.

D'autres études ont tenté de marier le paradigme connexionniste à d'autres méthodes plus anciennes. [morgan91b] étudie ainsi la possibilité d'utiliser la DTW avec un réseau neuromimétique tandis qu'[hataoka90] étudie l'exploitation des sorties d'un réseau connexionniste par un système d'inférence floue et un réseau de concepts. [ney91] étudie l'identification des mots par le biais de HMM et de *Radial Basis Functions*, la palme de l'originalité revenant à [carey91] qui considère un ensemble de HMM comme définissant un réseau connexionniste pour y appliquer l'algorithme de rétropropagation du gradient d'erreur. [zhu90] présente lui une méthode de classification de mots fondée sur l'extraction d'indices et la quantification vectorielle par dictionnaire de références à partir de coefficients MFCC, 24 trames contenant toutes ces informations étant ensuite traitées par un perceptron. Cette méthode a permis d'obtenir un taux de 90% d'identification correcte sur un corpus de 22 mots, des chiffres et des mots de contrôle, prononcés par 11 locuteurs différents. [dubois91] étudie lui aussi le problème de représentation et constate que les résultats obtenus par son HMM chargé d'identifier les chiffres s'améliorent lorsque les données d'entrée sont des coefficients dynamiques. Des coefficients dynamiques d'ordre 2 permettent d'améliorer encore les résultats obtenus avec les coefficients dynamiques d'ordre 1, prouvant que ces coefficients temporels limités sont utiles dans ce type de tâche. [fanty90] s'intéresse lui à la reconnaissance des lettres de l'alphabet et plus particulièrement à l'identification des lettres épelées du *E-set*. Sa méthode fait appel à divers indices acoustiques, à une normalisation spectrale entre -1 et +1 et à un calcul de trames avec un *shift* pouvant descendre jusqu'à 3 millisecondes, ce qui est très peu. Le taux de reconnaissance est cependant très bon. [husoy91] étudie lui aussi le problème du *E-set* en comparant les perceptrons et le TDNN. [husoy91] tire plusieurs conclusions de son étude. Il remarque tout d'abord que le nombre de trames doit être important ou faible mais qu'un choix intermédiaire n'amène qu'une dégradation des résultats. Il remarque ensuite que le TDNN obtient de meilleurs résultats que toute architecture reposant sur les perceptrons, les perceptrons eux-même ayant de meilleurs résultats avec un grand nombre d'unités en couche cachée. [husoy91] note enfin que les meilleurs résultats sont obtenus lors de l'utilisation de coefficients statiques et dynamiques du premier ordre, que ces coefficients soient spectraux ou cepstraux. La reconnaissance de mots en milieu bruité est également d'un grand intérêt. [ramesh91] s'intéresse ainsi à la reconnaissance de chiffres connectés en parole spontanée. Il utilise un système fondé sur des coefficients LPC, utilise un HMM et une grammaire pour la vérification de numéros de cartes de crédit et obtient un taux d'identification correcte de 97%, ce taux étant de 86% sans usage de la grammaire. [dobler92] étudie l'identification de chiffres connectés en milieu bruité par HMM. Il met en place un filtre passe-haut qui permet de supprimer les composantes stationnaires du bruit, en augmentant donc le RSSB, et tente d'améliorer la modélisation de la durée des chiffres grâce à une modélisation différente du premier, du dernier et des chiffres intermédiaires. Enfin, [unnikrishnan91] utilise des réseaux connexionnistes avec lignes de délais encastrés, qui ne sont cependant pas des TDNN, pour résoudre le problème de la reconnaissance des chiffres connectés en milieu bruité. Il obtient de très bons résultats sur un corpus de taille limité et montre l'intérêt de l'apprentissage multibruit avec de la parole propre et de la parole bruitée et montre les très mauvais résultats d'un système dont l'apprentissage est fait en ambiance calme et dont les tests sont fait en milieu bruité comme celà a été rapporté par ailleurs [gong95].

Notre choix s'est porté sur une méthode développée au sein de notre laboratoire qui a déjà été utilisée avec succès sur de petits vocabulaires de mots épelés de manière isolée. Cette méthode utilise les STNN, *Selectively Trained Neural Networks* [anglade92a]. Nous avons choisi de développer un système fondé, entre autres méthodes, sur celle-ci puisque les résultats obtenus en reconnaissance de mots isolés sont de bonne qualité [anglade93].

4.1.3/ Architecture envisageable

Ce paragraphe nous permet de présenter une première approximation de l'architecture connexionniste que nous avons choisi de mettre en œuvre. Nous allons parler de deux processus différents qui serviront de base conceptuelle au système à venir : la segmentation de la parole et la

reconnaissance des mots.

4.1.3.1/ Segmentation

La première étape que nous voudrions isoler est l'étape de segmentation. Comme nous venons de le voir à la fin du chapitre 3, cette étape nécessite de mettre en place un système possédant de connaissances en phonétique puisque l'exploitation des seules informations d'énergie présentes dans le signal ne permet pas d'élaborer une méthode vraiment fiable en milieu bruité. Le choix pour la mise en place d'une méthode de reconnaissance de petits vocabulaires possédant explicitement une phase de segmentation s'explique par la résistance reconnue des voyelles au bruit qui permet de trouver des points d'ancrage fiables. En effet, comme cela a déjà été rapporté au chapitre 3, paragraphe 3.2.6, [steenek92b] note que les voyelles ont un niveau de pression (SPL, *Sound Pressure Level*) moyen plus élevé de cinq décibels que le SPL moyen des consonnes. Cette caractéristique permet aux voyelles de résister plus longtemps au bruit puisque leurs caractéristiques principales, les formants, ont un rapport signal sur bruit spécifique plus élevé que l'ensemble des événements phonétiques de la phrase. Et comme nous l'avons déjà fait remarquer auparavant, ce fait peut être facilement observé dans les figures 3.10.b et 3.10.c du chapitre 3. Il est par ailleurs intéressant, voire même amusant, de rapprocher la valeur de ce SPL moyen de cinq décibels du rapport signal sur bruit à partir duquel les lettres et les chiffres épelés commencent à être mal perçus par l'homme. Ce rapport signal sur bruit est de cinq décibels négatifs comme l'a également montré [steenek92b]. L'article que nous venons de mentionner ne précise cependant pas si le SPL moyen des voyelles a été déterminé en fonction des taux de reconnaissance des voyelles ou s'il l'a été à partir d'une étude calculant un rapport signal sur bruit segmental. Le rapprochement est pourtant, à notre humble avis, très intéressant.

La phase de segmentation du signal n'est pas une tâche nécessitant a priori beaucoup de connaissances contextuelles. La segmentation devrait donc pouvoir se faire avec une méthode qui n'utilise que des connaissances disponibles à l'instant t du processus. Mais la critique de l'état de l'art que nous avons réalisée au paragraphe 4.1.2 montre qu'il y a tout avantage à utiliser des informations issues d'un contexte de taille variable. La méthode que nous développerons utilisera donc plusieurs trames issues de la phase de prétraitement.

Cette phase de segmentation identifiant les voyelles permettra d'obtenir une liste d'îlots de confiance, ces îlots permettant d'appliquer le processus de reconnaissance des mots en des endroits du signal où le résultat aura une signification. Nous économisons ainsi une mise en œuvre en parallèle de ces deux phases du traitement, parallélisation qui aurait été possible en utilisant une phase de validation par conjonction des résultats et au prix d'une utilisation accrue de la puissance de calcul de la machine utilisée.

4.1.3.2/ Reconnaissance des mots

La deuxième étape nécessaire à notre système de reconnaissance de petits vocabulaires est l'étape de reconnaissance des mots du vocabulaire considéré. Les problèmes qui nous sont posés sont beaucoup moins complexes que ceux qui devraient être pris en compte dans le cas de vocabulaires de grande taille. Notre sujet de thèse se restreint aux seuls chiffres et lettres épelés et le vocabulaire est de ce fait très limité et toute étape de prise en compte d'une grammaire peut être abandonnée. Il n'y aura donc aucune étape de vérification lexicale dans notre système alors que cette vérification permet d'obtenir de meilleurs résultats lorsqu'elle est utilisée (cf. paragraphe 4.1.2). Notre tâche a, cependant, une optique a priori généraliste.

La tâche qu'il nous est demandé de résoudre est simple du point de vue du vocabulaire. Les chiffres épelés sont, par exemple, au nombre de dix, le nombre de mots à reconnaître étant, bien sûr, équivalent. La résistance intrinsèque des voyelles au bruit nous a poussé à décider de la mise en place d'une première étape de segmentation permettant d'isoler des points d'ancrage pour la phase de reconnaissance. En partant de ce type d'informations, la phase de reconnaissance pourrait être

faite de deux manières. Il serait ainsi possible de développer une phase de reconnaissance de mots utilisant une large fenêtre sur le signal de manière à avoir accès à l'ensemble du mot c'est à dire à la voyelle et à son contexte. Notre tâche correspondrait alors à un agglomérat d'une phase de détection (*spotting*) des instants de voisement intéressants dans le signal et à l'application d'une phase de reconnaissance de mots utilisant une technique similaire à celle du *spotting*. L'inconvénient majeur de cette méthode est la grande rigidité de la fenêtre d'analyse mise en œuvre qui doit a priori être adaptée à tous les rythmes d'élocution et doit donc être maximisée, nécessitant par là même d'utiliser plus de coefficients qu'il faudra définir, au préalable, par apprentissage.

La méthode de reconnaissance dont nous venons de parler est une méthode globale, analysant en une fois un signal étendu. L'autre paradigme existant en reconnaissance des formes est celui des méthodes analytiques, étudiant une forme non comme une seule et même entité mais comme une suite de formes plus restreintes dans le temps ou l'espace. Nous allons maintenant voir comment ce type de paradigme, cette deuxième manière de faire, pourrait être appliqué et quels avantages peuvent en être tirés.

La table 4.1 montre la manière dont sont prononcés les dix chiffres en langue française et donne une transcription phonétique de ces prononciations. Un fait intéressant à remarquer est la grande diversité des voyelles employées. Ainsi, ce sont sept voyelles qui sont utilisées pour la prononciation des dix chiffres. Cette constatation permet de conclure que la seule connaissance de la voyelle permet de reconnaître quatre des dix chiffres, les six chiffres restant se partageant, par couple, les trois voyelles restantes. Ces trois couples sont "un" et "cinq", "trois" et "quatre" et, enfin, "six" et "huit". Pour résoudre le problème des trois voyelles employées dans deux chiffres différents, une étape supplémentaire de reconnaissance devient nécessaire. Cette étape devrait utiliser une connaissance a priori de la position de la partie du signal autour de la voyelle permettant de discriminer le chiffre en question. Ainsi, pour distinguer "un" de "cinq", il faudrait soit identifier le phonème /s/ avant la voyelle, soit identifier le phonème /k/ après. La présence de l'un ou l'autre permettrait d'identifier un "cinq" alors que l'absence des deux permettrait de conclure, aux erreurs de classification près, à la présence d'un "un". Des processus de nature identique pourraient être mis en place pour les autres couples de chiffres.

Chiffre	Prononciation française	Transcription phonétique
0	zéro	*zeRo*
1	un	ɛ̃
2	deux	*dø*
3	trois	*tRwa*
4	quatre	*katR*
5	cinq	*sɛ̃k*
6	six	*sis*
7	sept	*sɛt*
8	huit	*ɥi(t)*
9	neuf	*nœf*

Table 4.1 : Transcription phonétique API des chiffres épelés en langue française.

La constatation qui vient d'être faite pour le français peut également être faite pour l'anglais. Ainsi, en observant la table 4.2 qui donne la prononciation anglaise des dix chiffres et la transcription phonétique de ces chiffres en alphabet phonétique ARPABET, il est aisé de constater que, cette fois encore, sept voyelles différentes sont utilisées. A contrario des observations que nous avions faites sur le français, certaines voyelles se retrouvent maintenant dans plus de deux chiffres. Il

existe donc un triplet "*zero*", "*three*" et "*six*" et deux couples : "*zero*, *o*" et "*four*" d'une part et "*five*" et "*nine*" d'autre part. Mais, comme pour le français, la reconnaissance de quatre des sept voyelles permet la reconnaissance immédiate de quatre des dix chiffres, les autres voyelles nécessitant, cette fois encore, une phase d'analyse supplémentaire.

Chiffre	Prononciation anglaise	Transcription phonétique
0	zero, o	Z IH R AO, AO
1	one	W AH N
2	two	T UH
3	three	TH R IH
4	four	F AO R
5	five	F AY V
6	six	S IH K S
7	seven	S EH V AX N
8	eight	EY T
9	nine	N AY N

Table 4.2 : Transcription phonétique ARPABET des chiffres épelés en langue anglaise.

Un raisonnement similaire à celui que nous venons de tenir peut être adopté pour les lettres épelées qui sont au nombre de ... vingt six. La table 4.3 donne la liste des vingt six lettres de l'alphabet, leur épellation en langue française et la transcription phonétique correspondante. Il est aisé de constater que, bien que le nombre de lettres soit supérieur au nombre de chiffres, le nombre de voyelles utilisées dans la constitution des vingt six mots est cette fois restreint vis-à-vis du nombre de mots. Il est ainsi possible de comptabiliser un total de neuf voyelles différentes dans la table 4.3. Ce nombre peut paraître faible par rapport aux vingt six mots mais il faut rappeler que le français ne comporte que douze voyelles dont certaines sont, finalement, acoustiquement très proches, sans même tenir compte des accents régionaux, limitant ainsi l'emploi possible de l'ensemble (cf. chapitre 1, table 1.1). Cependant, et de même que pour les chiffres, la connaissance de la voyelle permet encore, dans certains cas proportionnellement plus restreints, d'avoir une connaissance immédiate de la lettre épelée qui a été prononcée. C'est ainsi le cas pour "E" avec le phonème /ø/ ou pour "O" avec le phonème /o/.

Lettre épelée	Prononciation française	Transcription phonétique
"A"	a	*a*
"B"	bé	*be*
"C"	cé	*se*
"D"	dé	*de*
"E"	eu	*ø*
"F"	ef	*ɛf*
"G"	gé	*ʒe*
"H"	ach	*aʃ*
"I"	i	*i*
"J"	gi	*ʒi*
"K"	ka	*ka*
"L"	el	*ɛl*
"M"	emm	*ɛm*
"N"	enn	*ɛn*
"O"	o	*o*
"P"	pé	*pe*
"Q"	cu	*ky*
"R"	err	*ɛR*
"S"	ess	*ɛs*
"T"	té	*te*
"U"	u	*y*
"V"	vé	*ve*
"W"	double vé	*dublœve*
"X"	ix	*iks*
"Y"	i grec	*igRɛk*
"Z"	zèd	*zɛd*

Table 4.3 : Transcription phonétique API des lettres épelées en langue française.

Une fois encore, les constatations qui ont été faites pour le français peuvent être faites pour l'épellation de l'alphabet en anglais. En particulier, les lettres prononcées en français avec la voyelle *e* sont identiquement regroupées en anglais et prononcées avec la voyelle IH. Toutes ces lettres sont d'ailleurs regroupées dans ce qui est appelé le "*E-set*" (prononcer i sept) dont nous avons déjà parlé [fanty90]. Il pourrait également être possible de parler d'un "*EH-set*" (prononcer è sept) pour d'autres lettres épelées [daly88], la taille du "*E-set*" étant cependant supérieure à celle du "*EH-set*".

Les remarques qui viennent d'être faites sur la phase de reconnaissance des mots et l'avantage que nous voyons à tirer parti de la présence de nombreuses voyelles, en particulier dans le cas des chiffres, nous pousse à développer un système qui ne met pas en œuvre le paradigme des méthodes globales mais plutôt celui des méthodes analytiques. Nous avons donc opté pour une reconnaissance des mots effectuée en deux étapes, une première étape permettant de reconnaître la voyelle et la deuxième étape permettant de reconnaître le mot dans le cas où la reconnaissance de la voyelle ne suffit pas.

4.2/ Description du système

Le système mis en œuvre pour l'analyse de petits vocabulaires prononcés de manière continue en milieu bruité repose ici sur un enchaînement de trois niveaux. Par rapport à l'étude qui vient d'être

faite sur les étapes de segmentation et de reconnaissance des mots (paragraphe 4.1.3), nous allons successivement étudier :

- l'étape de segmentation du signal identifiant le voisement ou non de la parole (paragraphe 4.2.1) en ambiance calme ou bruitée,
- l'étape d'identification des voyelles (paragraphe 4.2.2) en fonction des segments obtenus dans la phase de traitement précédente,
- l'étape d'identification des mots qui permet de lever l'indécision dans le cas où la reconnaissance de la voyelle ne permet pas de reconnaître le mot (paragraphe 4.2.3).

Comme nous allons le voir dans les paragraphes qui suivent, nous avons décidé d'utiliser des modèles connexionnistes statiques du type du perceptron pour résoudre ces différentes tâches.

4.2.1/ Segmentation du signal

La phase de segmentation du signal doit nous permettre de trouver les parties voisées du signal dans diverses conditions de bruit. Le terme voisé est ici à prendre avec mesure puisque nous avons aussi bien étudié des segmentations des voyelles, signaux vocaliques, que des segmentations des phonèmes fortement voisés, ce dernier ensemble comprenant les voyelles tout autant que les semi-consonnes par exemple.

Le bruit étant notre milieu d'étude, nous avons eu à choisir entre une méthode mettant en œuvre une amélioration du signal par calcul d'une moyenne du spectre ou du cepstre du bruit, c'est à dire une méthode de *speech enhancement*, et une méthode traitant un signal en ne se préoccupant pas du bruit éventuellement présent dans le signal. Comme nous l'avons déjà fait remarquer au chapitre 3, la première méthode a comme désavantage d'être dépendante du milieu de bruit utilisé pour sa définition et limite ainsi la généralité du système lui-même, ce qui va à l'encontre de nos contraintes d'implantation.

Il nous a donc fallu définir une méthode de segmentation qui possède une qualité intrinsèque de résistance au bruit tout en respectant la simplicité du schéma de principe présenté à la figure 4.1. Ainsi, en plus de segmenter les voyelles, phénomènes résistants, nous avons décidé d'utiliser un réseau connexionniste de type perceptron pour effectuer cette segmentation, les réseaux connexionnistes ayant montré à l'occasion de plusieurs recherches leur résistance intrinsèque au bruit, voire leurs capacités de débruitage.

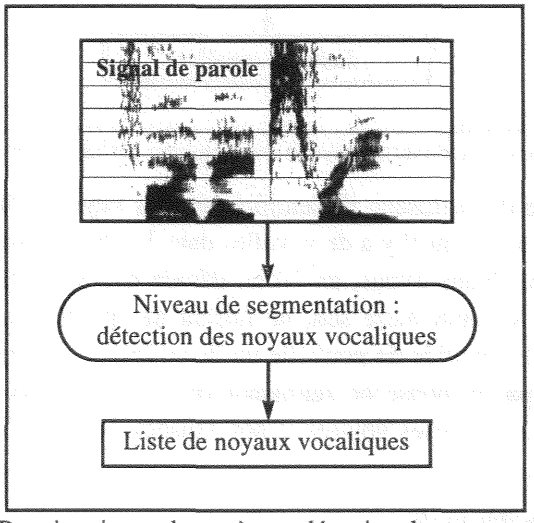


Figure 4.1 : Premier niveau du système : détection des noyaux vocaliques.

Le choix d'un modèle connexionniste ne permet cependant pas de connaître l'architecture employée in fine. Tenant compte des études qui ont été faites par ailleurs, nous avons choisi de définir un réseau utilisant plusieurs trames en entrée, de manière à étendre le champ de vision, ou de

prise en considération, du réseau. Ce choix semble, en effet, être gage d'une certaine qualité finale.

Les résultats issus de la phase de segmentation risquent, par ailleurs, d'être parfois erratiques du fait de conditions particulières très localisées. La courbe obtenue par segmentation connexionniste sera donc lissée pour tenir compte de la tendance de l'évolution, étendant et renforçant ainsi la vision temporelle du réseau qui n'a accès qu'à plusieurs trames.

4.2.2/ Reconnaissance des voyelles

La liste des segments vocaliques ayant été obtenue, il est désormais possible d'effectuer une classification acoustico-phonétique du signal de parole. Cette classification acoustico-phonétique repose sur le résultat de la segmentation (cf. figure 4.2) et étudie le signal par rapport aux frontières des segments qui ont été isolés. Ainsi, pour la reconnaissance des voyelles, le segment sera étudié en son milieu, pour maximiser la probabilité de positionner le réseau dans la partie la plus stable de la voyelle, et en son début, ce qui permet d'appréhender les effets de la coarticulation.

Le type de réseau connexionniste mis en œuvre dans cette étape n'est pas un perceptron mais un *Selectively Trained Neural Network*. La particularité de ce type de réseau réside dans le positionnement des trames d'entrée qui se fait en tenant compte de la courbe d'énergie. Ainsi, si le positionnement en milieu de segment se fait naturellement, le positionnement en début de voyelle se fait par observation de la courbe d'énergie du segment et de son contexte antérieur. La valeur minimale et la valeur maximale sont calculées avant que ne soit déterminé un ratio qui est fonction de ces deux valeurs. L'endroit de la courbe d'énergie dont la valeur correspond au ratio est alors choisie comme position de calcul des trames de début. Ce mécanisme permet de choisir une position qui est indépendante du temps et donc du rythme d'élocution.

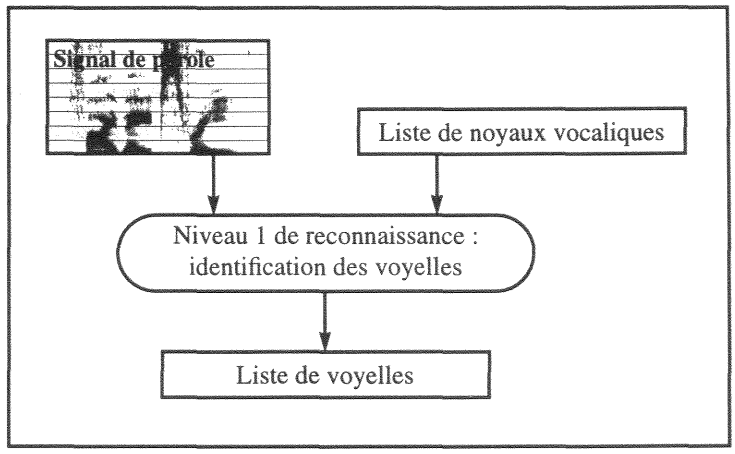


Figure 4.2 : Deuxième niveau du système : identification des voyelles.

La couche de sortie de ce réseau d'identification des voyelles doit, bien sûr, posséder au moins autant de neurones de sortie qu'il y a de voyelles dans le vocabulaire étudié puisqu'aucun codage de la sortie n'est effectué. Nous avons, en outre, décidé d'y ajouter une sortie supplémentaire pour classer les phonèmes, ou tout autre son, ne faisant pas partie des voyelles du vocabulaire. Cette sortie, qui pourrait être qualifiée de poubelle ou de collecteur, devrait nous permettre de traiter tous les segments insérés par la phase de segmentation, ces segments correspondant à des erreurs de classification du signal en parties voisées et non voisées. Il s'agit donc d'une sorte de procédure de récupération d'erreurs.

4.2.3/ Reconnaissance des mots

Si l'identification de la voyelle ne suffit pas à reconnaître un mot, c'est à dire s'il existe plusieurs mots dans le vocabulaire utilisant une même voyelle, il faut une étape supplémentaire de discrimination utilisant un réseau spécifique et un positionnement de la fenêtre d'analyse du réseau spécifique à la voyelle. Cette dernière étape de reconnaissance utilise donc l'information acquise lors

de la première étape de segmentation pour connaître les bornes de la voyelle étudiée et l'information acquise dans la deuxième étape pour savoir quel réseau spécifique utiliser, lorsque, bien sûr, il existe plusieurs voyelles du vocabulaire employées chacune dans plusieurs mots.

Le positionnement de la fenêtre d'analyse du réseau est spécifique à chaque voyelle. Cette position doit en effet être déterminée en fonction des connaissances en phonétique vis-à-vis des mots pour maximiser les capacités de discrimination du réseau. Ce choix se fera donc en fonction des phonèmes qui peuvent être observés en position antérieure ou postérieure et de leur éventuelle résistance au bruit.

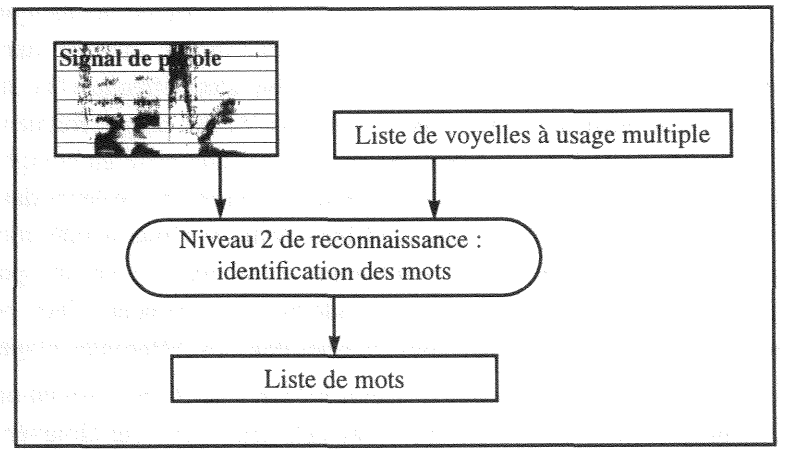


Figure 4.3 : Troisième niveau du système : identification des mots.

Le résultat obtenu dans cette phase d'identification des mots est direct et définitif puisque, comme nous l'avons fait remarquer précédemment, aucune phase de vérification lexicale ou grammaticale n'est mise en place dans notre système.

4.2.4/ Prétraitement du signal de parole

Une des caractéristiques principales du système n'a pas encore été présentée. Il s'agit de la phase antérieure à toute segmentation et à toute identification : le prétraitement du signal. Cette phase est importante car le choix pour l'une ou l'autre des méthodes possibles permet d'obtenir des résultats de plus ou moins bonne qualité [cooke93]. La méthode existante la plus intéressante et la plus reconnue dans la communauté de recherche en RAP lorsque nous avons débuté cette thèse était la méthode des coefficients cepstraux à échelle Mel, MFCC, *Mel Filter Cepstral Coefficients*. Cette méthode est relativement ancienne [davis80] mais est, en tout état de cause, plus récente que la méthode de la transformée de Fourier. Son emploi très large dans différentes recherches nous permettait d'emblée de pouvoir comparer nos résultats d'un point de vue architectural, en occultant totalement les considérations de représentation du signal de même que l'emploi de corpus de parole très répandus nous permettait de nous abstraire des problèmes de qualité de prise du son ou de segmentation manuelle.

Nous voyons deux avantages à l'emploi de la méthode des MFCC qui permet, en outre, d'obtenir une information synthétique sur le signal de parole de meilleure qualité que la transformée de Fourier tout en utilisant un espace de représentation plus restreint. La première qualité de la méthode MFCC est sa résistance reconnue au bruit. Certaines études ont ainsi montré que les MFCC étaient plus résistants que les coefficients log-spectraux avec lesquels ils partagent pourtant quelques caractéristiques computationnelles [mellor93]. La deuxième qualité majeure de la méthode MFCC est sa plausibilité biologique puisqu'elle utilise une échelle psychoacoustique des fréquences similaire à celle de l'oreille interne.

L'appareil auditif humain a la propriété de se comporter comme un banc de filtres qui se chevauchent. Ces filtres peuvent être modélisés de façon triangulaire, avec une fréquence centrale et

une largeur de bande appelée bande critique. Une bande critique définit une bande de fréquences pour laquelle le seuil d'audition d'un son change soudainement lorsque la modification en fréquence du son dépasse les limites de la bande. La largeur d'une bande critique d'un modèle inspiré de l'audition humaine, quelle que soit sa fréquence centrale, est appelée un Bark. Jusqu'à 1000 Hz, cette largeur de bande est linéaire et est égale à 100 Hz. Au-delà de 1 kHz, la largeur de bande est accrue d'environ 20% à chaque création d'une nouvelle bande, permettant d'obtenir une échelle logarithmique. Les sons en basses fréquences seront ainsi modélisés de manière plus fine. Le Mel est une mesure similaire au Bark et un Bark est toujours égal à cent Mel (cf. chapitre 1, figure 1.4).

Pour calculer les coefficients cepstraux en échelle Mel, le signal acoustique passe à travers un banc de filtres dont les fréquences centrales sont distribuées suivant une échelle Mel. Une transformée de Fourier et un passage aux logarithmes sont alors effectués. Le signal temporel initial est alors représenté par plusieurs canaux recouvrant chacun des plages fréquentielles de différentes importances. La méthode MFCC se caractérise ensuite par une prise en compte de tous ces canaux pour le calcul de tous les coefficients finals. La prise en compte des valeurs des canaux se fait à des degrés divers pour chaque coefficient final par l'intermédiaire d'une pondération par la valeur d'un cosinus. Ainsi, chaque coefficient MFCC correspond à une plage de canaux pondérés positivement qui a été opposée à une plage de canaux pondérés négativement. Ceci permet d'obtenir des coefficients relatifs reflétant la répartition des énergies dans les différentes plages de fréquences.

La méthode des MFCC ne fait appel à aucun mécanisme de mémorisation du signal dans les différents filtres qui constituent le banc. Elle est en cela critiquable car éloignée des méthodes qui se veulent d'inspiration encore plus biologique. Les modèles dits d'audition s'inspirent en effet beaucoup plus des connaissances relatives aux mécanismes de l'oreille interne et se caractérisent très souvent par une mémorisation à décroissance exponentielle du signal, une sorte d'amortissement. Le mécanisme mathématique mis en œuvre permet ainsi de simuler l'appauvrissement progressif des médiateurs chimiques au niveau des synapses des neurones. Ces modèles d'audition sont cependant encore du domaine d'une recherche en pleine évolution car nombre de modèles existent, apportant chacun une modélisation plus ou moins fine et plus ou moins approfondie de l'oreille humaine. Il faut pourtant reconnaître qu'aucun de ces modèles ne semble surclasser ses concurrents de manière franche lorsqu'ils sont comparés comme méthodes de prétraitement d'un système de reconnaissance des formes.

Enfin, d'autres modèles se voient aujourd'hui gratifiés de l'intérêt communautaire. Ces méthodes sont fondées sur une prédiction linéaire du signal traité selon une échelle auditive (PLP, cf. chapitre 1, paragraphe 1.7.2.4) ou, mieux encore, selon une telle prédiction utilisant un spectre perceptif relatif (RASTA-PLP, cf. chapitre 1, paragraphe 1.7.2.5). Cette dernière méthode a, en particulier, prouvé avoir une très bonne résistance vis-à-vis des bruits de convolution tels que ceux qui peuvent être perçus sur une ligne téléphonique. La résistance aux bruits additifs peut, en outre, être améliorée par l'adjonction d'un mécanisme supplémentaire (J-RASTA-PLP), rendant l'usage de telles méthodes très intéressant pour les membres de la communauté de recherche en RAP. Nous ne l'avons cependant pas mise en œuvre.

4.2.5/ Note sur le parallélisme

La décomposition du problème en différentes parties conduit logiquement à la création de réseaux de neurones successifs. Ainsi, la reconnaissance des mots du vocabulaire ne devrait se faire que lorsque l'identification des voyelles aura été faite, cette identification n'ayant lieu que lorsque le signal aura été segmenté. Il faut cependant noter que la littérature fournit de bonnes raisons pour ne mettre en œuvre qu'un seul réseau plutôt que plusieurs. Le système tel que nous venons de le définir verrait peut-être ses capacités de reconnaissance améliorées par la simple fusion en un seul réseau connexionniste des trois réseaux que nous allons mettre en œuvre. L'architecture de notre système passerait donc de l'emploi de m réseaux avec des couches de sortie du type 1 parmi n à un réseau

avec une couche de sortie du type *m* parmi *n*.

Sans aller jusqu'à cette extrémité architecturale, il est tout à fait possible d'envisager de faire travailler les trois réseaux en parallèle de manière à ce que les réponses données par chacun se recoupent ou s'infirment. Cette procédure permettrait de détecter plus facilement d'éventuelles erreurs innées du réseau et de modifier le processus de décision en conséquence.

4.3/ Segmentation du signal

4.3.1/ Architecture du réseau

Comme cela a déjà été vu au paragraphe 4.2.1, la segmentation du signal se fait grâce à l'emploi d'un unique réseau de neurones. La couche d'entrée de ce réseau est constitué de cinq trames successives de coefficients MFCC calculées à intervalles réguliers dans le signal. Chaque trame est calculée sur une plage de signal représentant 32 millisecondes (*ms*) de parole, un intervalle constant entre ces trames étant fixé à l'avance. Le cumul des cinq trames et des quatre intervalles entre les trames constitue la taille de la fenêtre de regard du réseau sur le signal. Les tailles de fenêtre étudiées allaient de 687 *ms* à 2061 *ms* ce qui signifie que les intervalles entre les trames s'étendaient de 132 *ms* à 475 *ms*. Nous ne pouvons donner toutes les tailles de fenêtres étudiées [1] mais la taille ayant permis d'obtenir les meilleurs résultats est la fenêtre de 1680 *ms*, soit un intervalle entre trames de 380 *ms*. Cette valeur est la meilleure obtenue en fonction du besoin d'une connaissance d'un contexte proche. La taille importante de la fenêtre la plus efficace semble montrer la nécessité de mettre en place un contexte phonétique important pour le réseau. La phase de lissage de la courbe de sortie, dont nous parlerons plus loin, pourrait donc être vue comme un processus local par rapport à la phase de segmentation connexionniste qui utilise des informations sur une plage temporelle beaucoup plus étendue, le pas de déplacement pour l'analyse du signal de parole étant de 10 *ms* et le nombre de points utilisés pour le lissage n'excédant pas neuf, comme nous le verrons plus loin. Il est à noter que le pas de 10 *ms* est un pas d'analyse standard dans le domaine de la RAP.

Le lecteur notera que nous avons également décidé de considérer la suite de phonèmes /IH R AO/ comme un seul et même phonème lors du traitement de corpus de chiffres en anglais. Cette suite de phonèmes nous a, en effet, posé beaucoup de problèmes lorsque nous voulions traiter les deux voyelles séparément puisque le phonème /R/ est ici très fortement voisé et même, selon la terminologie phonétique anglaise, parfois roulé.

1. Nous attirons l'attention du lecteur sur le fait que les programmes développés et les résultats obtenus lors des études entreprises dans cette partie de la thèse ont été totalement perdus lors du crash d'un disque dont la sauvegarde et l'archivage n'avaient pas été effectués comme préalablement convenu.

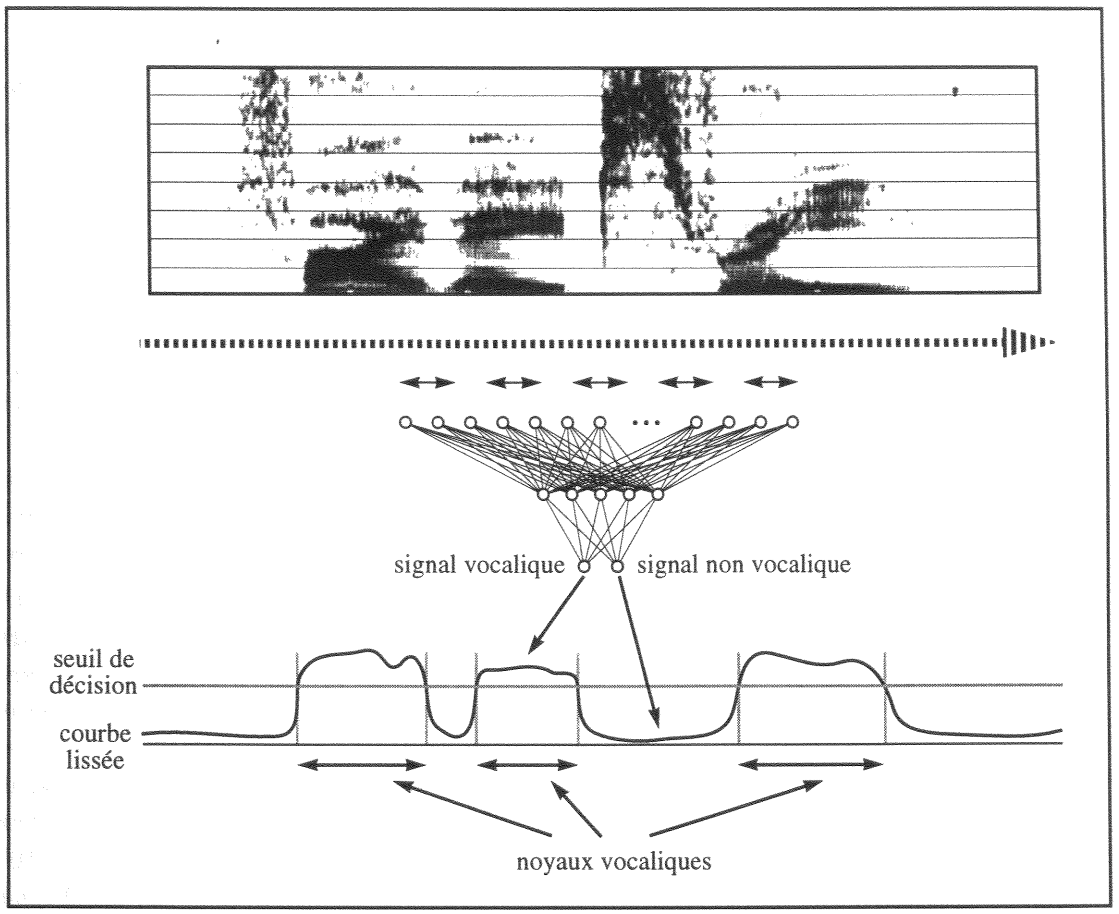


Figure 4.4 : Schéma synoptique de l'étape de détection des noyaux vocaliques.

La sortie du réseau de segmentation est constituée de deux unités. Chaque neurone est affecté à un type d'évènement que nous cherchons à identifier comme, par exemple, le voisement et le non voisement. Ce choix pour deux sorties plutôt qu'une s'explique par le fait que nous pensons qu'il est préférable de marquer les deux types d'événements de manière distincte à l'apprentissage plutôt que d'utiliser une seule sortie. Une raison annexe à celle qui vient d'être donnée est que la valeur des sorties du réseau employé ici s'étend de 0 à +1 et non de -1 à +1. Le choix pour ce dernier type d'intervalle nous aurait peut-être poussé à n'utiliser qu'une seule sortie mais nous aurait imposé d'employer une fonction sigmoïde plutôt que logistique. La fonction employée étant justement logistique, nous avons préféré cette redondance de l'information qui permet d'améliorer les qualités de l'apprentissage et de la représentation interne.

La courbe résultant du processus de segmentation par le réseau de neurones est lissée pour atténuer les phénomènes erratiques qui peuvent parfois être observés du fait de conditions locales particulières dans le signal. Cette étape est similaire à celle dont nous avons déjà parlé au chapitre 3, paragraphe 3.3.2. Plusieurs conditions de lissage ont été étudiées. Nous avons ainsi essayé de mettre en œuvre un lissage par la moyenne et un autre par la médiane sur des fenêtres de 3, 5, 7 ou 9 points successifs de la courbe, ces points résultant d'un processus d'analyse du signal effectué toutes les 10 *ms*. Le lissage par la moyenne correspond à une simple addition de la valeur des points considérés, cette valeur étant ensuite divisée par le nombre de points pris en compte pour obtenir la valeur moyenne sur la fenêtre étudiée. Le lissage par la médiane correspond à un processus plus complexe nécessitant un premier tri des valeurs étudiées. La valeur médiane du tableau trié est alors sélectionné comme résultat moyen de l'ensemble des valeurs observées. Ce dernier type de calcul de "moyenne" permet normalement d'obtenir de meilleurs résultats puisqu'il permet d'éliminer les valeurs anormales trop fortes ou trop faibles tout en permettant de retenir une valeur probablement dans la moyenne. Le lissage par la moyenne semble lui moins bon puisque les valeurs erratiques

seront prises en compte dans le calcul de la valeur résultante. Après différents essais, nous avons choisi d'utiliser la méthode du lissage par la moyenne qui permet d'obtenir de meilleurs résultats malgré la présentation négative que nous venons d'en faire. La cause principale de la qualité de cette méthode est la restriction de la sortie du réseau sur un petit intervalle borné qui permet de limiter l'influence des valeurs anormales.

La courbe lissée permet d'obtenir des segments par un processus supplémentaire de décision. Un seuil est préalablement déterminé, avant toute étude de la courbe lissée, et tout dépassement de ce seuil permet de marquer soit un début soit une fin de segment en fonction de la dérivée de la courbe au point de passage par le seuil. Tous ces segments sont ensuite étudiés de manière à éliminer les segments de trop petite taille c'est à dire dont la durée est inférieure à 24 *ms*, mesure définie expérimentalement cette fois encore.

4.3.2/ Type d'apprentissage

Un système de reconnaissance de la parole doit, pour être considéré comme robuste, pouvoir fonctionner correctement, sans une trop forte dégradation de ses performances de reconnaissance, dans des conditions environnementales variées. Il est cependant difficile de connaître a priori les conditions que va rencontrer le système lors de son utilisation. Il est donc souhaitable d'effectuer un apprentissage qui prenne en compte différentes conditions de bruit même s'il ne les prend pas toutes en compte. Ces conditions devraient également être choisies avec soin puisque certains bruits peuvent paraître plus simples à maîtriser que d'autres. Mais le "bruit" est par lui-même un terme assez vague et il n'en existe pas de définition qui soit scientifiquement acceptable.

Les résultats présentés dans cette partie, concernant la segmentation, ainsi que ceux présentés dans la partie suivante, concernant l'identification des voyelles, permettront aux lecteurs d'appréhender les différentes conditions que nous avons mises en œuvre pour l'établissement de notre système. Du fait de l'ambition de celui-ci, nous avons immédiatement décidé de ne pas tester en milieu bruité les capacités d'un système dont l'apprentissage aurait été effectué sur de la parole propre. Des études ont en effet montré les piètres qualités de tels réalisations [gong95]. Nous avons donc effectué des apprentissages selon différentes conditions. La plus simple correspond, par exemple, à la prise en compte d'un type de bruit à un rapport signal sur bruit, RSSB. Il est possible de généraliser cette condition d'apprentissage à la prise en compte d'un bruit à plusieurs RSSB. La dernière condition que nous verrons, la plus lourde que nous ayons tenté de mettre en œuvre, est la prise en compte de plusieurs bruits à plusieurs RSSB. Cette dernière condition permet d'obtenir des résultats assez variés et sera présentée dans le paragraphe traitant de la reconnaissance des voyelles.

4.3.3/ Résultats à l'échelle des trames

La répartition en classes qui nous sert à présenter les résultats de segmentation plus avant dans cette thèse est double. Une première répartition concerne la classification par trames, ou vectorielle, directement en sortie du réseau connexionniste. Dans ce premier cas, le résultat est réparti en deux classes : soit la sortie du réseau de neurones correspond à la cible fournie par l'étiquetage manuel du corpus d'apprentissage ou de test, soit la sortie du réseau ne correspond pas à cette cible. Le résultat est donc jugé correct en cas de concordance entre la cible et le résultat et incorrect autrement.

Nous ne présenterons ici aucun chiffre car les résultats vectoriels de segmentation ne sont d'aucune utilité au regard de la deuxième étape de la méthode développée ici puisque cette deuxième étape repose sur une utilisation de segments explicites.

Cependant, les différentes tentatives de segmentation du signal nous ont poussé à affiner toujours plus nos classes de sortie, dans l'espoir d'avoir des résultats de toujours meilleure qualité. Ces tentatives nous ont donc poussé à rapprocher notre étape de segmentation de notre étape de reconnaissance des voyelles, la segmentation se faisant en considérant des événements phonétiques de plus en plus élémentaires et de plus en plus spécifiques. Les classes comportent ainsi de moins en

moins d'éléments, ce processus d'affinage des classes s'arrêtant lorsque chaque classe correspond à une voyelle du vocabulaire. Ce processus d'affinage a été appliqué mais la nature du réseau mis en œuvre nous pousse à présenter les résultats dans la partie de ce chapitre traitant de la reconnaissance des voyelles et non de la segmentation.

4.3.4/ Résultats segmentaux

4.3.4.1/ Nature des résultats

Les résultats obtenus en sortie du réseau connexionniste peuvent être traités de manière à obtenir des unités de plus haut niveau que sont, par exemple, les segments vocaliques. Un segment vocalique, ou noyau vocalique, correspond à une période de voisement continue du signal observable à la sortie du réseau. Cette liste de segments, une fois obtenue, peut être mise en correspondance avec la liste des étiquettes manuelles de même nature.

Cette mise en correspondance nécessite de définir trois types d'événements qui produisent un total de cinq classes différentes, ces classes étant présentées de la figure 4.5 à la figure 4.9. Le premier type d'événements, idéal, est la mise en correspondance d'un noyau de l'étiquetage manuel avec un noyau de la segmentation automatique. Dans ce cas, le résultat obtenu en sortie du réseau de neurones est jugé correct (figure 4.5). La détection des segments corrects se fait par l'intermédiaire d'une procédure de comparaison des bornes de début et de fin des noyaux qui est assez libre puisque l'égalité parfaite entre débuts ou entre fins n'est pas imposée. Ainsi, tout chevauchement entre une étiquette manuelle et un segment connexionniste permettra de juger le segment comme étant correct, à quelques nuances près dont nous reparlerons.

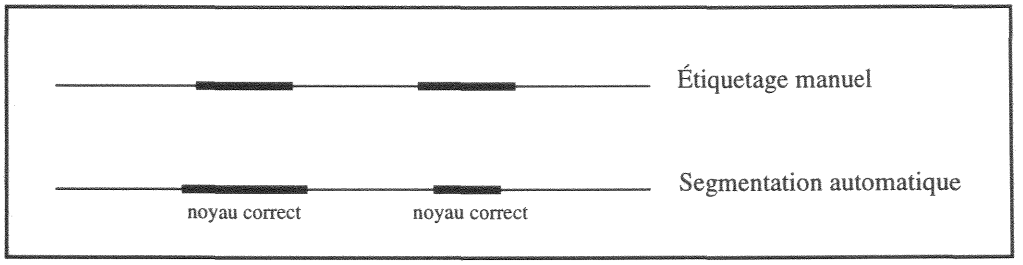


Figure 4.5 : Noyaux de segmentation automatique classés corrects.

Un deuxième type d'évènements pouvant apparaître est la présence, dans l'étiquetage manuel ou dans la segmentation automatique, de noyaux ne pouvant être mis en correspondance avec aucun noyau de la liste en vis-à-vis. Ces noyaux ne peuvent être mis en correspondance du fait de l'exclusion de leurs bornes par rapport à tout noyau de la liste en vis-à-vis.

Pour ce deuxième type d'événements, deux cas peuvent se produire. Il est ainsi possible que l'un des noyaux issus de la segmentation automatique ne puisse être mis en correspondance avec un segment de l'étiquetage manuel. Dans ce cas, il faut considérer ce noyau détecté automatiquement comme invalide. Il est alors question d'insertion (figure 4.6).

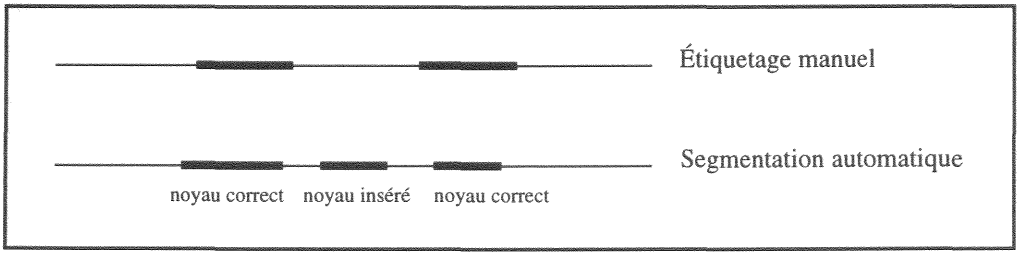


Figure 4.6 : Noyaux de segmentation automatique classés en insertion.

Le cas dual est le cas où un noyau de l'étiquetage manuel n'a pas été confirmé par la procédure de segmentation automatique. Ce noyau n'apparaît donc pas dans la liste des segments détectés par la procédure automatique alors que l'expert chargé de la segmentation manuelle du corpus l'avait

indiqué. Le noyau manuel est alors considéré en élision, c'est à dire supprimé (figure 4.7).

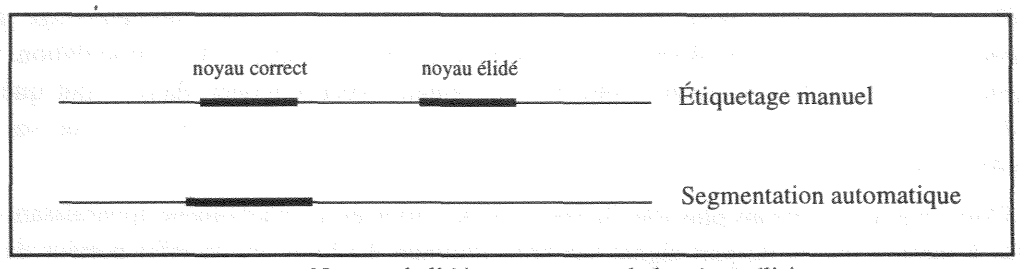


Figure 4.7 : Noyaux de l'étiquetage manuel classés en élision.

Un dernier type d'événements qui peut se produire lors de la phase de segmentation concerne la limite de la fonction approximée qui est implantée dans le réseau connexionniste. Il se peut en effet que certaines parties du signal appartiennent à un espace de paramètres considéré par le réseau de neurones comme étant le seuil de passage d'une catégorie à une autre. Dans ce cas, le réseau peut rapidement modifier sa réponse et juger qu'il est, alternativement, en présence d'une catégorie puis d'une autre. Une explication complémentaire possible de ce comportement est que le réseau n'a aucune mémoire de sa réponse lors des pas de temps précédents. Le réseau est en effet statique et donne une réponse en fonction des seuls paramètres d'entrée. Il lui est alors impossible de déterminer par apprentissage les durées des événements qu'il a à classer tout autant qu'il lui est impossible de moduler son jugement en fonction de ces durées. Ce comportement nous a d'ailleurs poussé à mettre en place une procédure de lissage qui apporte un peu de cohérence au comportement du réseau à un niveau local. Mais la procédure de lissage est parfois insuffisante au sens d'une cohérence plus globale et des pics répétés d'assez longue durée peuvent apparaître. Ces pics auront une durée supérieure à 24 *ms*, durée que nous avons utilisée comme étant la durée minimale d'un pic et qui nous a précédemment permis d'éliminer tous les pics de durée inférieure.

Là encore, comme dans les cas d'insertion et d'élision, deux cas peuvent se produire. Le réseau peut ainsi avoir du mal à décider à quelle catégorie appartient un extrait du signal qui peut alors être considéré, a posteriori, comme appartenant à l'espace délimitant la frontière entre les deux catégories. Le résultat observable dans ce cas est une sur-segmentation, un noyau de l'étiquetage manuel étant détecté sous la forme de plusieurs noyaux d'assez petites tailles par la procédure de segmentation automatique. Dans ce cas, le premier segment issu de la procédure automatique est considéré comme correct alors que les noyaux suivant sont considérés comme résultant d'une division du segment manuel (figure 4.8).

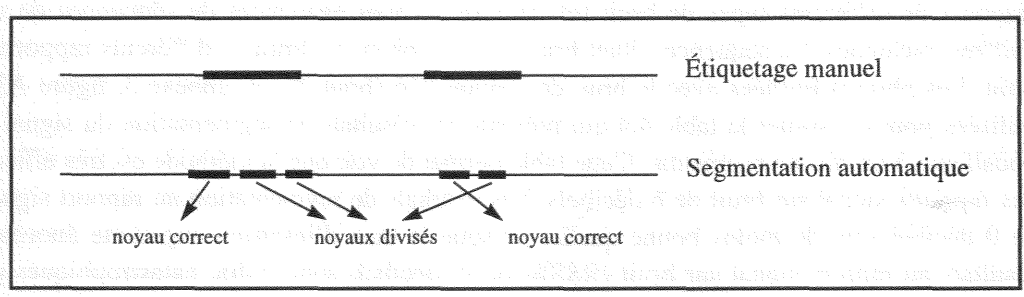


Figure 4.8 : Noyaux de segmentation automatique classés en division.

Le cas dual de la sur-segmentation est la sous-segmentation. Dans ce cas, alors que l'expert a isolé dans un passage deux noyaux successifs distincts, la procédure de segmentation automatique ne réussit à isoler qu'un seul noyau. Deux explications sont envisageables vis-à-vis de la sous-segmentation. La répartition des phonèmes, avant apprentissage, en différentes classes plus ou moins larges peut conduire à l'observation de tels résultats dans le cas où des signaux relativement similaires ont été répartis dans des classes opposées. Ainsi, les voyelles et les semi-consonnes, bien que n'appartenant pas à la même classe phonétique, sont assez similaires dans leurs formes

spectrales. Une répartition de ces deux classes phonétiques dans des classes d'apprentissage différentes et opposées peut, dans une certaine mesure, invalider l'apprentissage du réseau connexionniste et rendre ce dernier très sensible à des nuances dues à la coarticulation. Une autre explication de ce phénomène peut être trouvé, comme précédemment, dans le fait que le réseau utilisé ici pour la segmentation est statique et qu'il n'a donc aucune mémoire de ses décisions antérieures.

Comme nous le verrons plus loin, la sous-segmentation est un phénomène apparaissant de plus en plus à mesure que le rapport signal sur bruit diminue. Le bruit est en effet porteur de sa propre énergie et celle-ci semble accentuer le problème de la sous-segmentation, et donc de la fusion des étiquettes manuelles, à mesure que le bruit s'intensifie.

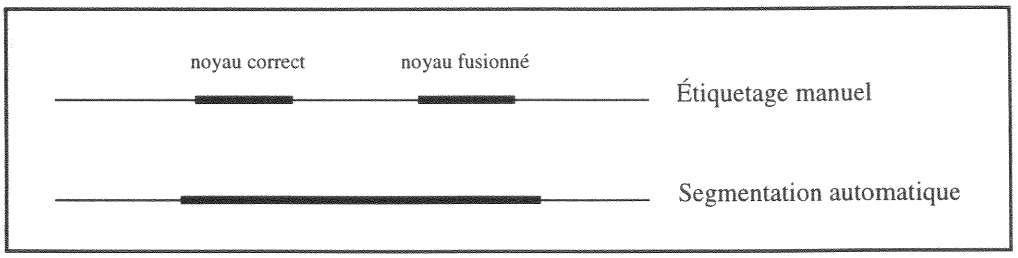


Figure 4.9 : Noyaux de l'étiquetage manuel classés en fusion.

Nous voulons attirer l'attention du lecteur sur les types de pourcentages générés par les cinq classes différentes de résultats que nous venons de voir. Nous traitons à travers ces classes tant les erreurs relatives à la segmentation automatique que celles en rapport avec l'étiquetage manuel. Les pourcentages sont donc relatifs tant aux erreurs de la segmentation automatique vis-à-vis de la segmentation manuelle que de la segmentation manuelle vis-à-vis de la segmentation automatique. En conséquence, la somme des pourcentages dans chaque ligne peut dépasser 100%, la comptabilisation se faisant par rapport au nombre de noyaux détectés automatiquement ou par rapport au nombre de segments manuels.

Les différents types de classes de résultats étant donnés et leurs caractéristiques étant présentées, nous allons maintenant exposer les résultats que nous avons obtenus avec cette architecture.

4.3.4.2/ Résultats segmentaux

Les premiers résultats, que nous présentons à la table 4.4, sont issus de [buniet93]. Cette table présente les résultats obtenus sur le corpus NOISEX [varga92]. Le corpus NOISEX permet de disposer de différents types de bruit (cf. annexe 3) mais également de séquences de un ou trois chiffres enchaînés, les séquences étant bruitées avec plusieurs bruits à différents rapports signal sur bruit. Les phrases bruitées avec le bruit de l'avion F16 (bruit 20, cf. annexe 3, figure A3.3) ont été utilisées pour constituer la table 4.4 qui présente les résultats de segmentation du signal en noyaux vocaliques à un niveau segmental. Cette table permet de voir que la méthode est très efficace jusqu'à des rapports signal sur bruit de 6 décibels. Les résultats de segmentation au rapport signal sur bruit de 0 décibel sont de moins bonne qualité puisque le taux d'insertion augmente énormément. Les résultats au rapport signal sur bruit (RSSB) de -6 décibels sont, enfin, catastrophiques puisque les taux d'insertion et de division augmentent tandis que le taux d'élision, très faible jusqu'alors, dépasse dix pour cent. Ce dernier taux d'erreur est le plus grave puisqu'il correspond à la perte définitive des segments vocaliques, les erreurs d'insertion et de division pouvant être gérées, dans une certaine mesure, par un étape lexicographique. Le faible taux de fusion à tous les RSSB s'explique par le fait que les corpus des chiffres épelés de NOISEX sont prononcés de manière enchaînée. Cette prononciation est certes plus complexe que ne le sont les mots isolés mais ne pose pas autant de problèmes que la parole continue.

Dans la table 4.4, comme dans toutes les tables présentées dans cette thèse, nous avons associé la parole propre, et donc non bruitée, au rapport signal sur bruit de $+\infty$. Cette symbolique pourra être

facilement critiquée pour ce qu'elle implique du point de vue de l'ingénieur et donc du point de vue technique mais peut être facilement justifiée par l'équation 3.1 du chapitre 3 qui formule le calcul du rapport signal sur bruit. En effet, dans cette équation, le calcul du RSSB sur un signal non bruité implique que $s(n) = s'(n)$ et donc que $s(n) - s'(n) = 0$. Le diviseur de l'équation devient donc égal à 0 et le RSSB est infini.

Rapport signal sur bruit utilisé en apprentissage et en test	correct (%)	insertion (%)	fusion (%)	division (%)	élision (%)
+∞	98	0	1	2	0
18 dB	96	0	2	1	0
12 dB	97	0	2	2	2
6 dB	97	2	2	1	0
0 dB	96	11	2	2	2
-6 dB	88	16	1	4	12

Table 4.4 : Détection des noyaux vocaliques obtenus avec le corpus NOISEX (bruit utilisé pour l'apprentissage et le test : F16).

La table 4.5 montre la qualité temporelle des bornes des segments de la phase de segmentation présentée à la table 4.4. Les résultats obtenus sont de bonne qualité et, fait qui était prévisible, cette qualité est fonction du rapport signal sur bruit. Dans cette table ne sont présentés que les écarts entre les segments de l'étiquetage manuel et les noyaux obtenus par le réseau et jugés corrects par la procédure de classement. La prise en compte de tous les noyaux détectés automatiquement aurait, en effet, posée des problèmes insolubles.

Rapport signal sur bruit utilisé en apprentissage et en test	Nombre de noyaux vocaliques	Moyenne		Écart-type	
		début (ms)	fin (ms)	début (ms)	fin (ms)
+∞	147	24	32	23	47
18 dB	145	23	30	21	43
12 dB	146	22	31	21	45
6 dB	146	23	31	23	42
0 dB	145	26	39	28	46
-6 dB	133	39	65	42	73

Table 4.5 : Différences entre segmentation manuelle et segmentation automatique.

Des tests similaires à ceux qui viennent d'être présentés, également publiés dans [buniet93], nous ont permis de juger de la qualité de la méthode de segmentation dans le sous-corpus de parole continue du corpus en langue française BDSON, le corpus NOISEX étant, lui, constitué de chiffres enchaînés en langue anglaise. La nature du corpus permet donc d'étudier la tâche dans un environnement qui pose plus de difficultés. Ces difficultés ne nous ont pas permis d'atteindre des taux de reconnaissance similaires à ceux de la table 4.4. Les résultats font en effet apparaître des taux de fusion et d'insertion beaucoup plus importants que dans le cas du corpus NOISEX, les taux de division et d'élision étant, par contre, plus faibles.

Le fort taux de fusion s'explique principalement par le fait que les mots commencent ou terminent le plus souvent par une voyelle. Il est, de ce fait, difficile de trouver une frontière entre deux voyelles consécutives et le taux de fusion n'en est que plus important.

Les résultats obtenus sur ce corpus de parole continue française laissent présager de l'obtention de résultats similaires sur tout autre corpus de parole continue, langue anglaise comprise. Ce type de résultats nous pose un problème évident de méthode. Les résultats obtenus lors de l'étape de segmentation nécessitent dans ce cas des traitements supplémentaires pour être exploitables par l'étape ultérieure de reconnaissance des voyelles. Il s'agira donc de quantifier la durée des phonèmes

étudiés pour être à même d'effectuer un découpage des noyaux trop longs.

Une autre consiste à abandonner le choix d'une segmentation en grandes classes pour commencer à segmenter le signal en unités phonétiques beaucoup plus précises, par exemple en fonction des voyelles à reconnaître. La première étape de notre système perd alors sa généralité mais gagne en qualité de segmentation dans la mesure où les éléments phonétiques à segmenter ont une définition beaucoup plus précise.

Rapport signal sur bruit utilisé en apprentissage et en test	correct (%)	insertion (%)	fusion (%)	division (%)	élision (%)
+∞	66	10	28	0	2
18 dB	66	10	28	0	2
12 dB	64	9	30	0	2
6 dB	63	8	32	0	2
0 dB	59	15	38	0	2
-6 dB	61	42	30	0	7

Table 4.6 : Détection des noyaux vocaliques obtenus avec le corpus BDSON (bruit utilisé pour l'apprentissage et le test : F16).

Les segments ayant été obtenus, il faut désormais mettre en place une méthode capable de reconnaître la voyelle prononcée. Comme nous l'avons fait remarquer précédemment, nous ferons appel tant à des réseaux du type du perceptron multicouche dans une sorte d'extension de notre phase de segmentation qu'à des *Selectively Trained Neural Network*, réseaux connexionnistes spécialement dédiés à la tâche d'identification des voyelles.

4.4/ Reconnaissance des voyelles

4.4.1/ Architecture utilisée

4.4.1.1/ Architecture connexionniste

L'architecture utilisée lors de la phase de reconnaissance des voyelles est, elle aussi, fondée sur l'usage de perceptrons multicouches. Deux architectures connexionnistes ont été utilisées : la première utilise les STNN, *Selectively Trained Neural Network* alors que la seconde ne constitue qu'une extension de notre précédente méthode de segmentation puisque la seule différence réside dans la définition sémantique des sorties, qui ne correspondent plus à de grandes classes phonétiques mais à des phonèmes. La méthode des STNN a été choisie car des travaux effectués au sein du laboratoire [anglade92a], [anglade93] ont montré la capacité de ces réseaux à résoudre des tâches de reconnaissance de lettres prononcées de manière isolée en milieu bruité, en particulier pour de la parole Lombard où les phonèmes sont déformés par l'effort vocal du locuteur essayant de porter sa voix à un niveau énergétique équivalent à celui du bruit ambiant [lombard11].

Les STNN, plus encore que les perceptrons multicouches, sont des architectures nous permettant de mettre en œuvre le paradigme des méthodes analytiques, en opposition au paradigme des méthodes globales, dont nous avons déjà parlé au paragraphe 4.1.3.2. La figure 4.10 permet de comprendre la différence entre ces deux grands concepts en présentant, à gauche, une application des modèles de Markov à une tâche de reconnaissance d'un mot dans son ensemble, en accord avec le paradigme des méthodes globales, tandis que la figure de droite présente la mise en œuvre d'un STNN à une tâche de reconnaissance d'un mot, l'information utilisée étant plus limitée et correspondant donc au paradigme des méthodes analytiques. Le schéma d'utilisation pourrait d'ailleurs être tout aussi bien utilisé pour une tâche de classification de mots que pour une tâche d'identification de voyelles, avec des résultats cependant variables.

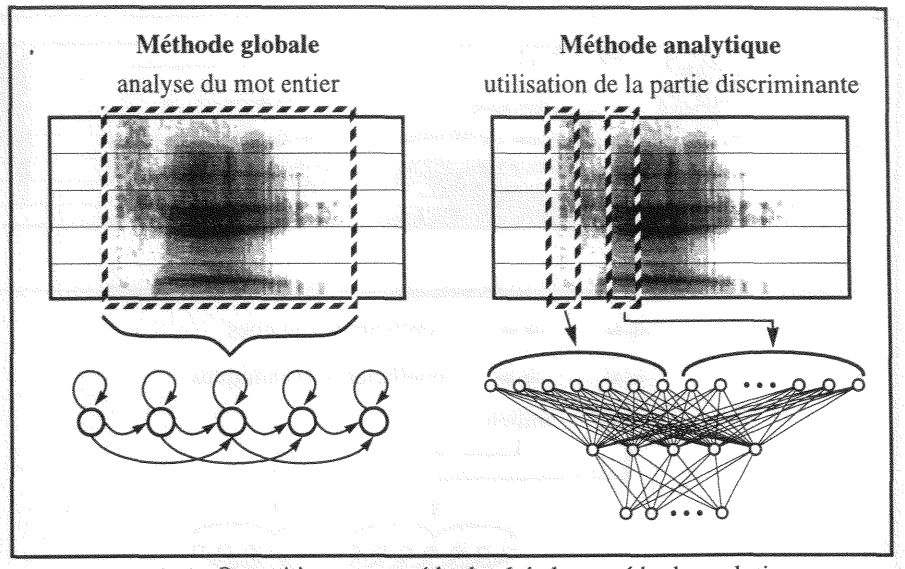


Figure 4.10 : Opposition entre méthode globale et méthode analytique.
Présentation du principe de fonctionnement des STNN.

Le réseau utilisé pour la classification des voyelles peut également être un perceptron, c'est à dire un réseau équivalent à celui utilisé pour la segmentation que nous avons présenté au paragraphe 4.3. Ce réseau de reconnaissance des voyelles se voit fournir les mêmes données que celles qui sont fournies au réseau de segmentation c'est à dire 5 trames de coefficients de cepstres Mel statiques, les trames étant espacées par un intervalle constant et non nul. La différence entre ce réseau et un réseau de segmentation réside dans la définition des sorties. Ces sorties ne sont plus les grandes classes que nous avons vu au paragraphe 4.3 mais correspondent à toutes les voyelles du vocabulaire. Une sortie supplémentaire est mise en place pour traiter les insertions et permet de classer les phonèmes "non voyelle" présents dans la liste des segments et ne correspondant pas à l'une des voyelles du corpus.

Ce détournement du réseau de segmentation est en fait une des réponses possibles au problème de concaténation des segments de voisement qui peut être posé lorsque la dichotomie des phonèmes en grandes classes sépare des phénomènes acoustiquement proches.

Lorsque l'étape d'identification des voyelles repose sur l'utilisation des STNN, l'architecture est celle présentée à la figure 4.11. Les données de base de cette méthode sont les segments isolés dans le signal lors de la phase de segmentation. L'espace des paramètres contenu dans ces différents segments est réduit par l'emploi de la méthode de prétraitement des cepstres Mel. Cette réduction permet d'obtenir deux paires de trames de coefficients qui sont fournis en entrée du réseau STNN et permettent d'effectuer la reconnaissance de la voyelle à partir du début et du milieu du segment. Ces deux paires de trames sont chacune constituées de douze coefficients statiques et de douze coefficients dynamiques. La différence entre ces deux types de coefficients sera expliquée dans le paragraphe 4.4.1.2 présentant de manière approfondie la méthode de prétraitement. Le nombre des sorties du réseau est déterminé par le nombre de voyelles présentes dans le vocabulaire étudié.

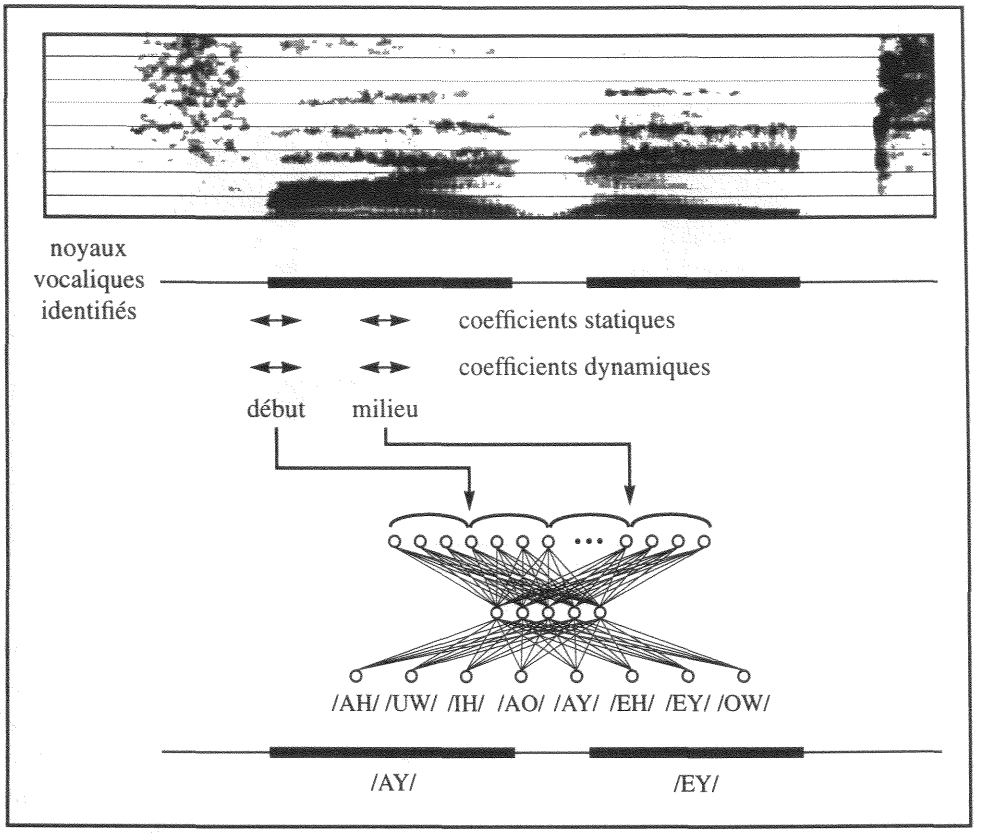


Figure 4.11 : Schéma synoptique de l'étape d'identification des voyelles.

Le positionnement des trames en milieu de segment (cf. figure 4.11) ne pose pas de problème puisque l'indice temporel est tout naturellement déterminé à partir des bornes de début et de fin du segment. Par contre, le positionnement en début de segment ne peut pas se faire à partir des indices temporels. Ceux-ci sont en effet trop variables et dépendent des caractéristiques du locuteur. La meilleure méthode consiste à traiter le signal relativement à l'énergie et non plus par rapport au temps. La méthode employée dans cette thèse est présentée à la figure 4.12.

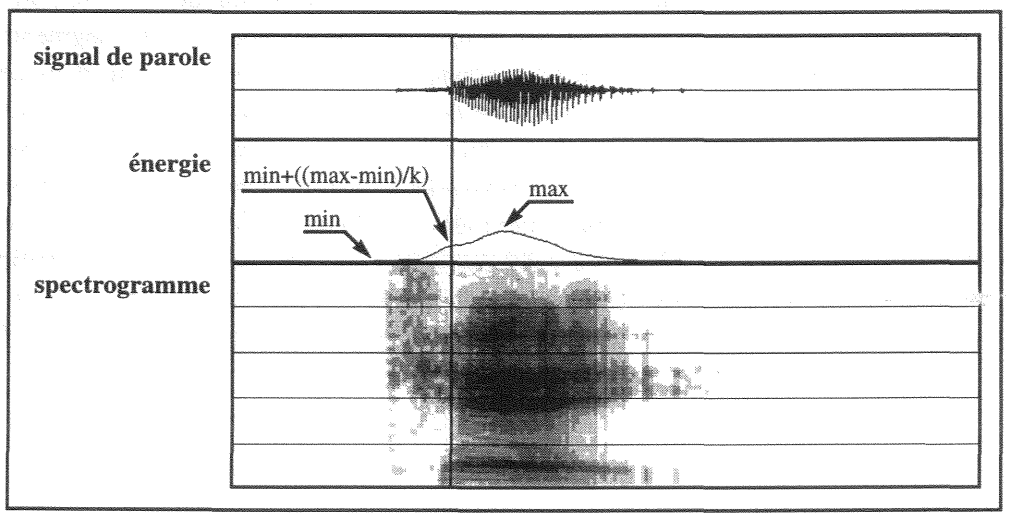


Figure 4.12 : Positionnement en fonction de l'énergie du signal des trames de coefficients d'un STNN.

Deux indices d'énergie sont calculés par rapport au segment à analyser : une première valeur d'énergie minimale permet de connaître le niveau de l'énergie du bruit présent dans l'environnement sonore alors qu'une deuxième valeur d'énergie maximale permet de connaître la valeur maximale

atteinte pendant la prononciation du phonème considéré. Ces deux valeurs permettent de calculer une différence entre l'énergie minimale et l'énergie maximale qui peut servir de mesure de la puissance vocale développée pour la prononciation du phonème. Cette différence permet de calculer, grâce à un coefficient noté *k* dans la figure 4.12, une valeur relative d'énergie dans la voyelle. Cette valeur relative permet de déterminer un indice temporel dans la première partie du segment, entre la borne de début et l'indice temporel du maximum. Cet indice temporel calculé marque une position d'énergie relative qui est une position indépendante du rythme d'élocution et de l'énergie absolue. De plus, le coefficient *k* peut être modifié d'un réseau à l'autre en fonction de la tâche à résoudre et des connaissances phonétiques détenues a priori.

4.4.1.2/ Prétraitement du signal

Les coefficients utilisés pour la reconnaissance des voyelles avec les STNN sont de deux types. Comme pour l'étape de segmentation (paragraphe 4.3), nous avons utilisé des coefficients cepstraux à l'échelle Mel, les MFCC (cf. paragraphe 4.2.4). Ces coefficients sont ceux directement issus de la méthode de calcul des cepstres Mel et peuvent être qualifiés de statiques puisqu'ils permettent de visualiser l'énergie et la fonction de transfert générée par le conduit à un instant précis, la précision de cet instant étant fonction de la taille de la fenêtre mobile utilisée pour le calcul de la transformée de Fourier.

D'autres coefficients peuvent être déterminés à partir de ces coefficients statiques selon la méthode des différences [furui81]. Les coefficients obtenus dans ce cas sont qualifiés de dynamiques. Les coefficients dynamiques d'ordre 1 sont obtenus par soustraction de la valeur des coefficients statiques de l'instant *t-1* aux coefficients statiques de l'instant *t* comme le montre l'équation 4.1. Il est possible d'itérer ces calculs pour l'obtention de coefficients d'ordre *n*. Les coefficients dynamiques d'ordre 1 permettent de déterminer les trajectoires temporelles suivies par les différentes valeurs du vecteur de prétraitement calculé à partir d'un signal de parole puisqu'ils correspondent à un calcul numérique de la dérivée première.

$$dyn_1(MFCC(i,t)) = MFCC(i,t) - MFCC(i,t-1) \quad \text{avec } MFCC(i,0) = 0 \qquad \text{(Éq. 4.1)}$$

Les coefficients dynamiques font partie de la catégorie des indices robustes pour le traitement de la parole en milieu bruité [hermansky95b]. Ils n'ont pas la fiabilité des indices robustes dont nous avons déjà parlé au chapitre 1, paragraphe 1.7.4 mais permettent cependant d'apporter une certaine robustesse au bruit ambiant. Les coefficients dynamiques d'ordre 1 permettent d'obtenir une résistance aux distorsions de convolution dues aux caractéristiques de l'environnement de communication. Toute variation lente de ces distorsions peut également être supprimée par cette méthode de calcul.

4.4.2/ Apprentissage

Deux systèmes différents ont été utilisés pour la tâche de reconnaissance des voyelles. Le premier système utilise le perceptron multicouche que nous avons déjà présenté au paragraphe relatif à la segmentation. L'apprentissage ne possède aucune caractéristique particulière et utilise la méthode standard de rétropropagation du gradient d'erreur.

Les *Selectively Trained Neural Network*, STNN, possèdent, eux, quelques particularités d'apprentissage. L'utilisation de ces réseaux se fonde explicitement sur la notion de segment et d'énergie présente dans le signal. Ces informations sont nécessaires et constituent des informations supplémentaires aux informations issues de la phase de prétraitement du signal, les coefficients cepstraux d'échelle Mel statiques et dynamiques de premier ordre. Le processus d'apprentissage doit donc être totalement équivalent au processus qui sera mis en œuvre à l'utilisation. Il faudra donc déterminer les valeurs minimale et maximale de l'énergie dans le segment et déterminer, par l'intermédiaire du coefficient *k*, la position d'où seront extraites les trames de début du segment. Le mode de calcul de cette position est donc le même que celui présenté au paragraphe 4.4.1.1.

4.4.3/ Résultats obtenus

4.4.3.1/ Nature des résultats

Les résultats que nous présenterons ici sont de deux ordres. Nous présenterons tout d'abord des résultats faisant suite à ceux présentés dans le paragraphe 4.3.4.2. Ces résultats seront donc donnés à un niveau segmental relativement à tous les segments isolés par l'étape de segmentation et ont été obtenus par utilisation de STNN. Nous présenterons ensuite des résultats de nature vectorielle relatifs à l'identification des voyelles par un perceptron multicouche au niveau de chaque trame.

4.4.3.2/ Apprentissage avec un bruit à un rapport signal sur bruit

Les résultats présentés à la table 4.7 font suite aux résultats de la table 4.4 qui présentait des résultats au niveau segmental. La table 4.7 présente trois pourcentages relatifs à une même condition de bruit, celle de l'avion F16, bruit stationnaire, aux six rapports signal sur bruit disponibles dans le corpus NOISEX. Les trois pourcentages présentés dans la table correspondent à l'analyse de tous les noyaux obtenus lors de la phase de segmentation automatique, que ces segments soient corrects, divisés ou insérés. Le pourcentage des voyelles classées comme correctes correspond aux cas où la voyelle isolée automatiquement est identifiée en accord avec l'étiquetage manuel qui lui est attaché. Les voyelles considérées comme correctes peuvent aussi bien être issues de noyaux corrects que de noyaux de division. Le pourcentage des substitutions marque le nombre de voyelles qui ont été classées incorrectement vis-à-vis de l'étiquette manuelle à laquelle le segment est rattaché. Enfin, le pourcentage d'insertion correspond au pourcentage de noyaux classés par la phase d'identification des voyelles alors que l'étiquette manuelle correspondante n'est pas une voyelle.

Rapport signal sur bruit utilisé en apprentissage et en test	**correct (%)**	**insertion (%)**	**substitution (%)**
$+\infty$	92	0	8
18 dB	92	0	6
12 dB	90	0	9
6 dB	87	2	12
0 dB	83	3	28
-6 dB	35	8	66

Table 4.7 : Classification des voyelles pour le corpus NOISEX (bruit utilisé pour l'apprentissage et le test : F16)

Comme le montre la table 4.7, les résultats obtenus sont assez satisfaisants jusqu'au rapport signal sur bruit (RSSB) de 12 décibels. Ce résultat semble marquer une relative faiblesse du réseau d'identification des voyelles puisque la phase de segmentation a fourni de bons résultats jusqu'à 6 décibels de RSSB. Les taux de reconnaissance correcte restent cependant acceptables jusqu'au RSSB de 0 décibel, le passage dans la plage des RSSB négatifs étant marqué par un très fort taux de substitution qui devient presque égal au double du taux de classification correcte. Le taux d'insertion peut être traité par la mise en place d'une sortie supplémentaire traitant le cas des segments en insertion, cette sortie étant entraînée à reconnaître les événements "non voyelle". Ce mécanisme a été mis en place dans le réseau qui nous a servi à établir la table que nous allons présenter maintenant.

La table 4.8 nous permet de présenter des résultats d'identification des voyelles à un niveau vectoriel [buniet94]. Cette table a été établie à partir des résultats obtenus par un réseau équivalent à celui utilisé pour la segmentation mais dont la couche de sortie a été modifiée suivant la manière qui a été précédemment exposée : le réseau utilisé est donc un perceptron simple et non plus un STNN. À la différence des résultats présentés à la table 4.7, ceux présentés ici ne correspondent qu'au taux

de classification correcte qui correspond normalement à l'analyse de l'ensemble des trames de tous les segments isolés par la phase de segmentation et à l'identification de chacune des trames relativement à l'étiquette manuelle correspondante. Nous avons appliqué notre réseau de classification des voyelles aux segments de l'étiquetage manuel pour découpler l'étude suivante de la phase de segmentation automatique qui la précède normalement. Sont ainsi classées correctes les trames correspondant à une voyelle qui a été correctement identifiée vis-à-vis de la segmentation manuelle. Les trames correspondant à une mauvaise identification sont celles où la voyelle n'a pas été reconnue selon la définition qui en est donnée dans l'étiquetage manuel et où elle a été substituée par une autre.

La table 4.8 nous permet également de nous intéresser pour la première fois au comportement du réseau lorsque les conditions de bruit à l'utilisation ne sont pas les même que celles définies à l'apprentissage. Cette table, du fait du réseau utilisé, permet de tirer des conclusions valables tant au niveau de la segmentation qu'au niveau de l'identification des voyelles. Le corpus de parole utilisé pour le calcul des résultats de cette table est le corpus de parole française BDSON [carre84], corpus de parole continue. Ce corpus de parole, initialement "propre", a été bruité avec le bruit de parole synthétique du corpus des bruits de NOISEX où il est baptisé *speech noise*. Ce bruit est stationnaire comme le montre la figure A3.2 de l'annexe 3. La phase d'apprentissage du réseau a été effectué au rapport signal sur bruit (RSSB) dont la case a été grisée dans la table tandis que les tests ont été effectué aux RSSB correspondant aux cases non grisées.

RSSB utilisé à l'apprentissage	**RSSB +∞ (%)**	**RSSB 18 dB (%)**	**RSSB 12 dB (%)**	**RSSB 6 dB (%)**	**RSSB 0 dB (%)**	**RSSB -6 dB (%)**	**moyenne générale**	**moyenne colonnes 1 à 5**
+∞	100	100	95	72	54	34	75	84
18 dB	98	97	92	79	52	40	76	83
12 dB	96	96	94	86	63	32	77	87
6 dB	72	84	90	96	81	50	78	84
0 dB	72	80	86	89	82	62	78	81
-6 dB	52	55	59	60	66	55	57	58

Table 4.8 : Classification des voyelles pour le corpus BDSON, bruit de parole synthétique, réseau à 9 sorties (une par voyelle plus une «non voyelle»), un rapport signal sur bruit en apprentissage.

Les résultats obtenus sont très intéressants car ils montrent la capacité qu'a la procédure d'apprentissage d'extraire des caractéristiques remarquables dans le signal lorsque celui-ci est plus ou moins bruité. Ainsi, un apprentissage effectué dans des environnements peu ou non bruités, d'un RSSB de +∞ à un RSSB de 12 décibels, permet au réseau de très convenablement généraliser ses connaissances à d'autres conditions. Les résultats vont cependant en se dégradant au fur et à mesure que la condition de bruit en test s'éloigne négativement de la condition utilisée en apprentissage. À l'inverse, une condition de bruit plus favorable que celle utilisée en apprentissage permet d'obtenir de meilleurs résultats.

La constatation de la dégradation des résultats qui vient d'être faite dans le cas de RSSB de plus en plus difficiles peut l'être également dans les cas où le RSSB d'apprentissage est nul ou égal à 6 décibels. Cependant, à l'inverse des observation précédentes, les résultats se dégradent lorsque le RSSB de test s'éloigne du RSSB d'apprentissage tant de manière positive que de manière négative. Ainsi, si les RSSB au voisinage de celui d'apprentissage permettent d'obtenir de bons résultats, il est étonnant de constater qu'un apprentissage fait à 0 décibel permettra de reconnaître de manière presqu'équivalente des voyelles à -6 décibels de RSSB qu'à un RSSB infini. Les rapports signal sur bruit dont nous venons de parler semblent donc marquer un tournant dans le comportement de la phase d'apprentissage du réseau qui devient incapable d'extraire correctement l'information

présente.

Enfin, un apprentissage réalisé avec un RSSB négatif montre que cette technique ne permet pas d'atteindre de bons résultats. En effet, le taux de classification correcte obtenu à la condition d'apprentissage est catastrophiquement mauvais puisqu'à peine meilleur que ce qui aurait été obtenu avec un procédure de décision faisant intervenir un mécanisme de choix au hasard. Cette très mauvaise qualité d'identification peut en fait être expliquée en partie par le choix du bruit qui est un bruit de parole dont l'énergie est ici plus importante que l'énergie du signal à traiter, le rapport signal sur bruit étant négatif. Les capacités de généralisation de l'apprentissage sont également très mauvaises puisque les résultats obtenus à de meilleurs rapports signal sur bruit ne sont guère meilleurs.

4.4.3.3/ Apprentissage avec un bruit à plusieurs rapports signal sur bruit

Des études supplémentaires ont été menées à partir de la base ayant servi au calcul des résultats de la table 4.8. Nous avons cette fois décidé d'effectuer l'apprentissage non plus à un rapport signal sur bruit mais sur de la parole propre, à RSSB infini, et sur de la parole bruitée à un rapport signal sur bruit variable selon les expériences. L'architecture du réseau, le corpus étudié et le bruit utilisé sont identiques à ceux définis au paragraphe 4.4.3.2 précédent.

L'observation des résultats, présentés à la table 4.9, permet de constater une très nette amélioration des capacités d'apprentissage et de généralisation du réseau. Celui-ci est en effet devenu capable, par simple modification des conditions d'apprentissage, de généraliser sa connaissance aux intervalles de bruit compris entre le RSSB variable et le RSSB infini alors qu'il n'a pas pu définir ses paramètres internes à partir de ces conditions de bruit intermédiaires, celles-ci étant absentes du corpus d'apprentissage.

RSSB utilisé à l'apprentissage	**RSSB +∞ (%)**	**RSSB 18 dB (%)**	**RSSB 12 dB (%)**	**RSSB 6 dB (%)**	**RSSB 0 dB (%)**	**RSSB -6 dB (%)**	**moyenne générale**	**moyenne colonnes 1 à 5**
+∞ et 18 dB	100	100	99	96	70	23	81	93
+∞ et 12 dB	99	99	100	96	65	32	81	91
+∞ et 6 dB	99	97	97	93	78	46	85	92
+∞ et 0 dB	99	100	100	99	92	66	92	98
+∞ et -6 dB	97	96	94	94	80	56	86	92

Table 4.9 : Classification des voyelles pour le corpus BDSON, bruit de parole synthétique, réseau à 9 sorties, deux rapports signal sur bruit en apprentissage

Il est ainsi possible de remarquer que la meilleure de toutes les conditions d'apprentissage est celle faisant intervenir de la parole propre et de la parole bruitée à un RSSB nul, cette dernière condition ayant prouvé être assez problématique lorsqu'elle est prise isolément (cf. table 4.8, 5ème ligne de résultats). D'une manière générale, il est possible de constater que le réseau est capable d'interpoler les deux conditions de bruit pour obtenir d'excellents taux de reconnaissance entre les deux conditions présentes à l'apprentissage. Comme lors de l'exposé des résultats de la table 4.8, il est également possible de constater que les résultats se dégradent au fur et à mesure que la condition de test s'éloigne des conditions d'apprentissage mais, cette fois, lorsque la condition de test est en dehors de l'intervalle formé par les deux conditions d'apprentissage.

Une troisième étude nous permet enfin de vérifier le bien fondé supposé d'une augmentation de la taille du corpus d'apprentissage à un ensemble continu de conditions. La table 4.10 présente donc les résultats de reconnaissance des voyelles lorsque le corpus d'apprentissage est progressivement étendu d'un corpus ne comprenant que de la parole non bruitée à un corpus comprenant toutes les conditions de bruits possibles en plus de la parole propre. Pour améliorer la lisibilité de ce tableau,

nous avons recopié la première ligne de la table 4.8 qui devient la première ligne de la table 4.10 et la première ligne de la table 4.9 qui constitue la deuxième ligne de la table 4.10. Les résultats peuvent être considérés comme étant rassurants puisque le taux de reconnaissance pour la partie du corpus qui n'a pas été pris en compte à l'apprentissage s'améliore à chaque extension. La raison qui vient d'être donnée ici doit être considérée relativement aux remarques des paragraphes précédents puisqu'il faudrait, en fait, voir ces améliorations successives comme le résultat du rapprochement entre les conditions de bruit en test et la pire des conditions de bruit présente à l'apprentissage.

RSSB utilisé à l'apprentissage	parole non bruitée (%)	18 dB (%)	12 dB (%)	6 dB (%)	0 dB (%)	-6 dB (%)	moyenne générale	moyenne colonnes 1 à 5
+∞	100	100	95	72	54	34	75	84
+∞ et 18 dB	100	100	99	96	70	23	81	93
+∞, 18 et 12 dB	100	100	99	99	84	54	89	96
+∞ et de 18 à 6 dB	100	100	100	97	88	57	90	97
+∞ et de 18 à 0 dB	100	99	96	93	85	57	88	94
+∞ et de 18 à -6 dB	98	99	99	100	88	66	91	96

Table 4.10 : Classification des voyelles pour le corpus BDSON, bruit de parole synthétique, réseau à 9 sorties, plus de deux rapports signal sur bruit en apprentissage

Les résultats que nous venons de présenter peuvent être retrouvés par ailleurs de manière plus limitée. Ainsi, [unnikrishnan91] présente des résultats concernant la reconnaissance de chiffres enchaînés. Ces résultats montrent clairement que le système utilisé, un réseau connexionniste, voit ses capacités grandement améliorées lorsque le corpus d'apprentissage utilise, en plus du signal de parole non bruité, un signal de parole bruité par un bruit gaussien à onze décibels. D'après nos résultats, nous constatons cependant que la condition d'apprentissage utilisant de la parole non bruitée et de la parole bruitée à douze décibels ne permet d'atteindre que 82% sur l'ensemble des conditions de bruits alors qu'un corpus d'apprentissage constitué de parole non bruitée et de parole bruitée à 0 décibel permet d'obtenir 92% de reconnaissance correcte (cf. table 4.9). Des résultats d'ordre plus général, obtenus par [copelli96], montrent que la résistance au bruit additif dépend de la taille du réseau et d'une bonne connaissance a priori du niveau du bruit tandis que la résistance au bruit convolutionnel, ou multiplicatif, est indépendant de la taille du réseau. La perte de nos résultats ne nous permet cependant d'infirmer ou de confirmer ces résultats. Il est à noter que les réseaux connexionnistes utilisés lors de ces dernières expériences sont des perceptrons particuliers baptisés *Tree Parity Machine* et *Tree Committee Machine* et que la tâche étudiée concerne l'apprentissage de règles.

Les tests que nous venons de réaliser sur le corpus BDSON avec le seul bruit de parole synthétique nous ont poussé à étudier les résultats qui pouvaient être obtenus avec un réseau dont le corpus d'apprentissage ne comprendrait pas un mais plusieurs bruits. Ces études font l'objet du paragraphe suivant.

4.4.3.4/ Apprentissage avec plusieurs bruits à plusieurs rapports signal sur bruit

Les résultats présentés dans la table 4.11 suivent des règles assez simples puisque nous avons choisi de généraliser au maximum les conditions d'apprentissage et de test. Les pourcentages fournis dans cette table correspondent donc au taux d'identification correcte des voyelles à partir des trames de la segmentation manuelle. Ces taux d'identification correspondent à la moyenne des taux d'identification des voyelles à tous les rapports signal sur bruit possibles en présence de bruit c'est à dire pour de la parole du corpus BDSON bruitée à 18, 12, 6, 0 et -6 décibels. Ces taux d'identification peuvent être partiellement rapprochés des chiffres présentés dans les colonnes de

moyenne générale, de la table 4.8 à la table 4.10. Les conditions d'apprentissage sont présentées en ligne et correspondent, à l'exception de la première ligne, à la jonction de deux types de bruits dans le corpus d'apprentissage, chaque bruit étant pris en compte à tous les RSSB comme cela vient d'être précisé. La condition d'apprentissage est rappelée dans chaque ligne par un assombrissement de la case du tableau de manière à améliorer la lisibilité. La première ligne correspond à un apprentissage effectué sur de la parole propre et permet de contrôler les capacités de reconnaissance du réseau sur les différents bruits que nous avons utilisés pour réaliser ces tests. Les différents bruits utilisés sont présentés ci-dessous mais c'est l'utilisation du numéro Noise-Rom-0 qui a prévalu sur l'utilisation du nom pour nous permettre de présenter un tableau succinct. Les bruits utilisés sont :

- le bruit synthétique de parole (*speech noise*), bruit stationnaire, bruit numéro 06 de Noise-Rom-0, figure A3.2 du paragraphe A3.3.1 de l'annexe 3,
- le bruit de l'hélicoptère Lynx sur plate-forme, bruit stationnaire, bruit numéro 12 de Noise-Rom-0, figure A3.4 du paragraphe A3.3.1 de l'annexe 3,
- le bruit de salle d'opérations d'un contre-torpilleur, bruit non stationnaire, bruit numéro 14 de Noise-Rom-0, figure A3.5 du paragraphe A3.3.2 de l'annexe 3,
- le bruit de rafales de mitrailleuse, bruit non stationnaire, bruit numéro 16 de Noise-Rom-0, figure A3.9 du paragraphe A3.3.3 de l'annexe 3,
- le signal de test du bateau STITEL, bruit non stationnaire, bruit numéro 18 de Noise-Rom-0, figure A3.6 du paragraphe A3.3.2 de l'annexe 3,
- le bruit d'une usine de fabrication automobile : bruits de soudures électriques lors de l'assemblage du bas de caisse, bruit non stationnaire, bruit numéro 21 de Noise-Rom-0, figure A3.8 du paragraphe A3.3.3 de l'annexe 3,

Les résultats présentés à la table 4.11 montrent des résultats assez variables selon les conditions d'apprentissage et de test. Une première remarque générale peut être faite pour appréhender les pourcentages les plus faibles de la table : ils marquent généralement un effondrement des taux de reconnaissance, surtout aux rapports signal sur bruit les plus faibles. Ces faibles taux de reconnaissance sont à rapprocher des taux de moyenne générale donnés dans les tables précédentes en se rappelant cependant que le pourcentage de reconnaissance sur la parole propre n'est pas pris en compte.

Les taux de reconnaissance donnés dans la table 4.11 sont très intéressants car ils permettent de voir les comportements variables du réseau en fonction de la difficulté des bruits sélectionnés à l'apprentissage. Ainsi, l'apprentissage de l'identification des voyelles avec les bruits 14 et 16, bruit de salle d'opérations d'un contre-torpilleur et bruit de rafales de mitrailleuse, ne permet pas d'obtenir de meilleurs taux de reconnaissance par ailleurs, le corpus de parole propre excepté. À l'inverse, l'apprentissage effectué avec les bruits 18 et 21, signal de test du bateau STITEL et bruit d'une usine de fabrication automobile, a permis d'obtenir dans tous les autres cas de bruits des résultats au moins aussi bons que ceux obtenus sur les bruits présents dans le corpus d'apprentissage.

Ceci tend à prouver que les conditions choisies lors de l'apprentissage peuvent très fortement influencer les capacités de généralisation du système en phase d'utilisation hors de son milieu de mise au point. Malheureusement, ce choix doit être fait de manière judicieuse car il ne permet pas d'obtenir de bons résultats avec toutes les conditions de bruit de même que le choix pour un rapport signal sur bruit plutôt qu'un autre influait positivement ou négativement sur le résultat comme nous l'avons vu lors de nos précédentes expériences. Il semble cependant, à la vue de la troisième et de la quatrième ligne, que la sélection de bruits stationnaires ne soit pas gage de succès. Les apprentissages effectués avec le bruit synthétique de parole ou le bruit de l'hélicoptère Lynx ne nous ont en effet pas permis d'atteindre les meilleurs résultats de généralisation.

En outre, même s'il semble évident que le choix de bruits non stationnaires permette d'améliorer les capacités de reconnaissance, ce choix n'est pas simple à réaliser puisque plusieurs bruits non

stationnaires ont été utilisés avec des résultats forts différents.

Bruit(s) utilisé(s) en apprentissage	Test clean	Test bruit 06	Test bruit 12	Test bruit 14	Test bruit 16	Test bruit 18	Test bruit 21
Aucun (parole non bruitée)	98	86	87	78	91	81	86
14 et 16	98	66	68	92	94	71	92
06 et 18	99	90	94	74	86	89	79
12 et 18	99	93	95	79	90	91	83
14 et 18	99	88	93	91	87	90	96
16 et 18	98	91	95	79	94	91	82
21 et 18	98	92	95	92	89	90	91

Table 4.11 : Apprentissage multibruit dans le corpus BDSON.

Il nous est impossible de présenter d'autres conditions d'apprentissage que celle présentées dans la table 4.11 car bien que nous en ayons réalisé d'autres avec des résultats parfois moins intéressants, nous les avons tous perdus.

Nous pensons que les résultats présentés dans la table précédente constituent une première partie d'une des réponses à [bourlard96a] qui note au sujet de la sensibilité au bruit et au rythme d'élocution des systèmes actuels de RAP que "Ces problèmes sont généralement abordés en améliorant la caractérisation du signal acoustique ou en adaptant les paramètres des modèles. Malheureusement, il ne semble pas que les méthodes développées jusqu'à présent soient vraiment satisfaisantes, et il n'est pas impossible que le modèle de base doive être modifié de façon significative de façon à pouvoir faire face aux différents types de variabilités qui n'ont pas été observés dans la base d'entraînement". Il nous semble, au regard de nos derniers résultats, que la sélection de bruits d'apprentissage, selon un critère de difficulté ou une caractérisation restant à définir, est une méthode efficace pour permettre à un réseau connexionniste d'extraire une information très pertinente d'un signal de parole qui pourra ensuite être bruité suivant différentes conditions.

4.5/ Reconnaissance des mots

La dernière étape de notre système est présentée à la figure 4.13. Cette étape permet de lever les indécisions qui peuvent encore exister, lorsque la voyelle reconnue à l'étape d'identification des voyelles est utilisée pour l'épellation de plusieurs chiffres ou lettres du corpus étudié.

Cette étape repose, de même que les deux étapes précédentes, sur l'emploi de réseaux connexionnistes statiques. Nous avons choisi d'employer des perceptrons multicouches pour cette étape puisque les STNN permettent de traiter des phénomènes énergétiques alors que certains des phonèmes à reconnaître maintenant pourraient avoir une énergie inférieure à l'énergie minimale du segment vocalique, posant problème à l'étape de calcul de la position des deux premières trames. Il faut en effet désormais analyser les phonèmes de nature consonantique qui entourent une voyelle employée dans plusieurs mots. Cette étape pourrait se faire dans le signal précédant la voyelle ou dans celui la suivant. Le choix à faire doit avant tout reposer sur la capacité de résistance au bruit des phonèmes présents en place antérieure ou postérieure à la voyelle.

La figure 4.13 présente le cas d'une suite de trois chiffres, typiquement extraite de NOISEX, qui ne peuvent pas tous être analysés à partir de la seule voyelle. Ainsi, seul le deuxième chiffre, *eight*, peut être reconnu à partir de la voyelle. Les deux autres voyelles posent problème puisqu'elles sont toutes deux utilisées dans plusieurs chiffres. Il faut donc employer deux réseaux particuliers

supplémentaires pour effectuer la reconnaissance du mot prononcé.

Comme le montre la figure 4.13, nous avons choisi, pour la discrimination des chiffres associés à la voyelle /AY/ tout comme pour la discrimination des chiffres associés à la voyelle /IH/, d'utiliser le phonème antérieur pour discriminer les mots prononcés. Ainsi, dans le cas de /AY/, la discrimination se fonde sur la différence acoustique évidente existant entre le /F/ de *five* et le /N/ de *nine* tandis que la discrimination des chiffres associés à /IH/ repose sur la différence entre le /S/ de *six* et le groupe /TH R/ de *three*. Ce dernier choix peut être critiqué puisqu'il existe une certaine proximité acoustique entre /S/ et /TH R/ dans la prononciation anglaise. La perte d'énergie se produisant sur le /R/ est cependant assez forte pour permettre une discrimination aisée. Il aurait aussi été possible d'étudier la présence ou l'absence du groupe phonétique /K S/ en fin de mot pour effectuer cette même discrimination.

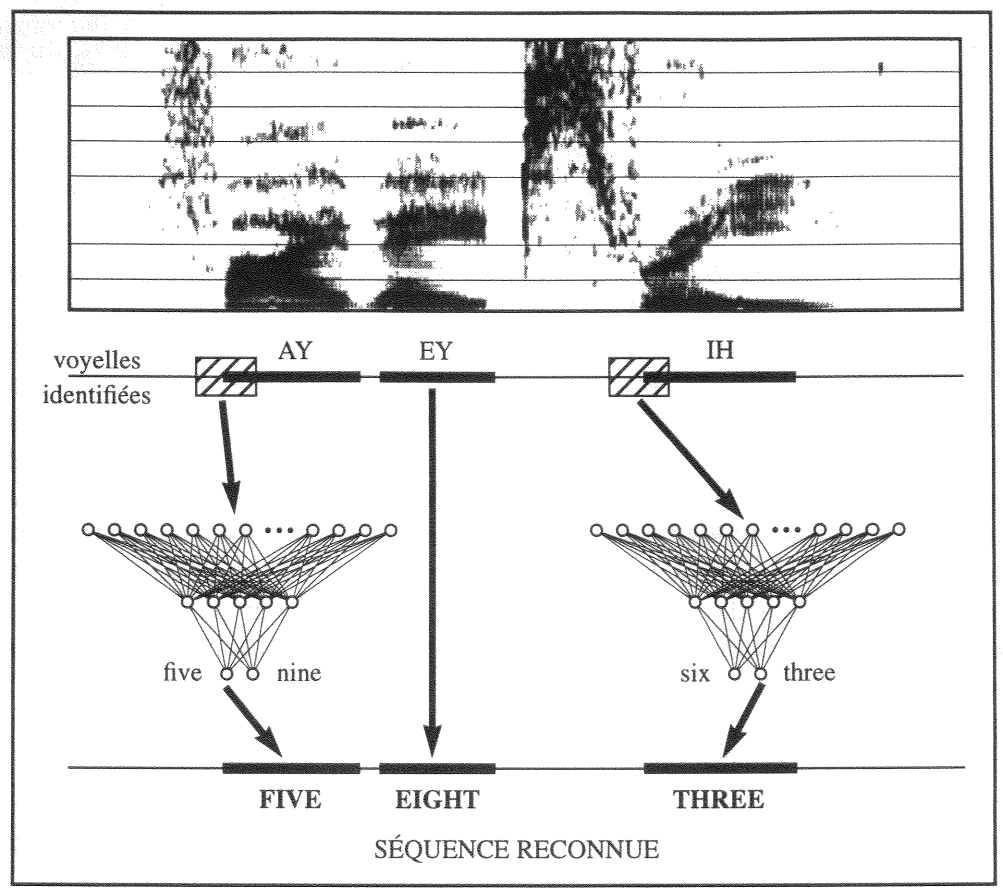


Figure 4.13 : Schéma synoptique de l'étape d'identification des mots.

Les résultats obtenus lors de cette étape d'identification des mots à partir de la voyelle étaient de bonne qualité. La simplicité de la tâche dans le cas des chiffres permet de très facilement atteindre un taux tout à fait acceptable de reconnaissance puisque la discrimination est faite à partir d'indices phonétiques très différents comme dans le cas du /AY/ où les phonèmes à reconnaître appartiennent à des classes aussi différentes que les fricatives et les nasales. Cette bonne qualité des résultats doit cependant être nuancée puisqu'il est évident que des tests menés sur une tâche de reconnaissance de lettres épelées auraient eu à résoudre des problèmes comme ceux du *E-set*. La perte de nos données stockées sur disque ne nous permet cependant pas de présenter le moindre résultat et ne nous permet donc pas de valider nos propos.

4.6/ Critiques des résultats et problèmes posés

Les résultats obtenus aux différentes étapes de traitement sont assez variables. Ainsi, il est clair que la phase d'identification des voyelles, étape principale de notre système, permet d'obtenir de

bons résultats, même à d'assez faibles rapports signal sur bruit. Ces résultats sont cependant variables en fonction des conditions de bruit utilisées à l'apprentissage et laissent apparaître d'autres problèmes.

4.6.1/ Faiblesse des réseaux statiques dans le bruit

Le problème majeur qu'il nous reste à résoudre est le problème posé par l'étape de segmentation. Les noyaux obtenus en phase de segmentation à de grands RSSB sont en effet assez bons, tant du point de vue du faible nombre d'insertions que de la qualité des bornes des segments définis automatiquement et, ce, jusqu'à des RSSB de 6 décibels. Malheureusement, notre étude "in vivo" des sorties du réseau de segmentation nous a permis de constater que les segments obtenus à de très faibles rapports signal sur bruit étaient inexploitables du fait d'un très fort taux de fusion des noyaux, tout particulièrement sur le corpus BDSON de parole continue. Dans ce cas, la proximité des voyelles dans la parole tout autant que les problèmes induits par l'énergie du bruit lui-même provoque l'apparition de segments très plausibles, c'est à dire pour lesquels la sortie du réseau est très élevée quand elle n'est pas maximale.

L'étendue de la fenêtre du réseau de segmentation, très importante comme nous l'avons vu au paragraphe 4.3, est la très probable cause de ce phénomène. Le contexte pris en compte grâce aux trames équidistantes est en effet très vaste. Les meilleurs résultats ont cependant été obtenus avec ce type de fenêtres très larges et il nous semble nécessaire de traiter le problème d'une toute autre manière que celle choisie jusqu'à présent.

4.6.2/ Système fondé sur des heuristiques

Une possibilité envisageable pour résoudre le problème du fort taux de fusion à de faibles rapports signal sur bruit est l'emploi de mesures statistiques effectuées sur la durée du signal. La mise en place d'heuristiques permettant de tronçonner un segment trop important en autant de segments de plus petite taille respectant la durée moyenne des segments cherchés dans un corpus est en effet une solution possible. Ces durées moyennes de segments peuvent être explicitement trouvées dans la littérature [junqua94a] ou extrapolées à partir d'autres recherches [mirghafori95] sur lesquelles nous reviendrons dans le prochain chapitre. Cependant, l'inconvénient majeur de cette méthode est la séparation totale existant entre la connaissance phonétique acquise par les différents réseaux connexionnistes d'une part et la connaissance temporelle issue de mesures statistiques d'autre part. Cette dichotomie ne peut permettre de pondérer les décisions de chacune de ces deux étapes, cette pondération étant nécessaire à une bonne segmentation. Il nous semble donc intéressant de mettre en place une méthode apte à modéliser à un même niveau de représentation les connaissances acoustiques et les connaissances temporelles relatives à la tâche.

CHAPITRE 5 : BRUIT, PAROLE, TEMPS ET MÉMOIRE

"L'espace, dans sa simplicité, peut passer pour quelque chose comme du temps dégradé"

Jean d'Ormesson
Presque rien sur presque tout

"The biggest difference between time and space is that you can't reuse time"

Merrick Furst

Résumé

Nous exposons dans ce chapitre le problème que nous pose le temps et la manière dont celui-ci intervient dans la représentation et le traitement de la parole, du bruit et de la musique. Nous verrons ensuite comment peuvent être appréhendés les phénomènes temporels par l'intermédiaire de systèmes capables de mémoriser et de restituer, à brève échéance, des informations passées.

5.1/ Caractéristiques des phénomènes temporels

5.1.1/ Problème posé

Le problème qui nous est désormais posé est un problème de temps. La méthode de classification des parties vocaliques et non vocaliques du signal développée précédemment devient inopérante lorsque le rapport signal sur bruit est trop faible. Le bruit est bien souvent de nature formantique, tout particulièrement pour les bruits que nous avons étudiés, et ces formants deviennent de plus en plus énergétiques à mesure que le rapport signal sur bruit diminue. Ils finissent donc par être assimilés à de la parole par le perceptron multicouche en charge de la segmentation qui, par conséquent, produit des réponses totalement aberrantes dans le domaine temporel, la tenue du voisement ainsi obtenue devenant incohérente avec la durée moyenne des voyelles. Un phénomène aux résultats similaires se produira, dans le cas d'un signal de parole continue, lorsque la segmentation se fera en grandes classes et donc lorsque le nombre de classes sera trop faible pour qu'il soit possible de distinguer le passage d'un événement phonétique à un autre, ces événements étant fusionnés en une même classe à la sortie du réseau, à l'apprentissage et à l'utilisation.

Une solution possible pour résoudre ce problème est l'abandon de la méthode de prétraitement par *Mel Filter Cepstral Coefficient*, *MFCC*, au profit d'une autre plus robuste au bruit. Le choix des *MFCC* a déjà été justifié au chapitre précédent (cf. chapitre 4, paragraphe 4.2.4) puisqu'elle nous a permis de faire une comparaison des résultats obtenus avec notre architecture à d'autres obtenus en

utilisant les mêmes corpus et la même méthode de prétraitement. D'autres méthodes de prétraitement sont connues pour avoir une meilleure résistance au bruit que celle présentée par les MFCC telles que, par exemple, les méthodes PLP [morgan91] ou RASTA-PLP [morgan92], [hermansky94]. Mais ces deux dernières techniques ne sont pas encore les plus efficaces dans le domaine de la robustesse au bruit (cf. chapitre 1, paragraphe 1.7.4).

Tout un axe de recherches en reconnaissance automatique de la parole essaie de définir une méthode efficace d'amélioration du signal de parole (*speech enhancement*) qui supprime le bruit tout en conservant un signal de parole le plus proche possible d'un signal original non bruité de manière à ce que les systèmes définis en environnement calme puissent être employés sans modification. Ces méthodes d'amélioration du signal sont cependant restreintes à des signaux de bruit stationnaires, ou quasi stationnaires, ce qui limite leur application à des environnements bien ciblés. De plus, le choix d'une telle technique de traitement de signal, offrant une meilleure résistance au bruit, va à l'encontre des contraintes qui nous étaient posées (cf. chapitre 3, paragraphe 3.1). Nous avons donc orienté nos recherches vers une méthode qui soit la plus proche possible de l'architecture présentée au chapitre 4. Nous avons ainsi opté pour le développement d'un système nous permettant de modéliser correctement les informations qui n'étaient pas du tout représentées au sein de notre niveau de segmentation de l'architecture initiale : les durées moyennes des phonèmes.

La durée moyenne peut être déterminée statistiquement sur l'ensemble du corpus d'une tâche. Des mesures ont par exemple été faites sur un corpus d'images radioscopiques [junqua94a]. De telles mesures peuvent tout à fait servir de base à des heuristiques permettant, après obtention des résultats de segmentation par le perceptron, d'isoler différents noyaux vocaliques par simple découpage des noyaux trop longs en noyaux de taille moyenne. Ce découpage est cependant très aléatoire au niveau des segments ainsi constitués et rien ne garantit que cette procédure, algorithmique et aveugle, ne créera pas des entités en complète contradiction avec les règles de la phonétique. Le mécanisme de découpage statistique est, a priori, totalement ignorant de ces règles.

Un autre problème qui se posera lors d'un découpage concerne la prise en compte de l'écart-type en plus de la moyenne. La moyenne permettra de connaître la vitesse moyenne d'élocution des locuteurs du corpus mais ne permettra pas de prendre en compte les variations de cette vitesse. Il faudrait donc, avant découpage d'un noyau a priori trop long, définir la liste de toutes les possibilités de découpage en tenant compte de la moyenne et de l'écart-type. Cette liste de solutions possibles devrait ensuite être analysée par les trois étapes de notre système avant de fournir la liste des différentes réponses possibles. Mais l'analyse de ces réponses et le choix d'une réponse plus probable que les autres doit obligatoirement faire appel à une grammaire des phrases possibles du vocabulaire, mécanisme dont la mise en œuvre n'était pas un de nos objectifs initiaux.

La détermination d'une durée moyenne sur un corpus de taille trop faible peut également être critiquée mais les résultats obtenus dans [junqua94a] sont similaires à d'autres résultats obtenus en phonétique [lonchamp90] et ils n'ont donc pas été influencés, dans le cas de la parole propre, par une élocution particulière telle qu'il serait possible d'en trouver dans des corpus "régionaux". Des mesures faites par ailleurs sur le corpus de phrases anglo-américaines TIMIT, qui est composé de sous-corpus régionaux, ont d'ailleurs permis de montrer la faiblesse d'éventuelles différences dans ce cas particulier [mirghafori95]. Cette analyse d'un corpus de taille imposante a permis d'obtenir des mesures de nature gaussienne, le nombre moyen de phonèmes prononcés par seconde étant de 13,71 avec un écart-type de 1,95 phonèmes (les phonèmes sont ici définis selon le *CMU symbol set* [timitdic91] et la vitesse d'élocution est calculée par division du nombre de phonèmes transcrits observés par la durée qui a été nécessaire à leur articulation ; ceci explique la valeur élevée de la moyenne et de l'écart-type). L'ensemble des mesures montre qu'il n'y a pas de véritable différence entre hommes ($m = 13,83$ et $\sigma = 1,99$) et femmes ($m = 13,43$ et $\sigma = 1,81$). Cet article montre cependant que les locuteurs rapides sont une source d'erreur possible pour les systèmes de RAP, les indices acoustiques étant modifiés en fonction de la vitesse d'élocution [siegler95]. Cette variabilité

de la vitesse d'élocution est par ailleurs très importante dans les cas où la parole est prononcée de manière artificielle (parole articulée) ou lorsque le bruit ambiant est important ([junqua92] et [junqua94a]).

Il semble donc intéressant de se diriger vers un système intégrant des notions aussi bien temporelles que phonétiques. Cette dualité des connaissances semble plus profitable que la mise en place de deux systèmes successifs ignorant chacun les règles de fonctionnement et de décision de l'autre.

Avant de commencer à étudier les systèmes et les modèles nous permettant d'assurer la mise en œuvre conjointe des règles phonétiques et temporelles, nous allons tout d'abord étudier des modèles assez généraux de traitement des connaissances ou de reconnaissance de la parole permettant de modéliser de telles capacités et qui sont inspirés d'études psychologiques ou physiologiques faites sur l'être humain. Mais évoquons tout d'abord quelques lieux communs sur le temps.

5.1.2/ Importance de la notion de temps

Le temps est un phénomène, un fait, voire un milieu, dont l'étude a commencé il y a bien longtemps avec l'avènement de la conscience humaine dont il semble indissociable. Il est également un important sujet de discussion depuis que l'homme maîtrise la langue orale et écrite. Il en existe deux grandes catégories. La première catégorie est le temps concret, ou temps relatif, qui est le temps humain et le seul qui nous intéressera dans cette thèse puisqu'il est en rapport avec la perception. Ce type de temps est composé de trois différents éléments qui appartiennent tous à un présent cognitif : le présent du passé, mémoire d'événements antérieurs, le présent du présent, observation de l'environnement courant, et le présent du futur, attente de réalisations. À un niveau plus philosophique existe le temps abstrait ou temps absolu, le Temps, qui est ici hors de propos. Il semble important de rappeler l'inexactitude partielle des échelles de temps humaines puisqu'Einstein a démontré, dans sa théorie de la relativité, que l'écoulement du temps physique était fonction de la vitesse de l'observateur.

Le temps est une variable incontournable des phénomènes changeant dans l'espace et ... le temps. Ces phénomènes possèdent trois grandes caractéristiques qui permettent de tous les distinguer les uns des autres dans une tâche ou un milieu restreint.

5.1.3/ Trois grandes caractéristiques des phénomènes temporels

La première caractéristique d'un phénomène temporel est sa durée. C'est elle qui nous intéresse le plus en fonction du problème qui nous est posé dans notre tâche de segmentation.

Les autres grandes caractéristiques peuvent être déduites d'études sur les logiques temporelles développées par Allen ([allen83] et [allen84]) et McDermott [mcdermott82]. La deuxième caractéristique est relative aux instants de déclenchement des événements. Cette notion de déclenchement est à prendre au sens large puisque la logique de McDermott, logique temporelle d'instants, représente le début et la fin des événements de manière identique. La logique temporelle d'Allen, logique temporelle d'intervalles, distingue elle les débuts et les fins des événements comme étant les bornes des phénomènes observés et quantifiés.

Une troisième caractéristique concerne la relation des événements comparés les uns aux autres : l'ordonnancement. Tout phénomène peut être classé relativement à un autre. La logique d'Allen développe ainsi treize opérateurs différents pour analyser finement les relations de deux événements. Cette quantité importante d'opérateurs est une des conséquences du traitement et de la modélisation des événements sous forme d'intervalles. Une des treize relations de comparaison entre événements est, par exemple, la synchronisation qui permet de repérer les coïncidences d'apparition des événements. Le grand nombre d'opérateurs provoque également une difficulté de mise en œuvre de la logique d'Allen dans les systèmes automatiques, le problème étant NP-complet.

Quelques notions supplémentaires existent permettant de classer les phénomènes temporels. Il ne

s'agit en fait que de généralisations des concepts que nous venons de voir. Il est ainsi possible de considérer un ensembles d'événements comme constituant une période marquée par un début et une fin ([kowalski86] ou [sadri87]).

Les logiques temporelles sont principalement utilisées dans le cadre de l'intelligence artificielle symbolique et des bases de données. Le lecteur trouvera une bonne étude de ces logiques temporelles et des outils qui leurs sont associés dans [bestougeff89].

Les logiques temporelles présentées ci-dessus sont cependant mal adaptées au traitement de la langue naturelle. Les logiques précédentes reposent en effet sur une définition très précise des instants, comme éléments de représentation atomique pour McDermott ou comme éléments de définition des intervalles pour Allen. Si la définition d'instants précis est déjà critiquable d'un point de vue physique, les relations entre deux instants étant les mêmes que les relations entre deux nombres pris dans l'ensemble des réels, elle l'est encore plus pour la définition de la notion d'instant dans la langue naturelle. Les instants de la langue sont en effet très rarement précis et, dans ce cadre, les logiques temporelles d'Allen et de McDermott deviennent caduques. Pour répondre à ces problèmes, [romary89] propose l'emploi d'une logique temporelle d'intervalles, utilisant deux opérateurs ("prédécesseur" et "inclus") et indépendante de la notion d'instant. Cette logique a montré de bonnes qualités pour les tâches de représentation de la langue.

5.1.4/ Variabilité de la perception du temps dans le bruit

Avant d'exposer les relations existant entre la parole et le temps, nous allons ici brièvement parler de la variabilité de la perception du temps par l'homme dans le bruit. S'il est clair, d'après les études qui ont été faites lors d'autres recherches, que la production de la parole varie en fonction du bruit perçu de l'environnement par le locuteur (cf. paragraphe 5.1.1) et qu'il est clair également que le bruit influe négativement sur l'auditeur, qu'il soit humain ou qu'il s'agisse d'un système de reconnaissance automatique, l'influence du bruit sur la perception du temps chez l'être humain est en revanche moins bien perçue.

Le bruit influence pourtant la perception du temps même si ce fait a été très peu étudié. Les seules études entreprises ne l'ont en effet pas été dans le cadre de la reconnaissance de la parole mais dans le cadre de l'étude de l'influence du bruit dans le monde du travail. Ainsi, les travaux de [jerison55] ont permis de vérifier la grande subjectivité de la mesure du temps par une personne exposée au bruit. La figure 5.1 présente ainsi la perception d'une période de 10 minutes tout au long d'une session de travail de deux heures. La courbe de contrôle a été enregistrée dans un environnement où le bruit ambiant était de 77,5 décibels (ronds blancs) et permet de vérifier la perception de la période considérée dans un environnement calme. La courbe d'étude correspond aux ronds noirs, le bruit dans cette session étant de 111,5 décibels sauf pendant la première demi-heure où il était de 77,5 décibels. Cette deuxième courbe montre clairement que la perception subjective d'une période de 10 minutes est fortement influencée par le bruit. Le temps jugé équivalent à une période de 10 minutes en milieu calme oscille entre 8,5 minutes et 9,5 minutes alors que ce temps jugé équivalent varie, en environnement bruité, entre 7 minutes et 7 minutes et demie. Cette perception est sans doute influencée par la réaction du sujet vis-à-vis de l'agressivité de son environnement sonore qui provoque chez lui une fatigue plus grande, cette fatigue étant l'étalon de mesure du temps subjectif qui permet d'obtenir les mesures en temps objectif de la figure 5.1.

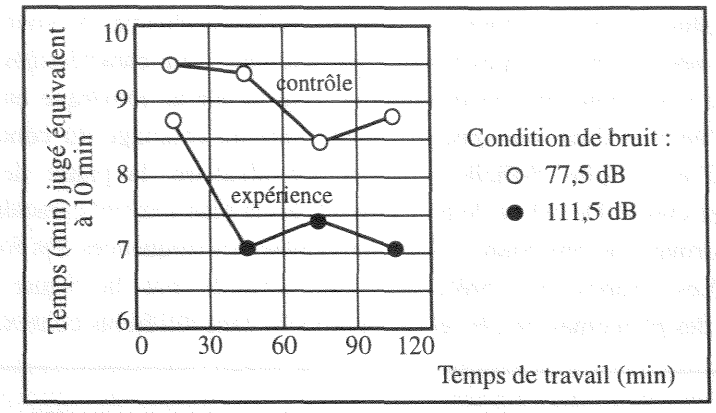


Figure 5.1 : Changement de perception d'une durée de 10 minutes tout au long d'une session de deux heures de travail (d'après [jerison55]).

Des études supplémentaires sur les capacités de travail dans le bruit viennent cependant pondérer les observations précédentes. [jerison57] démontre ainsi que les capacités de travail ne sont pas affectées par le bruit bien que la perception du temps soit elle-même modifiée. Ces dernières études ont été réalisées dans des conditions similaires à celles de [jerison55].

Il ressort des études citées précédemment que le bruit influence la perception du temps tout autant que la perception de la parole. Or la parole est un phénomène temporel comme nous allons le voir dans le paragraphe suivant.

5.2/ La parole comme phénomène temporel

5.2.1/ Regard temporel sur la parole

La parole est un phénomène temporel à part entière. Les trois grandes caractéristiques précédentes peuvent être retrouvées dans la parole. Elles peuvent également être retrouvées dans les procédures de traitement des systèmes dédiés à la reconnaissance et à la classification de la parole.

La parole est composée d'événements sonores qui peuvent être regroupés selon des échelles différentes. Des alphabets phonétiques ont ainsi été définis pour classer des événements phonétiques d'assez courte durée. L'API, Alphabet Phonétique International, ou l'ARPABET, alphabet phonétique défini dans le cadre d'un projet de recherche militaire américain, permettent ainsi de représenter les phénomènes "élémentaires" de la parole grâce à une série de symboles. Les symboles de ces différents dictionnaires sont cependant très proches de l'alphabet de la langue écrite, permettant une lecture, ou une oralisation, directe de la chaîne des symboles phonétiques même avec très peu de connaissances. Ces symboles représentent tous des signaux qui sont clairement définis comme ayant des caractéristiques stables. La modification de la prononciation d'un phonème en contexte est cependant un fait connu en reconnaissance de la parole même si ce fait est parfois difficilement compris par des non spécialistes. Cette contextualisation des phonèmes montre clairement le besoin de connaître la séquences des phonèmes antérieurs, et parfois postérieurs [effet carabine], pour connaître les modifications qui peuvent être engendrées sur un phonème particulier dont on connaît la définition spectrale dans l'absolu. Seul l'ARPABET définit par exemple quelques symboles phonétiques mis en contexte, l'API ayant été adapté à de telles contextualisations. Le problème du contexte est clairement un problème d'enchaînement et donc de temps mais la proximité des définitions des différents alphabets phonétiques avec la langue écrite fait que certains symboles des alphabets sont eux-même constitués de séquences d'événements plus élémentaires encore, rendant la séquence de symboles phonétiques sécable d'un point de vue événementiel.

Les voyelles de la langue orale sont, par définition, des phénomènes stables et elles sont constituées d'une période où les fréquences de voisement sont tenues par le locuteur. Mais d'autres symboles phonétiques se caractérisent par des phénomènes dynamiques plutôt que statiques, rendant

leurs définitions plus complexes. Ainsi les sonantes ou les occlusives ne sont pas des phénomènes sonores véritablement atomiques. Les nasales, par exemple, sont caractérisées par l'observation du phénomène d'œil nasal dans un spectrogramme. Une nasale se remarque en effet par la presque totale absence d'énergie dans son spectrogramme, due au couplage du conduit nasal au conduit vocal, rendant d'autant plus difficile, voire même aléatoire, la phase de reconnaissance des phonèmes de cette classe. De même, le phonème /R/ est le plus souvent caractérisé par une chute des fréquences des formants le précédant et par une montée des fréquences des formants qui le suivent [lonchamp90]. Les propos qui précèdent sont illustrés par la figure 5.2 présentant des spectrogrammes des phonèmes /m/, /n/ et /R/ prononcés dans différents contextes.

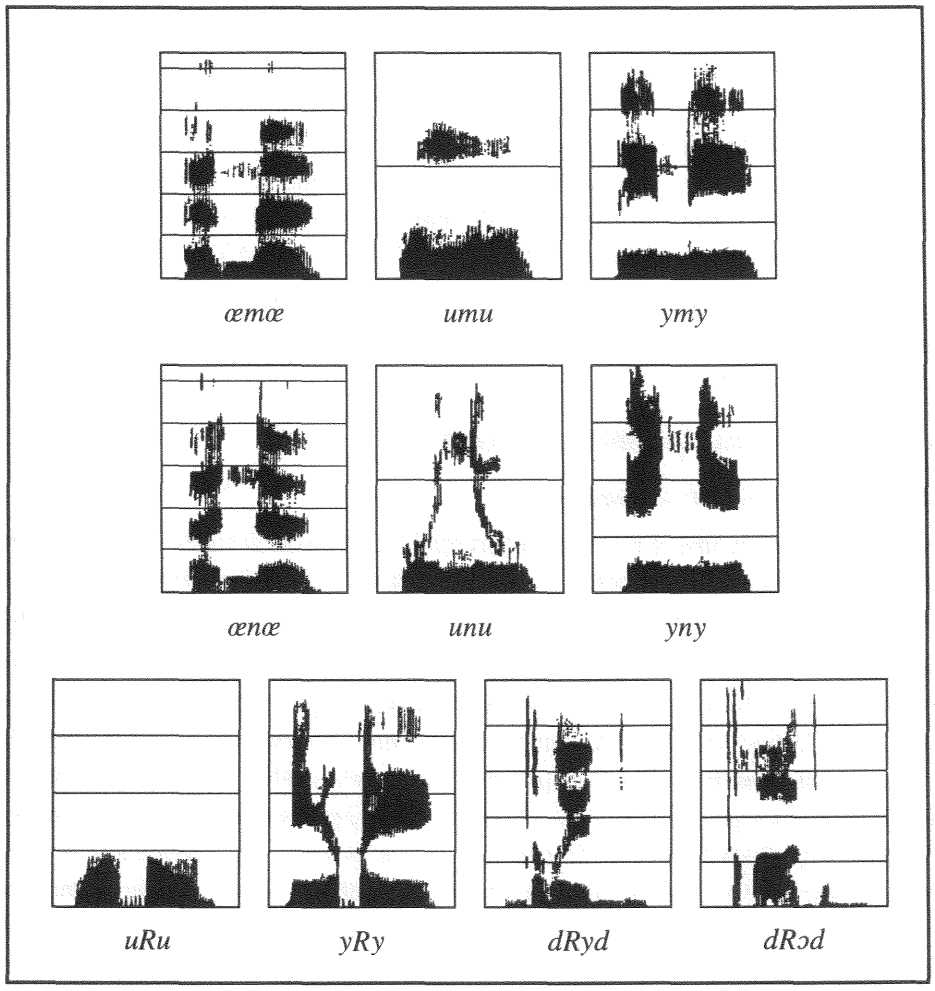


Figure 5.2 : Les phonèmes /m/, /n/ et /R/ prononcés dans différents contextes, cf. chapitre 1, figure 1.1 (d'après [lonchamp90]).

De la même manière, les occlusives, qu'elles soient voisées ou non, ne sont pas des phénomènes stables mais sont composées de deux parties principales : un silence de préparation de l'occlusive et une barre d'explosion, *burst*, marquant le relâchement des lèvres. Un total de quatre ou cinq parties peuvent être observées selon les langues. Ces deux phénomènes principaux, bien distincts, sont normalement regroupés au sein d'une même symbolique marquant l'ensemble du processus. Certains alphabets phonétiques, comme l'alphabet du corpus de parole TIMIT, distinguent cependant ces deux parties. Le phonème /d/ de l'API est ainsi représenté par les symboles /dcl/ et /d/, /dcl/ marquant l'occlusion (*closure*) et /d/ la barre d'explosion. Les symboles d'occlusion, contrairement à ce qu'il est possible de penser, ne sont pas exclusivement associés aux barres d'explosion respectives des occlusives. Ainsi, en anglais, l'occlusion du /dcl/ peut se retrouver dans *joke* (/dcl/ /jh/ /ow/ /kcl/ /k/) et l'occlusion de /tcl/ peut être retrouvée dans *choke* (/tcl/ /ch/ /ow/ /kcl/ /k/).

Le nombre de ces éléments phonétiques est, bien sûr, limité et, ce, pour deux raisons. La première raison est due à la simple restriction des capacités de production de l'appareil phonatoire. Les organes humains de production de la parole (chapitre 1, para 1.3) restreignent le nombre des phonèmes qu'il est possible d'articuler à 70 environ, ce nombre tenant compte de la variabilité intrinsèque de la production de la parole chez l'homme puisque bien plus de sons peuvent en fait être prononcés même si tous ces sons ne peuvent pas toujours être distingués les uns des autres. La deuxième raison de ces limitations est de caractère culturel puisque toutes les langues du monde n'utilisent pas toutes les possibilités de l'appareil phonatoire.

Une relation évidente entre la parole et le temps à l'échelle du phonème est l'ordonnancement d'une séquence. Il est évident qu'une séquence de phonèmes, portant un message, ne peut pas être modifiée aléatoirement sans énormément dégrader le message. La séquence ne pourra pas non plus être modifiée si les phonèmes sont considérés par groupes de phonèmes, ces groupes marquant les règles de construction syllabique de la parole et, plus généralement, de la langue.

Le nombre de phonèmes d'une langue particulière peut également avoir une influence sur la vitesse de transmission d'un message et donc sur le temps nécessaire à la transmission. L'influence du nombre de phonèmes est une simple application de la théorie de l'information puisque la longueur des messages est inversement proportionnelle au nombre de symboles qui servent à le composer. Ainsi, une langue pauvre en voyelles, comme le japonais, ou pauvre en consonnes, comme le malgache, imposera la création de mots, et donc de phrases plus longues, pour faire passer un message qui aurait pu nécessiter moins de temps de transmission. Le temps de transmission influe également sur la fragilité du message, plus le message est long et plus il est exposé aux agressions de l'environnement. Cependant, la longueur du message permet aussi une plus grande redondance de l'information et permet de comprendre le message à partir d'îlots de confiance qui peuvent être plus nombreux. Mais la pauvreté en phonème d'une langue, et son influence sur le temps de l'échange, n'est pas la seule caractéristique qui marque la temporalité de la parole.

Une autre caractéristique qui marque la temporalité de la parole reste, jusqu'à présent, typiquement humaine. Lors d'une conversation, ou lors d'un cours, un auditeur peut très bien prévoir la fin d'une phrase entamée par le locuteur. La prédiction qu'il est possible de faire sur la fin d'une phrase marque une apparente logique dans la suite des événements sonores, la prédiction se faisant généralement à partir de données périodiques. Une telle prédiction, en plus de l'aspect temporel, indique aussi, dans le contexte considéré, la bonne compréhension du contexte par l'auditeur et la pauvreté en information du message du locuteur, le facteur de branchement ayant été réduit à 1. Ce processus pourrait trouver sa place dans une application automatique mais seulement si le vocabulaire était très limité, réduisant l'horizon de la prédiction à la fin du mot qui est prononcé à un instant donné. Les applications actuelles de la RAP sont en effet généralement limitées et les messages de l'homme vers la machine sont toujours très porteurs d'information, sauf dans le cas de machines à dicter ou toute prédiction semble à jamais impossible.

5.2.2/ Modèles de traitement auditif des sons

Comme nous l'avons vu dans le paragraphe précédent, la parole est un phénomène temporel par les caractéristiques d'enchaînement des événements sonores mais également par la prédiction qui peut être faite sur la séquence de production de ces événements.

Certains modèles, neurophysiologiques ou psychologiques, essaient de décrire simplement ces caractéristiques. Il est possible, à partir des idées exposées, d'extrapoler des modèles connexionnistes ou des automates qui permettent de modéliser le comportement perceptif. Nous allons maintenant présenter quelques unes de ces architectures qui nous semblent intéressantes pour représenter tant la perception des durées d'événements que la perception de schémas qui peuvent être rappelés ultérieurement.

5.2.2.1/ Modèle de la cascade

Une des architectures neurophysiologiques intéressantes dans le domaine de la reconnaissance automatique de la parole est le modèle de la cascade de McClelland [mcclelland79] qui est en fait intéressant pour toute recherche utilisant les paradigmes du connexionnisme, tant dans le domaine de la reconnaissance de la parole que dans celui de la reconnaissance d'images. Ce modèle pourrait être rapproché des concepts d'agents et de traitements parallèles concurrents qui sont étudiés aujourd'hui en informatique. Il peut également être vu comme une variante connexionniste et simplificatrice des architectures de tableaux noirs [lesser75]. Cependant, à la différence des architectures de tableau noir et bien que le modèle soit divisé en strates à la sémantique bien définie, il n'est pas ici possible de retrouver la notion d'expert spécialiste dans son domaine, la connaissance étant, comme dans tout modèle connexionniste, répartie au sein des unités du réseau.

Le modèle de la cascade est fondamentalement opposé au concept du modèle à étages disjoints, *discrete stage model*, de la neurophysiologie et en propose une alternative. Ce concept de modèle à étages disjoints, dont on pourra trouver des exposés dans [sternberg69] et [hunt78], militent pour une organisation séquentielle et ordonnancée des traitements. Ainsi, une décision à un niveau n ne pourra être prise qu'après que toutes les décisions du niveau n-1 aient été prises. Identiquement, une activation à un niveau n ne pourra être calculée que lorsque toutes les activations de la couche inférieure auront été calculées. Le concept de modèle à étages disjoints peut être vu comme étant très proche des perceptrons multicouches. Dans ceux-ci, le calcul des activations se fait suivant un flot de la couche d'entrée vers la couche de sortie et les activations des neurones de chaque couche ne sont calculées que lorsque toutes les activations de la couche inférieure ont été calculées. Il est possible de considérer ce mode de calcul comme très cartésien, chaque couche représentant une étape dans la résolution du problème et toutes les étapes étant clairement séparées les unes des autres.

Le modèle de la cascade dévie fortement de ces notions. Les activations sont ici constamment recalculées par les neurones en fonction de leurs valeurs précédentes et des activités de la couche inférieure. Bien que l'auteur insiste dans son article sur la notion du calcul de l'activité en fonction des résultats des processus de la couche inférieure, comme cela se fait dans un perceptron, l'activité de chaque neurone est calculée de manière dynamique en se référant à l'équation générale aux différences donnée par l'équation 5.1. Cette équation montre clairement que l'activation de chaque neurone est fonction de l'activation au pas de temps précédent. L'unité a_{nj} (unité j de la couche n) voit sa valeur modifiée en fonction de i_{nj}, le niveau d'activation de l'entrée de la cellule a_{nj} au temps t, cette variation étant, elle, fonction d'un coefficient k_{nj} appelé constante de taux (*rate constant*).

$$\frac{d}{dt}(a_{nj}(t)) = k_{nj}(i_{nj}(t) - a_{nj}(t)) \qquad \text{(Éq. 5.1)}$$

L'équation 5.1 accepte une solution générale qui permet de calculer l'activation d'une unité a_{nj} en fonction de la présentation d'un stimulus S au temps $t = 0$. $a_{ns/S}$ représente l'activité asymptotique de l'unité a_{nj} dans le cas où le stimulus S serait laissé indéfiniment en entrée de l'unité, les k_i étant les constantes de taux [mcclelland79].

$$a_{\mathrm{nj/S}}(t) = a_{\mathrm{nj/S}}\left(1 - \sum_{i=1}^{n} K_i e^{-k_i t}\right) \qquad \text{(Éq. 5.2)}$$

Les coefficients K_i de l'équation 5.2 sont des constantes dont le mode de calcul est donné par l'équation 5.3. Cette équation implique évidemment que tous les k_i soient différents les uns des autres. Ces constantes K_i sont attachées aux différents termes exponentiels de la somme.

$$K_i = \prod_{l \neq i}^{n} \frac{k_l}{k_l - k_i} \qquad \text{(Éq. 5.3)}$$

Dans ce modèle, les neurones sont beaucoup plus indépendants les uns vis-à-vis des autres et effectuent des traitements qui sont sans nul doute plus complexes que ceux effectués par les neurones de McCulloch et Pitts. En plus de la prise en compte de l'activation de la couche inférieure (le terme i_{nj} de l'équation 5.1), le neurone tient compte de son activation passée. Ceci permet de commencer à modéliser une activation qui est véritablement locale au neurone, un neurone réévaluant sans cesse son activation en fonction de son apprentissage, de son activation au pas de temps précédent et des activations de la couche inférieure, couche qui représente son environnement. Le recalcul se veut véritablement constant et n'est donc pas dépendant d'un certain flot de données. On peut très bien concevoir qu'un neurone d'une couche n voit son activation calculée à partir d'activations de la couche inférieure dont les indices temporels seraient t et t-1 plutôt que d'être tous du temps t, le respect strict des indices temporels marquant l'ordonnancement des calculs du modèle à étages disjoints.

À un niveau plus général, il est aisé de constater que l'architecture générale d'un ensemble de neurones en cascade ne varie pas des modèles séquentiels et correspond, grosso modo, aux architectures des perceptrons multicouches. La figure 5.3 présente l'architecture donnée par l'auteur dans son article [mcclelland79]. Elle permet de constater que la différence, si elle existe, ne peut être vue qu'à un niveau microscopique dans le réseau et non à un niveau méso ou macroscopique.

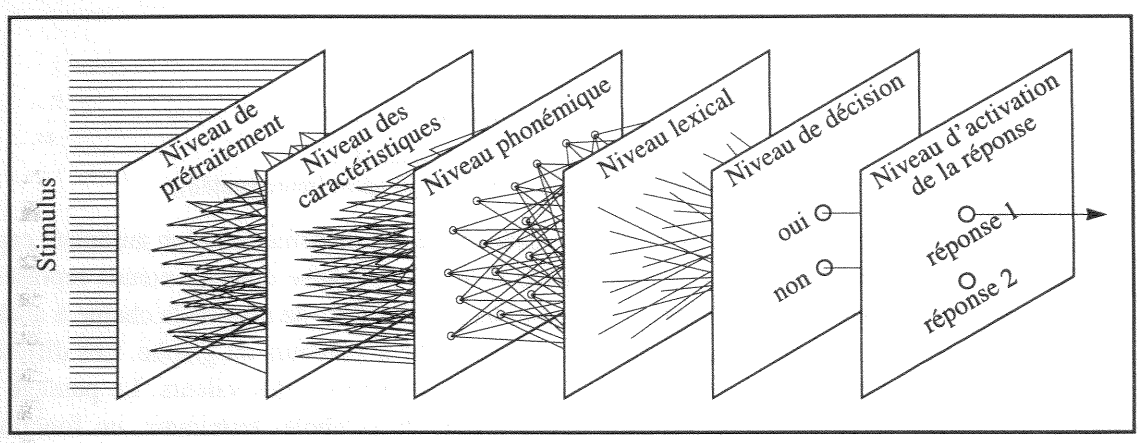


Figure 5.3 : Modèle en cascade de McClelland (d'après [mcclelland79]).

Le modèle de la cascade constitue donc un pas sur la voie de la complexification de la définition du neurone et du réseau tout entier puisque l'architecture générale seule ne permet pas de connaître les fonctions implantées et les traitements possibles, ajoutant encore à l'effet "boîte noire" des réseaux de neurones. Mais la définition d'unités capables d'effectuer des traitements complexes à un niveau local peut être poussée encore plus loin qu'elle ne l'est dans le modèle de la cascade. Il est en effet possible de définir des unités implantant des automates.

5.2.2.2/ Machine multiniveau d'automates

Les automates d'états finis sont des outils d'une grande puissance qui permettent de représenter simplement des grammaires. De nombreux modèles informatiques en ont été dérivés pour reconnaître ou traiter des grammaires complexes. Les automates de traitement de ces grammaires sont alors implantés informatiquement à l'aide de la structure de données de graphes. Des graphes ont été mis en œuvre dans des tâches de reconnaissance de la parole [sakoe78]. Certains travaux ont permis de complexifier le concept d'automate en mettant en place des procédures de traitement au niveau des nœuds [pierrel81] ou au niveau des transitions [laubsch79]. D'autres modèles d'automates ont, enfin, vu leurs transitions augmenter de valeurs indiquant les probabilités de transitions [rabiner89].

Les automates sont d'usage très répandu dans tous les domaines de l'informatique, reconnaissance de la parole comprise. Aussi, avant d'étudier les manières de simuler des automates grâce à

différents modèles mathématiques, nous allons étudier la manière de les définir formellement.

Les automates d'états finis permettent d'analyser de nombreux types de grammaires. Il est possible de représenter mathématiquement ces dernières grâce à un ensemble de données et de règles regroupées sous forme d'ensembles. Formellement, cela se traduit par un système G dont les constituants sont donnés par la formule $G = (N, T, \rightarrow, X)$ [marchand88]. N représente l'ensemble des non terminaux de la grammaire G, T représente l'ensemble des terminaux c'est à dire l'ensemble des symboles qui seront effectivement traités, X représente le non terminal à partir duquel toute phrase de la grammaire peut être écrite et la flèche représente l'ensemble des règles de réécriture de la grammaire G. Cette représentation des grammaires permet de traiter des séquences de symboles éléments de T lorsqu'est fournie la liste des règles de la grammaire G dont un exemple est donné dans la figure 5.4. La grammaire est ainsi représentée sous la forme d'un ensemble de règles de réécriture qui utilisent aussi bien des terminaux (lettres minuscules) que des non-terminaux (lettres majuscules). Cette mixité entre les terminaux et les non-terminaux provoque la mise en relation de symboles n'ayant pas le même niveau d'abstraction.

$$G = (N, T, \rightarrow, X)$$

$$X \rightarrow aA|aB|bA|cC$$
$$A \rightarrow \Lambda|bA|bB$$
$$B \rightarrow cA|bX$$
$$C \rightarrow aA|\Lambda|bC|aX$$

Figure 5.4 : Un exemple de grammaire formelle (d'après [marchand88])

La définition formelle d'une grammaire est la première étape de définition d'un automate capable de la reconnaître. L'automate est construit sous la forme d'un graphe dont les nœuds représentent soit des états de transition, soit des états d'acceptation, soit des états de rejet de la chaîne à analyser. Un nœud particulier du graphe sert de nœud de départ pour le parcours du graphe. Les différentes transitions du graphes ne sont pas, à proprement parler, associées à des valeurs. Le passage d'un nœud à un autre se fera, lors de l'analyse d'une chaîne de symboles terminaux, en fonction du symbole traité qui sert donc de valeur de transition. La définition formelle d'un graphe permet de mieux appréhender cette construction puisque la définition d'un graphe se résume à $Gr = (X,U)$ où X est l'ensemble des nœuds du graphe et U est l'ensemble de transitions, chaque transition étant définie par un couple de nœuds de X [mery95]. Le graphe ainsi défini supprime totalement les références aux non terminaux. Seuls les symboles terminaux sont conservés pour définir les transitions et plus aucune référence aux non terminaux n'est faite. Au mieux, certaines sous-parties du graphe peuvent elles être retrouvées par rapport aux règles de réécriture mais aucun niveau d'abstraction ne sera plus identifiable, pas plus que ne pourra être connue la profondeur de la récurrence sur les règles de réécriture. Toutes ces règles sont donc projetées sur un plan de même abstraction et seuls les terminaux apparaissent.

La projection sur un même plan de symboles de niveaux différents va à l'encontre des principes qui ont été exposés pour la représentation en cascade de la chaîne perceptive. Il semble d'ailleurs très difficile de vouloir définir une grammaire apte à prendre des décisions complexes et de haut niveau à partir d'un simple stimulus d'entrée telle que la parole ou tout autre catégorie de stimuli de bas niveau et très redondants. Le graphe qu'il faudrait définir dans ce cas serait de taille imposante et difficilement gérable. Cette projection sur un même plan peut cependant être contournée grâce à des machines multiniveaux qui permettent, elles aussi, de représenter des grammaires complexes tout en conservant, cependant, certains des niveaux d'abstraction de la grammaire.

Un exemple de définition d'une machine multiniveau pourra être trouvée dans [dimartino87] et [dimartino93]. Les unités de base d'une telle machine sont appelées des cellules. Ces cellules sont en

fait des automates assez simples possédant plusieurs nœuds de départ, les états d'entrée, et plusieurs nœuds de fin, les états de sorties. Ces cellules sont reliées entre elles par des transitions qui sont appelées ici des liens sémantiques (cf. figure 5.5). Cette liaison permet de constituer une machine. En considérant que les cellules sont ici du niveau le plus bas possible, le niveau 0, l'agrégation de ces cellules entre elles permet de constituer une machine de niveau 1 qui n'est en fait rien d'autre qu'un automate (cf. figure 5.6).

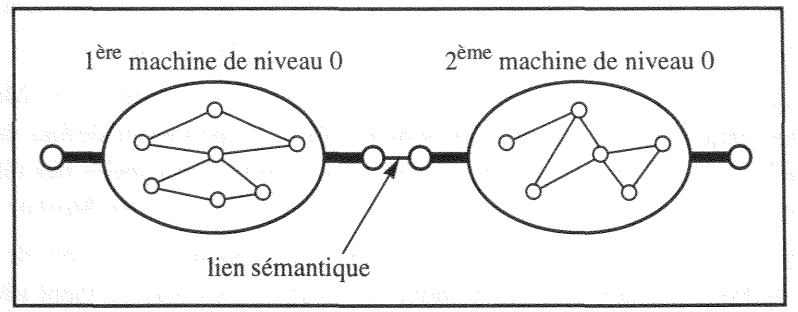


Figure 5.5 : Un exemple de lien sémantique dans une machine de niveau 1 (d'après [dimartino87])

Ce processus peut être itéré autant que nécessaire vis-à-vis de la tâche, cette itération de construction conduisant à la définition d'une machine de niveau *n*. Si les cellules de base, qui sont des machines de niveau 0, ne traitent que des symboles terminaux, les machines de niveau supérieur seront, elles, en prise avec des symboles abstraits qui sont équivalents aux non terminaux des règles de réécriture des grammaires formelles.

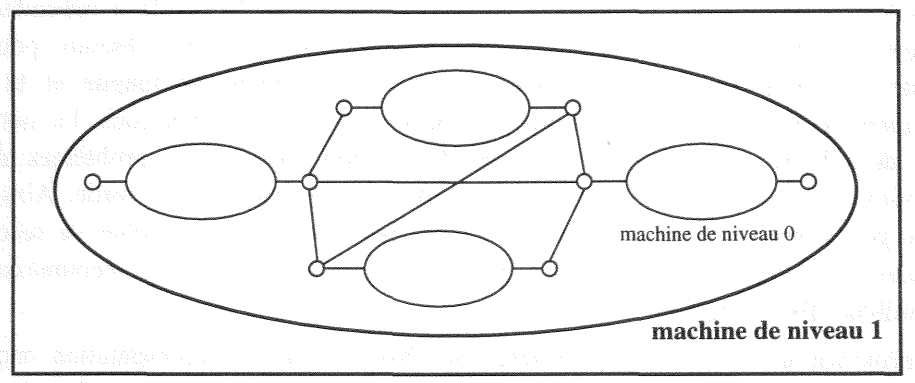


Figure 5.6 : Un exemple de machine de niveau 1 (d'après [dimartino87])

Cette notion de machine multiniveau d'automates n'est pas un concept isolé et des résultats similaires à ceux de [dimartino87] pourront être retrouvés dans [colla85] [bush86].

Ces définitions de machines multiniveaux permettent d'obtenir une alternative, grâce à l'utilisation explicite d'automates, aux modèles de Markov. [dimartino94] compare d'ailleurs les résultats d'une même tâche de reconnaissance de la parole en employant des automates simples d'une part et des réseaux de Markov à états cachés d'autre part. Les résultats obtenus sont équivalents et l'article présente les avantages et les inconvénients respectifs des modèles en fonction des connaissances de l'époque, qui sont déjà dépassées puisque les modèles de Markov ne cessent de progresser du fait de leur grande popularité et du peu d'intérêt que la communauté porte aux machines multiniveaux d'automates. Ce paradigme de machine à plusieurs niveaux de traitement est également étudié aujourd'hui par le biais des réseaux neuromimétiques récurrents [hihi96] utilisant des temporisations au niveau des connexions synaptiques à la manière de [kim92] (cf. chapitre 6, paragraphe 6.4.3.5).

5.2.2.3/ Réseaux de Markov et réseaux de Markov hybrides

Les réseaux de Markov constituent une extension des automates auxquels ont été ajoutés des

capacités de calcul statistique. Les différents nœuds du graphe représentent les états observables dans la chaîne des symboles terminaux et les transitions du graphe sont associées à des probabilités de passage d'un symbole terminal à l'autre. La somme des valeurs des transitions partant d'un nœud est donc égale à 1 étant entendu qu'il peut exister une transition d'un nœud vers lui-même. Les réseaux de Markov permettent, dans l'absolu, de simuler des automates de grammaire qui peuvent être complexes. L'analyse d'une séquence de symboles terminaux par une chaîne de Markov permet, d'autre part, de connaître la probabilité d'apparition de la chaîne analysée, cette probabilité étant obtenue en effectuant le produit des probabilités des transitions parcourues.

Les nœuds du graphe ayant une sémantique, une extension des réseaux de Markov a été définie pour supprimer cette partie de la définition structurelle. Ceci permet de définir des états qui portent sur des symboles qui ne sont pas connus a priori. Ces réseaux sont appelés des réseaux de Markov à états cachés et sont plus connus sous leur appellation anglaise de *Hidden Markov Models*, *HMM*.

Les HMM sont très employés en reconnaissance automatique de la parole où ils remportent actuellement la faveur de nombreuses équipes de recherche. Ce succès vient très probablement de leurs capacités à modéliser des grammaires qui permettent de définir des séquences d'enchaînement entre différents événements sonores. Les différents états d'un HMM n'ayant pas a priori de sémantique, la grammaire peut en outre être déterminée par apprentissage sur un ensemble de séquences du même type. Il n'est donc pas nécessaire de connaître la grammaire, et donc l'ensemble des symboles terminaux, à l'avance. Les états qui seront définis par la phase d'apprentissage seront supposés quasi stationnaires et marqueront ainsi différents états remarquables de la séquence apprise.

L'utilisation des réseaux de Markov en reconnaissance de la parole a cependant imposé des modifications, des simplifications voire une sous-utilisation de ces réseaux pour des raisons calculatoires. La phase d'apprentissage dans de tels modèles est longue et la faiblesse des probabilités en sortie a imposé d'utiliser quelques astuces mathématiques. Le nombre des états cachés de tels réseaux est en outre très faible pour limiter les problèmes dans la phase d'apprentissage et, encore une fois, la faible valeur des probabilités de sortie. Ainsi, la littérature propose généralement l'emploi de trois états par phonème dans des tâches de reconnaissance de phonèmes et l'emploi d'un état par phonème dans des tâches de reconnaissance de mots [bourlard95a], [bourlard96].

Le principal inconvénient des réseaux de Markov est la représentation modulaire de la connaissance acquise. Un HMM ne représente qu'un seul type de formes et donnera, après analyse d'une séquence quelconque, la probabilité pour qu'elle appartienne à la classe de celles qu'il a appris à modéliser. La classification se fait donc après comparaison des probabilités de sortie de l'ensemble des HMM, chacun représentant un des types de séquences observables. Ce type de représentation modulaire ne correspond pas pleinement au modèle de la cascade exposé précédemment (paragraphe 5.2.2.1). Dans ce dernier modèle, la connaissance est complètement intégrée dans chaque niveau, chaque neurone pouvant voir l'ensemble des activations et des décisions prises au niveau précédent. À l'opposé, un HMM agit sur une connaissance définie localement et ne possède aucun indice d'infirmation de sa solution. Ce type de représentation de la connaissance ne correspond pas non plus à la représentation des connaissances qui est faite par l'intermédiaire des réseaux connexionnistes puisque, dans ce cas, toute la connaissance est synthétisée au sein d'un seul réseau. Les différentes couches d'un réseau connexionniste ne possèdent cependant pas toujours de sémantique, au contraire du modèle de la cascade. Dans un réseau connexionniste, la couche de sortie possède une sémantique du fait de la mise en place des classes lors de la phase d'apprentissage supervisé. La couche d'entrée peut elle aussi posséder une sémantique, en fonction du corpus d'apprentissage et donc des signaux à analyser. La sémantique des couches cachées est, elle, beaucoup plus difficilement accessible et nécessite l'emploi de techniques dites d'extraction de règles [goh91] qui sont de plus en plus étudiées mais dont les résultats ne sont pas toujours très convaincants...

Un autre inconvénient des HMM est leur faible capacité à modéliser des durées. Les probabilités de passage d'un état à un autre ne prennent normalement en compte qu'une notion de distance entre les symboles terminaux définis sur deux états successifs mais ne modélisent pas la probabilité de rester plus ou moins longtemps dans un état donné. Deux possibilités existent pour prendre en compte la durée d'un événement : il est possible de modifier le mode de calcul des probabilités de transition, il est également possible d'effectuer des modifications architecturales sur un HMM dont les probabilités de passage auront été déterminées au préalable. La modification de la définition des probabilités de transition passe par l'utilisation d'une fonction de densité de probabilité temporelle qui doit, elle aussi, être déterminée par apprentissage. Cette fonction temporelle permet de faire varier les probabilités de transition d'un état à un autre au cours de l'utilisation d'un HMM [levinson86], [kenny91]. La modification architecturale, quant à elle, constitue une sorte de modification des probabilités de transitions. Dans ce cas, des états sont rajoutés au réseau avant certains états pour permettre de modéliser une durée minimale. Le symbole terminal est dupliqué dans les états supplémentaires et les transitions de passage entre ces différents états, jusqu'à l'état dupliqué, sont associées à une probabilité égale à 1 pour ne pas modifier la valeur finale des probabilités calculées sur les séquences à analyser [gu91], [gupta91], [robinson92].

L'emploi des HMM impose donc que toutes les solutions possibles soient étudiées dans leur ensemble pour que les probabilités d'occurrence de toutes les réponses soient étudiées a posteriori au niveau décisionnel. Un modèle de la psychologie cognitive permet de voir comment il est possible d'agir, éventuellement, plus efficacement : le modèle de Neisser.

5.2.3/ Le cycle perceptif de Neisser

Un des modèles d'étude de la perception humaine est le cycle perceptif de Neisser [neisser67]. Ce cycle est principalement destiné à modéliser le comportement de haut niveau puisqu'une des étapes du cycle porte sur la mise en concordance des plans et du sens commun avec les faits et les objets observés dans l'environnement courant. On peut cependant se demander dans quelle mesure ce cycle n'est applicable à des tâches de plus bas niveau [lindsay80].

Le but du cycle perceptif de Neisser est de trouver une modélisation plausible aux comportements humain, voire à la compréhension de la langue. Le cycle est composé de trois étapes reliées en boucle. Une première étape concerne la reconnaissance d'objets ou l'identification de buts dans une situation ou un environnement donné. Cette reconnaissance/identification sert de base à la découverte des schémas mentaux ou des plans mis en œuvre par les parties prenantes d'un dialogue ou d'une relation. La sélection des plans ou des schémas mentaux entraînera, elle, une phase de recherche active pour identifier de nouveaux buts ou reconnaître de nouveaux objets qui permettront de confirmer ou d'infirmer les plans et les schémas mentaux préalablement sélectionnés. Le mécanisme de ce cycle perceptif est résumé dans la figure 5.7.

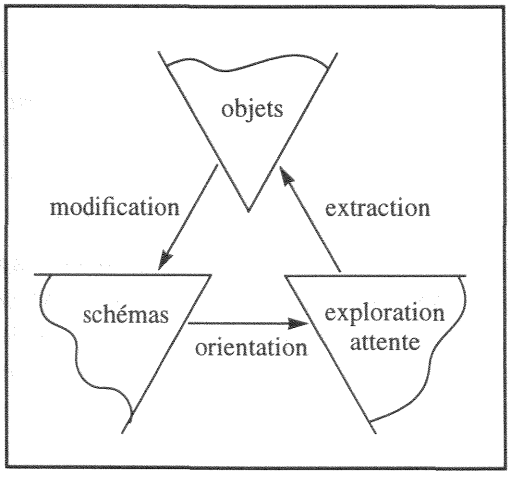


Figure 5.7 : Le cycle perceptif de Neisser (d'après [neisser67])

Cette modélisation de la perception est principalement définie pour tenter de comprendre le raisonnement et le comportement à un haut niveau. Il est cependant intéressant de se demander si ce cycle, qui permet une adéquation constante entre la modélisation choisie de l'environnement et l'environnement lui-même, en constante évolution et répondant rarement à un modèle général prédéfini, ne serait pas applicable à autre chose que le comportement.

Dans le domaine de la reconnaissance de la parole, il est ainsi possible d'envisager de définir des schémas pour certains types de locuteurs comme, par exemple, un schéma pour les voix d'hommes, un pour les voix de femmes et un dernier pour les voix d'enfants puisqu'il est reconnu que ces grands types de voix ne sont pas les mêmes. Il est même possible d'envisager d'aller plus loin puisque l'ensemble des locuteurs pourrait être divisé en sous catégories correspondant à des types de voix qui pourraient ainsi être regroupées en *clusters*. Des études pour démontrer ce fait ont été menées avec succès [pisoni93] et laissent à penser qu'une trop forte agrégation des connaissances peut nuire à la qualité des capacités de reconnaissance. La désagrégation ne doit cependant pas se faire dans le sens où elle est faite dans les HMM mais de manière orthogonale à celle qui y est faite : la désagrégation ne doit pas être faite suivant l'axe des différents modèles et de manière détaillée mais suivant l'axe des types de production qui peuvent être retrouvés dans tous ces modèles et, ce, de manière grossière.

Dans le domaine de la modélisation du bruit, il est également possible d'envisager de telles schématisations puisque les bruits peuvent être regroupés en différentes catégories, non en fonction de leur origine, militaire ou civile, d'intérieur ou d'extérieur, mais en fonction de leurs similarités spectrales puisque les bruits, malgré leur grande variété, ont parfois des similitudes non négligeables lorsqu'ils sont comparés deux à deux (voir à ce sujet l'annexe 3).

La définition de schémas de parole tout autant que la définition de schémas de bruits se heurte cependant à notre compréhension de certains mécanismes cognitifs. Certaines études des mécanismes d'assimilation de mélodies musicales les unes aux autres laissent entrevoir des mécanismes d'assimilation des bruits entre eux qui ne doivent pas être fondamentalement différents des mécanismes d'assimilation de la parole. Les facteurs étudiés dans [pisoni93] sont, entre autres, des facteurs de variations du style de locuteur et de vitesse d'élocution, facteurs qu'il est possible de rapprocher des concepts de hauteur et de hiérarchie de notation en musique.

5.2.4/ Bruit et musique

Ce paragraphe ne se veut pas polémique envers les musiciens et nous n'aborderons donc pas la musique expérimentale. Nous évoquerons cependant ce type de musique moderne qu'est l'improvisation de jazz qui est exécutée lors de concerts, un artiste ou un *band* improvisant totalement un morceau en restant cependant à une distance relativement proche d'un morceau de jazz connu. Tout l'art des interprètes est, dans ce cas, de faire preuve d'imagination tout en restant dans des tonalités et dans une mélodie connues.

L'étude des similarités entre un morceau original, qui sert de référence, et une improvisation permet donc de vérifier et/ou de quantifier une distance qui est a priori très difficile à définir mathématiquement. Nous nous retrouvons en quelque sorte devant une tâche consistant à déterminer si un *copyright* peut restreindre ou non la diffusion d'une œuvre artistique autre que l'original. Le lecteur conviendra, j'en suis sûr, de la difficulté d'une telle tâche.

Ce problème peut être résolu mathématiquement, au moins partiellement, à l'aide d'un réseau auto-associatif récurrent (RAAM, *Recursive Auto-Associative Memory*). Les problèmes de mesure des similitudes, des substitutions, des élisions et des insertions ont été étudiés sous deux aspects dans [large95a] : test de ressemblance de l'improvisation à l'original (*tests of well-formedness*) et tests de structure de représentation (*tests of representational structure*). Ces tests ont bien sûr été faits sur des corpus musicaux limités avec un faible nombre de morceaux originaux. Les résultats sur ces cas restreints sont cependant de bonne qualité et viennent confirmer les hypothèses de représentation

réductionniste de la musique chez l'homme [dowling86]. Cette théorie prône l'idée qu'il existe un codage compact de la musique sous forme de séquences auxquelles un auditeur se réfère lorsqu'il écoute un nouveau morceau, ces représentations compactes servant de base de références à une comparaison et une éventuelle prédiction des séquences musicales auditionnées.

Des conclusions similaires ont été faites dans [mozer94]. Le but des études menées ici est de juger, subjectivement, la qualité des séquences musicales produites par un réseau de neurones récurrent (cf. figure 5.8) effectuant une tâche de prédiction à partir des notes précédentes. L'apprentissage du réseau peut être effectué sur un grand nombre de types musicaux dont un sous-ensemble des œuvres de Bach utilisées par l'auteur. Après apprentissage, le réseau est laissé libre de prédire les notes d'une séquence musicale qu'il compose en fait seul, en prenant en compte les notes prédites lors des pas de temps précédents. L'auteur observe dans ce cas un bon comportement au niveau local mais un piètre comportement au niveau global, les mélodies semblant assez "décousues".

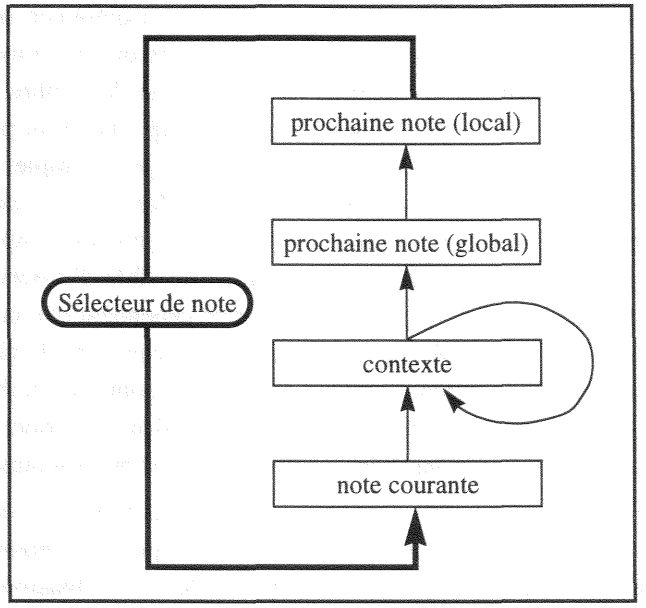


Figure 5.8 : L'architecture CONCERT (d'après [mozer94])

Pour améliorer le comportement de son réseau, l'auteur a mis en place un mécanisme permettant de prendre en compte la mélodie à une plus grande échelle. L'équation de mise à jour du contexte, qui correspond initialement à l'équation 5.4, est ainsi transformée en l'équation 5.5. Dans cette dernière équation, le facteur τ permet de prendre en compte le contexte précédent de manière directe. Cette prise en compte permet d'obtenir une meilleure vision globale des phénomènes en cours de production ou d'apprentissage.

$$c_i(n) = f\left[\sum_j w_{ij} x_j(n) + \sum_j v_{ij} c_j(n-1)\right] \qquad \text{(Éq. 5.4)}$$

L'utilisation du facteur τ permet d'atténuer la rapidité du changement dans les unités de contexte. Cette rapidité de changement sera d'autant plus atténuée que le coefficient τ sera proche de 1. Ce coefficient doit cependant être déterminé par le concepteur du réseau qui doit donc analyser les séquences musicales par lui-même pour déterminer ce coefficient.

$$c_i(n) = \tau_i c_i(n-1) + (1-\tau_i) f\left[\sum_j w_{ij} x_j(n) + \sum_j v_{ij} c_j(n-1)\right] \qquad \text{(Éq. 5.5)}$$

Dans le domaine de la composition automatique de musique, le réseau CONCERT est jugé de manière très positive par son concepteur qui le juge supérieur à d'autres méthodes utilisant pourtant des paradigmes similaires, telles que, par exemple, la table des transitions de [lorrain80] qui

représente, dans une matrice, les probabilités de transition d'une note à l'autre, à la manière d'un automate probabiliste.

Plus généralement, l'audition de séquences musicales appelle quelques remarques sur la psychologie de la perception. [spender93] note que le cerveau est prédisposé à reconnaître les régularités ou l'organisation non aléatoire des structures et, ce, indépendamment de la modalité sensorielle. Un auditeur aura ainsi tendance à chercher une structure, ou une règle, et à l'impliquer inconsciemment au reste du message, une fois la structure établie. La perception sonore est cependant limitée par la mémoire à court terme ce qui rend difficile la compréhension des structures étendues. Ainsi, les palindromes musicaux sont beaucoup plus difficilement discernables alors que leur découverte ne pose généralement aucun problème dans la modalité visuelle. [spender93] voit, dans ces capacités et dans d'autres, le lien possible entre l'audition et la théorie du Gestalt [tenney80], [rock91].

D'autres études sur la perception de la musique laissent à croire que celle-ci est représentée grammaticalement par chaque auditeur en fonction de son expérience musicale propre et que ce codage influe sur l'écoute postérieure d'autres œuvres. Les quatre grandes caractéristiques de la musique, à savoir la puissance, le rythme, la hauteur et le timbre, ne sont cependant pas indépendantes les unes des autres et ne permettent pas, lorsqu'elles sont prises isolément, de définir de telles grammaires. La notion de donnée sensorielle brute, par exemple, qui permettrait de calquer directement les événements physiques du monde extérieur dans les processus cognitifs est un leurre hérité d'un réalisme pythagoricien. C'est pourtant sur cette notion que s'appuie le modèle de l'oreille de Helmholtz qui code chaque hauteur de son qu'il est possible d'observer par une fibre nerveuse différente. Des études, citées dans [spender93], montrent cependant que la perception des intervalles de temps entre les notes est influencée par la différence de fréquences, la différence de temps influant bien évidemment sur le rythme perçu. D'autres études ont montré qu'il était possible d'entendre la fréquence fondamentale malgré l'absence totale d'énergie dans sa propre fréquence, cette présence étant perçue lorsque trois ou quatre harmoniques adjacentes sont présentes.

Toutes ces connaissances ne sont cependant pas encore prises en compte dans les systèmes tentant, aujourd'hui, de modéliser la perception de la musique [carpinteiro96]. Les études en cours tentent plutôt de modéliser la perception de séquences grâce à l'utilisation conjointe des paradigmes supervisés et non supervisés comme cela peut se faire en reconnaissance automatique de la parole [chappell93], [kangas94], [durand95].

Une grammaire de perception musicale est acquise par chacun de nous, même sans aucune formation musicale et, dans ce cas, par une simple écoute d'œuvres plus ou moins classiques. Cette grammaire suppose que chaque note soit, par exemple, codée selon sa hauteur tonale et son niveau harmonique mais également suivant une fonction grammaticale, forgée au fil des expériences, qui, par exemple, permet à chacun de déceler une fausse note dans l'exécution d'une œuvre. Cette fonction grammaticale fait référence à des interactions entre les niveaux rythmiques et tonaux et étend la perception d'une œuvre à un passé et un futur très proche, à la mémoire et à l'anticipation à court terme. "Le présent, pour ainsi dire cognitivé, n'est pas une arête tranchante mais un dôme d'une certaine largeur. Assis sur lui, nous pouvons regarder dans les deux directions au même moment" [james90]. L'écoute d'une œuvre est donc toujours critique, la qualité de cette critique étant en rapport avec notre capacité et notre expérience musicales et avec la grammaire qui en découle. [lashley51] prend ainsi la musique en exemple lorsqu'il insiste sur le fait que toute activité humaine séquentielle (parole, geste, capacités motrices et perceptuelles) est basée sur une grammaire et fondée sur une organisation hiérarchique des décisions, comme cela se retrouvera plus tard dans [neisser67].

5.3/ Le bruit

La reconnaissance automatique de la parole doit faire face à de nombreux problèmes. Au rang de ceux qui limitent son applicabilité et sa diffusion se trouve le problème du bruit. L'étude de la résistance des systèmes de RAP au bruit, bien que n'étant pas nouvelle [dersch63], s'est énormément développée durant ces dernières années [gong95], alors même que les systèmes conçus en laboratoire devenaient commercialisables bien qu'étant encore d'une utilisation contraignante.

D'une manière générale, ce problème peut être compris en comparant les conditions de bruit auxquelles sont soumis les systèmes de RAP développés en laboratoire, où les contraintes de bruits étaient initialement presque inexistantes, avec les conditions de bruit qui existent dans les environnements "réels", conditions qui n'étaient pas initialement reconnues. C'est cette différence qui explique en partie les problèmes de mise en œuvre.

Cette différence est due à deux raisons distinctes. La première est l'éventuelle dégradation de la qualité de la parole, l'environnement pouvant agir négativement sur l'émetteur du message (cf. chapitre 3, figure 3.2). La deuxième et principale raison correspondant à la différence acoustique pouvant exister entre les environnements de mise au point et de mise en œuvre d'un système [stern95]. Les conditions environnementales de développement qui sont prises en compte lors de l'apprentissage peuvent en effet être assez éloignées de celles qui seront rencontrées dans l'environnement effectif d'utilisation. Cet éloignement conduit à la mise en place de techniques qui essaient, majoritairement, de débruiter le signal pour le rendre le plus proche possible d'un signal de parole tel que ceux qui ont été étudiés pendant les premières années de développement de systèmes de RAP. Ces techniques permettent de conserver les acquis des recherches déjà effectuées en laboratoire dans des environnements contrôlés tout en permettant une mise en œuvre rapide de la RAP dans des environnements acoustiquement éloignés de ceux initialement étudiés. [gong95] présente une équation permettant de résumer ces techniques de manière élégante et succincte avec l'équation 5.6. Dans cette équation, un système de RAP q ayant appris à reconnaître des phrases ou des mots d'un corpus S présentés dans un environnement α est adapté à un nouvel environnement β grâce à une fonction f.

$$q_\beta(S) = f(q_\alpha(S)) \qquad \text{(Éq. 5.6)}$$

Cette équation est suffisament générale pour laisser la possibilité d'agir de différentes manières. Ainsi, la technique du débruitage dont nous venons de parler correspondra à une application de f sur β de manière à ce que $\alpha = f(\beta)$. Une autre technique, consistant à adapter les paramètres internes du modèle pour le rendre compatible avec la nouvelle condition de bruit, l'adaptation, peut être décrite comme l'application de f sur q_α de manière à ce que $q_\beta = f(q_\alpha)$.

Mais le débruitage et l'adaptation des modèles initiaux ne sont pas encore des techniques universellement applicables puisqu'elles sont à l'heure actuelle cantonnées au traitement des seuls bruits stationnaires. Il est donc parfois nécessaire de s'appuyer sur des techniques plus rapides et directes pour extraire directement des indices dans le signal bruité.

5.3.1/ Techniques de reconnaissance de la parole en milieu bruité

Comme nous l'avons déjà vu au chapitre 1, paragraphe 1.7.4, il existe trois grandes classes de techniques pour améliorer les capacités de résistance des systèmes de RAP au bruit. Ces techniques sont nombreuses et sont plus ou moins bien adaptées au traitement d'un signal de parole déformé par un bruit additif ou convolutionnel.

La première classe de techniques consiste à ne pas différencier la parole bruitée de la parole non bruitée et à considérer le système de RAP comme étant indépendant des conditions de bruit. Cette technique conduit à l'utilisation de mesures de distances et à l'extraction d'indices acoustiques dont la résistance au bruit est connue et sûre. C'est l'approche que nous avons choisie lors de cette thèse.

Les indices robustes peuvent être obtenus grâce à l'analyse linéaire discriminante ou à la prédiction linéaire.

Une deuxième classe de techniques consiste à transformer le signal de parole bruité en un signal le moins bruité possible qui soit le plus proche possible en qualité d'un signal de parole non bruité. Cette technique essaie donc d'effectuer une amélioration qualitative du signal d'entrée. Le bruit est donc réduit avant que le signal de parole ne soit traité par le système de reconnaissance. Cette réduction peut se faire dans le domaine spectral ou dans le domaine cepstral, par soustraction ou filtrage du signal original.

Une troisième technique, enfin, essaie de transformer les modèles de référence de la parole de l'environnement d'origine, où a été fait l'apprentissage, en des modèles tenant compte du bruit de l'environnement effectif. Cette technique effectue donc une adaptation, ou compensation, des modèles au bruit. Contrairement à ce qui est fait par les techniques d'amélioration du signal, le bruit n'est pas amoindri et sera présent lors de l'étape de reconnaissance puisqu'il est considéré comme une partie du signal à traiter. L'adaptation peut se faire par utilisation de modèles en parallèle, par utilisation de prototypes de bruit ou adaptation directe des paramètres du système de décodage de la parole, par régression linéaire ou ajustement stochastique.

Ces trois grandes classes de techniques constituent aujourd'hui les méthodes de l'état de l'art de la RAP en milieu bruité. Elles présentent cependant des lacunes puisque tous les bruits ne peuvent pas encore être traités.

5.3.2/ Modélisation du bruit

5.3.2.1/ Tendances actuelles

L'étude des méthodes précédentes de traitement du bruit permet une constatation très simple : il n'existe pas aujourd'hui de méthode générale permettant de caractériser un bruit quelconque. Les seuls bruits qu'il est donc possible de traiter sont les bruits stationnaires ou quasi stationnaires. Une étude bibliographique du domaine permet de constater la présence constante et presqu'exclusive des bruits stables comme, par exemple, les bruits stationnaires de NOISEX (ce sont les bruits de l'hélicoptère Lynx, de l'avion de chasse F16, d'une conduite intérieure ou de parole synthétique) ou des bruits à consonance industrielle ou ménagère mais stables (un sèche cheveux par exemple). Bien qu'il soit possible de trouver des "microvariations" au sein de ces bruits, leurs spectres (voir annexe 3) permettent de constater qu'aucune véritable variation n'est présente. Ces variations sont d'ailleurs très peu audibles. D'autres bruits, du corpus NOISEX en particulier, présentent des spectres beaucoup plus changeants puisque concernant des bruits non stationnaires.

Ces derniers bruits, non stationnaires, n'ont pas fait l'objet de beaucoup d'études car ils sont mal maîtrisés, principalement parce qu'ils sont très difficiles à modéliser a posteriori. Il est en effet très difficile de caractériser simplement un bruit non stationnaire : il n'est plus ici seulement question de trouver un spectre moyen du bruit mais plutôt de trouver les instants ou périodes de validité de différents spectres qu'il faudrait définir aussi précisément que le sont aujourd'hui les spectres de bruits stationnaires. Il manque donc un outil de modélisation permettant de réaliser un tel système, fiable et efficace, permettant d'obtenir une grammaire du bruit rencontré. Les différents modèles que nous avons vus précédemment semblent offrir une alternative. Le modèle de Neisser en particulier, fondé sur le concept de schémas et de plans en perpétuelle comparaison avec l'environnement perçu, semble être une bonne voie pour une modélisation des bruits non stationnaires.

5.3.2.2/ Modélisation rythmique

Des approches aptes à la modélisation rythmique ont déjà été employées dans le domaine de la reconnaissance automatique de la parole en milieu bruité à l'aide de HMM [gales93] mais les résultats publiés n'ont jamais concerné d'autres bruits que les bruits stationnaires de NOISEX bien que les auteurs aient affirmé en certaines occasions et de manière informelle avoir obtenu de bons

résultats sur des bruits non stationnaires de NOISEX, le bruit de mitrailleuse en particulier. L'approche dont nous parlons ici [nozalco93], [gales95] est basée sur une évaluation en parallèle de différents modèles de bruit dont chaque spectre est considéré comme indéfiniment stable. Elle n'est peut-être pas optimale pour le bruit non stationnaire bien qu'il soit possible de comparer cette approche à celle de Neisser.

Un être humain, d'âge adulte et sans apprentissage identifiable, peut reconnaître un bruit quelconque de manière assez précise bien que cette affirmation soit péremptoire. Cependant, la caractérisation d'un bruit fait appel à deux notions principales : la reconnaissance des différentes composantes fréquentielles du bruit, que celui-ci soit stationnaire ou non, et la reconnaissance des différentes composantes temporelles (ou séquentielles) du bruit, dans ce cas périodique, associée à la capacité de réordonnancer correctement les différentes composantes isolées. L'identification des composantes séquentielles doit obligatoirement faire appel à un mécanisme fondé sur la reconnaissance des différents rythmes pouvant se produire. On se retrouve donc ici devant un problème qui pourrait ressembler à un décodage d'un enregistrement musical avec cependant quelques nuances :

- il n'y a aucune mesure : le bruit perçu n'est pas, a priori, cadencé comme peuvent l'être des notes de musique,
- il n'y a aucune règle de construction : en musique classique, des règles ont été établies pour assurer une harmonie dans l'écriture des partitions. Ces règles sont très majoritairement respectées et constituent une sorte de grammaire générative,
- il n'y a aucune connaissance a priori des différentes sources sonores qui peuvent être rencontrées et donc des spectres sonores qui leurs sont associés.

Certaines études pour générer ([mcauley93]) ou reconnaître ([cummins94], [large95a]) des séquences rythmiques, à l'aide de réseaux de neurones dans les cas cités, ont été entreprises et permettent d'envisager, à terme, que de telles modélisations puissent être faites. Ces architectures se veulent proches des notions neurobiologiques et approximent mathématiquement certaines caractéristiques de la mémoire humaine.

5.4/ Mémoire humaine et mémoire des réseaux connexionnistes

5.4.1/ Quelques remarques sur la mémoire des Hommes

La mémoire est une notion très vaste et difficile à cerner. Le terme lui-même est générique puisqu'il prend en compte des phénomènes qui sont très éloignés les uns des autres dans la dimension temporelle.

À un bout du spectre se trouvent les mémoires relatives aux connaissances et aux jugements responsables du comportement d'une personne. Il existe ainsi une mémoire culturelle relative à l'environnement d'une personne en général, une mémoire sociale relative à son environnement particulier et une mémoire académique relatives aux connaissances scolaires acquises, toutes ces mémoires étant des mémoires à long terme.

À l'autre bout du spectre se trouvent toutes les mémoires à très court terme, en relation avec la perception, qui ne sont généralement pas accessibles à la conscience et très rarement verbalisées. La mémoire perceptive, ou registre de l'information sensorielle [lindsay80], peut être visuelle, tactile ou auditive. Le goût et l'odorat sont également concernés par de tels registres mais l'intérêt que nous y portons reste très limité.

Entre ces deux mémoires, l'une immédiate et l'autre à long terme, existe une mémoire à court terme qui est parfois qualifiée de mémoire de travail. Elle enregistre un nombre restreint de données pendant un laps de temps assez court. Ces données pourront éventuellement être intégrées à la mémoire à long terme grâce à un processus de répétition comme lorsqu'un élève apprend ses leçons.

Quelques règles relatives à la mémoire humaine ont été découvertes grâce aux recherches en psychologie. Certaines de celles-ci ont également été vérifiées pour les modèles formels que sont les réseaux de neurones. Les règles dont nous allons parler n'ont cependant pas été établies à partir d'une modélisation mathématique proche d'une réalité neurobiologique microscopique mais sont plutôt des règles portant sur les mémoires de haut niveau : les mémoires à court et long terme.

Ainsi, [foucault13] exprime mathématiquement le temps nécessaire à l'apprentissage d'éléments d'une liste par une loi non linéaire, le temps requis étant exponentiel par rapport au nombre d'éléments M. Cette loi, qui permet de quantifier le temps de passage des informations de la mémoire à court terme vers la mémoire à long terme, s'exprime en fonction d'une constante c et d'un facteur exponentiel D selon l'équation 5.7 suivante :

$$t(M) = cM^D \qquad \text{(Éq. 5.7)}$$

[foucault13] définit le facteur D comme étant égal à 2. Des études ultérieures [cateau92] ont permis de réexaminer cette loi, tant pour des sujets humains que pour des systèmes formels. Ces études ont ainsi redéfini le coefficient D selon l'encadrement $1<D<2$ car certains sujets humains semblent avoir d'excellentes capacités de mémorisation. La seconde partie de l'étude porte sur la vitesse d'apprentissage d'un réseau connexionniste de type perceptron utilisant la rétropropagation du gradient. [cateau92] a ainsi démontré que la loi de Foucault était à nouveau opérante avec des valeurs de D cependant supérieures à celles établies pour les sujets humains : $2<D<3$. La généralité de la règle de Foucault semble valider la plausibilité neurobiologique du processus d'apprentissage utilisé.

D'autres études ont également été réalisées sur les capacités de mémorisation à court terme, dans la mémoire de travail. Ainsi, la règle de stockage de 7 termes dans la mémoire de travail, plus ou moins 2 termes en fonction du sujet, qui a été exposée dans [miller56], peut être retrouvée dans les modèles connexionnistes [ingber95b]. Cette dernière étude permet d'ailleurs de faire une distinction entre la règle des 7 ± 2 qui semble être vérifiée pour les phénomènes acoustiques et une règle des 4 ± 2, qui semble se vérifier pour les phénomènes visuels, sémantiques ou moteurs.

Toutes ces règles n'ont cependant qu'un rapport éloigné avec la modélisation connexionniste puisqu'elles relèvent plus des capacités issues de l'agrégation d'unités que de capacités d'entités isolées. La modélisation de bas niveau de la mémoire humaine présentée dans [usher95] regroupe certains concepts répandus dans la communauté connexionniste. Ces études se fondent en partie sur [mcclelland79] et en constituent un développement. Les concepts mis en avant dans cette étude sont la propagation graduelle de l'activation par passage à travers les non linéarités des cellules, la variabilité intrinsèque des réponses à des stimuli extérieurs identiques, l'interactivité par excitation récurrente et l'inhibition latérale en excluant toute inhibition aval et, enfin, la représentation locale.

L'activation d'une cellule est modélisée suivant une reformulation stochastique des équations du modèle de la cascade [mcclelland79]. Cette équation est la suivante :

$$\tau \frac{dx_i}{dt} = I_i - kx_i + \xi \qquad \text{(Éq. 5.8)}$$

Dans cette équation, x_i représente l'activité de la cellule, I_i représente l'activité reçue des cellules afférentes et ξ représente un bruit intrinsèque de nature gaussienne. Enfin, le coefficient k correspond au taux passif de décroissance de l'activité. Les relations de ce modèle avec d'autres modèles neurobiologiques sont exposés dans [usher95].

La partie qui nous semble la plus intéressante dans l'équation 5.8 est la présence de la récurrence locale qui, selon les auteurs, est aussi importante pour la modélisation et la représentation de la connaissance que ne l'est la représentation intrinsèquement distribuée utilisée dans les réseaux connexionnistes. Cette récurrence locale n'est pas partie intégrante de la modélisation de McCulloch

et Pitts [mcculloch43] et n'a donc pas été prise en compte par les modèles connexionnistes les plus répandus aujourd'hui comme le perceptron multicouche. Cette récurrence locale semble cependant avoir des capacités modélisatrices non négligeables grâce aux mécanismes d'excitation et d'inhibition retardées. Ces derniers mécanismes pourraient très bien être une forme de mémoire qui n'a pas encore été prise en compte à sa juste valeur.

5.4.2/ Implantation de la mémoire dans les réseaux connexionnistes

Les réseaux connexionnistes formels, comme tous les systèmes de reconnaissance des formes, possèdent une mémoire des différents éléments qui devront être classés selon la définition qui en aura été donnée pendant la phase d'apprentissage. Mais cette mémoire n'est pas la seule qui puisse exister dans ce type de réseaux et d'autres mémoires sont ou peuvent être mises en œuvre.

5.4.2.1/ Mémoire à très long terme

La mémoire à très long terme des réseaux connexionnistes est la mémoire d'apprentissage. Elle correspond à la mémoire académique dont nous avons parlé précédemment. Cette mémoire permet au réseau connexionniste de conserver l'ensemble des définitions des différentes classes qui pourront être rencontrées lors d'une tâche, ces définitions étant synthétisées, dans le cas connexionniste, au sein d'une seule et même structure.

Cette mémoire à très long terme est responsable de la qualité des réponses du réseau et ne doit normalement pas être modifiée. Elle peut l'être parfois dans une moindre mesure [alpaydin91] mais l'énorme majorité des applications pourrait mal s'accommoder de telles capacités.

5.4.2.2/ Mémoire instantanée

La mémoire instantanée des réseaux connexionnistes correspond à la mémoire perceptive humaine, ou mémoire de l'événement courant, que nous avons précédemment qualifiée de registre de l'information sensorielle. Il ne s'agit donc pas véritablement d'une mémoire puisque les valeurs captées en entrée du réseau sont simplement propagées selon l'architecture jusqu'à la couche de sortie. Cette mémoire correspond à la simple présence au sein des unités connexionnistes des résultats des calculs effectués pendant la propagation de l'activité et donc à l'état courant du réseau. La définition que nous venons de donner n'est valable que dans le seul cas discret, la mise en place d'une notion de calcul de l'activation en continu rendant la présentation précédente caduque.

La mémoire instantanée est cependant un concept que nous considérons difficile à définir très exactement. Il est en effet possible de définir l'entrée d'un réseau à partir de vecteurs de données échantillonnés à des instants différents et concaténés pour constituer le vecteur d'entrée. Dans ce cas, la mémoire instantanée du réseau est établie à partir de données d'origine temporelle différente sans que cette différence soit explicitement représentée. L'intégration effectuée par l'étape de prétraitement vient donc amoindrir l'instantanéité de ce type de mémoire et rendre caduque son lien avec une perception immédiate.

Nous avons nous même utilisé ce type d'agrégation temporelle dans notre étape de segmentation définie au chapitre 4 selon le principe exposé par la figure 5.9 et présenté dans [buniet93a]. Le pas de temps que nous avons utilisé pour l'espacement du calcul de deux trames de coefficients cepstraux a été trouvé optimal pour 68 millisecondes en utilisant 5 trames différentes de coefficients calculées sur 32 millisecondes de signal de parole. Ce type d'agrégation a été également utilisé, avec des pas de calcul différents, dans [zhu90] ou [pratt91].

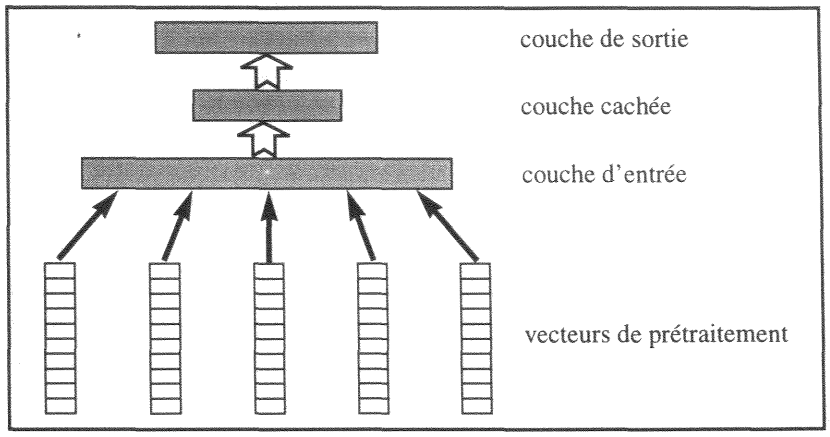


Figure 5.9 : Schéma de principe de l'agrégation de vecteurs de prétraitement d'indices temporels différents.

Cette agrégation de trames en entrée du réseau oblige à recalculer les valeurs des différentes trames avant chaque propagation dans le réseau puisque le pas de temps séparant le calcul de deux trames ne correspond pas forcément au pas de temps qui sera utilisé pour le déplacement de la structure connexionniste lors de la phase d'analyse. Ce dernier pas de temps permet d'obtenir la courbe de réponse du réseau et nous avons utilisé pour ce dernier un intervalle de 10 millisecondes qui correspond au pas standard utilisé dans les études effectuées en RAP.

Il est possible de mettre en place une structure d'entrée qui conserve un signal sur plusieurs pas de temps tout en rendant la structure de mémoire explicite : la mémoire peut alors être qualifiée comme étant de taille finie.

5.4.2.3/ Mémoire de taille finie

La mémoire connexionniste que nous qualifions comme étant de taille finie correspond à une extension, limitée dans le temps, de la mémoire instantanée. Cette extension permet de ne plus avoir à recalculer les différentes trames constituant la couche d'entrée grâce à la mise en place de lignes de délais encastrés (figure 5.10). Une ligne de délais encastrés peut être considérée comme une application au domaine du traitement du signal de la structure de donnée de file, c'est à dire une liste du type premier entré - premier sorti. Une trame est donc calculée à un instant donné puis chacun de ses constituants est fourni en entrée d'une ligne de délais qui conserve cette information pendant un nombre de pas de temps équivalent au nombre de délais qui la constitue.

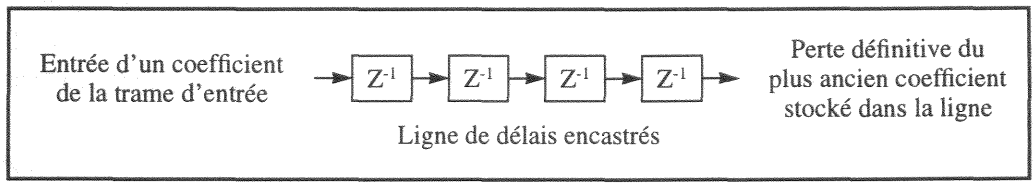


Figure 5.10 : Fonctionnement d'une ligne de délais encastrés.

Contrairement aux trames agrégées dont nous avons parlé dans le cas de la mémoire instantanée, le pas de temps utilisé pour déplacer le réseau connexionniste dans le signal est désormais égal au pas de temps séparant le calcul de deux trames successives utilisées pour la définition du vecteur d'entrée du réseau. La suppression de cette différence est simplement due à la conservation des constituants de la trame dans les lignes de délais de la couche d'entrée.

La mémoire de taille finie a été principalement utilisée en RAP conjointement à la technique de partage des poids conduisant à la définition du modèle de TDNN dont nous avons déjà parlé au chapitre 2 (paragraphe 2.4.1.3). Ce type d'approche a permis de mettre en œuvre des techniques beaucoup plus proches de la lecture de spectrogrammes que celles qui avaient jusqu'alors été développées avec les modèles connexionnistes. Il devenait enfin possible de traiter des formes de

manière géométrique dans un faible intervalle de temps, la mise en place du partage des poids ne faisant que renforcer cette capacité acquise par l'emploi de lignes de délais.

Mais l'emploi de lignes de délais simples pose quelques problèmes. Il est en effet impossible de s'assurer a priori que la taille de la fenêtre d'entrée sera suffisament grande pour résoudre le problème posé. Le choix peut s'orienter vers l'utilisation d'une fenêtre de taille maximale mais cette solution n'est évidemment pas acceptable du fait des problèmes de calcul que cela pose [mozer93], [vries92]. Un bon résumé de ce problème a été donné dans [berthommier92] : "l'horizon temporel d'un système [...] défini à partir d'une mémoire explicite est précisément déterminé par le délai de cette mémoire".

Ce problème ne peut pas être résolu avec une technique ne faisant intervenir que le paradigme statique, le seul qui ait été exposé jusqu'à maintenant dans cette présentation.

5.4.2.4/ L'échelon manquant

La mémoire de taille finie permet au réseau de prendre en compte plusieurs pas de temps mais la définition d'une plaque de taille optimale par rapport au problème est difficile à trouver. Il est donc nécessaire d'effectuer un compromis entre la taille de la plaque, qui devrait être idéalement grande pour qu'aucune information ne soit perdue, et la charge de calcul qui est fonction du nombre de poids exploitant la plaque d'entrée.

Il semble intéressant de réduire la taille d'une plaque d'entrée, définissant l'espace temporel, par l'utilisation d'un mécanisme de récurrence permettant de conserver les valeurs passées par rémanence. Ce mécanisme permettrait de mettre en place une mémoire théoriquement illimitée dans le temps, au contraire de la mémoire de taille finie.

La rémanence, imposant la mise en place d'une récurrence, peut être définie de plusieurs manières. Il est possible de définir une récurrence de manière globale à tout le réseau ou de manière locale à un neurone, ces deux options représentant les extrêmes d'un spectre dont l'échelon intermédiaire est occupé par des réseaux dont la récurrence est définie à partir des différentes couches qui le constituent. Quel que soit le choix effectué parmi toutes ces possibilités, il doit permettre de résoudre le problème soit par création d'une notion implicite de durée, qui remplace une plaque d'entrée de taille variable, soit par élaboration d'un automate qui permet de suivre l'évolution d'une forme en cours d'analyse.

5.5/ Extension à apporter au système

Ce paragraphe présente les extensions fonctionnelles que nous avons jugé bon d'apporter au système précédemment développé. Les modèles de haut et de bas niveau présentés dans ce chapitre nous ont permis de voir quelles étaient les possibilités envisageables pour modéliser des processus cognitifs qui tiennent compte du temps ou de grammaires. Ces modèles sont une étape vers le développement d'un ou de plusieurs systèmes nous permettant de résoudre notre problème.

5.5.1/ Apprentissage de la durée moyenne des phonèmes

La première étape du développement futur passe par la modélisation des durées au sein d'un système également capable d'intégrer des connaissances phonétiques. La qualité des résultats de segmentation à des rapports signal sur bruit assez élevés dans nos précédentes expériences nous poussent à tenter de développer plus avant notre méthode connexionniste.

Il faut donc désormais trouver un modèle connexionniste qui puisse conserver les acquis de notre étape de segmentation tout en y ajoutant des capacités temporelles pour rendre le réseau apte à modéliser les durées des événements phonétiques que nous voulons segmenter. Le choix qui sera fait devra, tout comme pour le système étudié au chapitre 4, répondre aux impératifs énoncés dans le chapitre 3 relatifs à l'applicabilité directe (sans réapprentissage) et immédiate (sans adaptation).

L'étude des systèmes connexionnistes qu'il est possible de mettre en œuvre pour accomplir une

telle tâche sera faite au chapitre suivant où seront présentés les réseaux connexionnistes dynamiques.

5.5.2/ Modélisation du bruit

Une étape supplémentaire de modélisation est envisageable. Au delà de la modélisation de la seule durée des parties vocaliques d'un signal de parole, il est tentant de vouloir développer un système capable de modéliser le bruit de manière assez générale.

La parole est un phénomène très changeant. Elle peut parfois être considérée comme un bruit et donc nécessiter une modélisation lors de l'emploi de techniques d'amélioration du signal. Notre approche à venir, basée sur l'utilisation d'indices robustes d'une part et une modélisation temporelle d'autre part, pourrait servir de base à la définition d'un modèle du bruit de parole connu sous le nom de *babble noise* ou du bruit de *cocktail-party*. Des systèmes plus élaborés pourraient également être définis pour modéliser les bruits sous la forme d'automates qui permettraient d'effectuer la soustraction du bruit dans le signal en fonction de la grammaire apprise et de la comparaison de la prédiction de ces automates au son effectivement perçu, généralisant le concept de combinaison parallèle de modèles utilisé dans [gales95].

Nous effleurerons ce type de modélisation dans le chapitre 8.

CHAPITRE 6 : RÉSEAUX CONNEXIONNISTES RÉCURRENTS

"(au sujet de la formalisation mathématique du système nerveux) En premier lieu, il est exagéré de décrire ce travail comme «une tentative pour comprendre» ; ce n'est qu'un ensemble un peu organisé de spéculations sur la façon dont il faudrait procéder lors d'une telle tentative"

John von Neumann
L'ordinateur et le cerveau

Résumé

Nous présentons dans ce chapitre un état de l'art des différents modèles connexionnistes dynamiques en nous attachant tout particulièrement à la présentation des modèles neuromimétiques à récurrence locale. Cette présentation permettra au lecteur de mieux comprendre le choix architectural que nous avons effectué pour modéliser la durée des phonèmes dans notre étape de segmentation.

6.1/ Taxonomie des réseaux récurrents

6.1.1/ Taxonomie des architectures récurrentes

Les réseaux connexionnistes du type des perceptrons multicouches voient leurs activations calculées selon un ordre précis qui définit une sorte de flot de données au sein du réseau. Les données, stockées dans les cellules d'entrée du réseau, sont utilisées pour calculer les activations des cellules de la première couche cachée. Ces dernières cellules serviront, elles, à calculer les activations des cellules de la deuxième couche cachée. Ce processus est répété jusqu'au calcul des activations des cellules de la couche de sortie. Cette notion de flux de l'activité neuronale peut se retrouver dans l'équation 6.1 décrivant le mode de calcul général mis en œuvre.

$$y_{Nj} = f\left(\sum_{i=1}^{\text{nb(cellules de N-1)}} w_{ji} \times y_{N-1i}\right) \qquad \text{pour N de 1 au nombre de couches du réseau} \quad (\text{Éq. } 6.1)$$

Tous les modèles connexionnistes respectant l'équation 6.1 n'implantent pas, à proprement parler, de mémoire. La seule mémoire présente est instantanée, ou immédiate, puisqu'un seul indice temporel est considéré. Seule l'implantation d'une fenêtre temporelle en entrée de ce type de réseau peut permettre une mémorisation à plus long terme du signal en cours d'analyse mais le principe de calcul des activations reste cependant le même. Les réseaux de ce type sont qualifiés de réseaux statiques. Un autre type d'architecture connexionniste permet de faire émerger une capacité de mémorisation interne : les réseaux récurrents.

Les réseaux récurrents, également qualifiés de réseaux connexionnistes dynamiques, se distinguent des réseaux statiques par un mode différent de calcul des activations neuronales. Dans le cas des architectures récurrentes, un neurone peut théoriquement utiliser les activations de toutes les cellules du réseau pour calculer son activation propre. Ce relâchement dans les contraintes impose en retour de tenir compte de la référence temporelle des activations utilisées pour les calculs. Alors que, dans le cas des réseaux statiques, les données d'entrée et les activations des unités de sortie correspondent à un unique instant t, un réseau récurrent peut prendre en compte des données de l'instant courant et des données du ou des instants précédents selon la définition de l'architecture. Un réseau récurrent peut être modélisé de manière générale par l'équation 6.2 suivante :

$$y_{k,t} = f\left(\sum_{i=1}^{N} \sum_{j=1}^{T} w_{ki} \times y_{i,t-j} \right) \qquad \text{(Éq. 6.2)}$$

Dans l'équation 6.2, aucune restriction n'est imposée pour le calcul de l'activation d'une unité. Le nombre de connexions du réseau peut alors être très élevé et correspondre, pour la borne supérieure, à N^2 connexions pour un réseau de N neurones dont les connexions synaptiques sont orientées et lorsque T est limité à 1. Dans un tel cas, plus le nombre de connexions est grand et plus le processus de mise à jour des activations est gourmand en temps de calcul. Corrélativement, plus le nombre de degrés de liberté, qui correspondent aux nombres de connexions, est grand et moins est forte la probabilité que la phase d'apprentissage converge vers un espace des poids permettant au réseau de résoudre la tâche à apprendre.

Au delà de la définition assez simple d'un réseau récurrent donnée par l'équation 6.2 et du fait des difficultés d'apprentissage, un grand nombre d'architectures ont été définies. Ces architectures, très variées bien qu'étant toutes qualifiées de récurrentes, peuvent être classées en trois grandes catégories selon la manière dont est implanté le mécanisme de récurrence. L'interconnexion entre les cellules du réseau peut tout d'abord être très forte sans qu'aucun sous-élément architectural de grande taille ne soit défini. Nous avons dans ce cas qualifié le réseau comme étant à récurrence forte (paragraphe 6.2). Une autre possibilité pour implanter une récurrence passe par la définition a priori de sous-structures architecturales facilement identifiables et dont le rôle est bien identifié. Nous parlerons dans ce cas de réseaux à récurrence par plaque (paragraphe 6.3). Enfin, une dernière catégorie regroupe tous les réseaux dont la récurrence n'est pas obligatoirement visible au niveau architectural sans une définition préalable des cellules du réseau qui sont elles-même récurrentes. Cette dernière catégorie de réseaux a été regroupée dans notre taxonomie sous le patronyme de réseaux connexionnistes à récurrence locale (paragraphe 6.4).

À lire les lignes précédentes, il est aisé de constater que la taxonomie ainsi définie peut souffrir, comme toute classification, d'un problème de clarté des frontières entre les différentes classes. Cependant, en définissant notre taxonomie des réseaux récurrents, notre but n'était pas d'éviter tout problème diplomatique mais plutôt d'essayer de trouver un cheminement, aussi bien historique qu'architectural, au sein du pandémonium existant pour expliquer l'intérêt que nous avons porté au modèle gamma qui a été utilisé lors de la thèse (cf. chapitre 7).

6.1.2/ Taxonomie des mémoires

Il est intéressant d'étudier, par rapport à notre taxonomie, une autre classification possible des réseaux "temporels" qui a été donnée dans [chappelier94]. Elle n'est pas non plus exempte de critiques mais permet de montrer le domaine à partir d'un autre point de vue. Cette taxonomie se fonde sur une différentiation des architectures de réseaux à partir du mécanisme qui a été choisi pour implanter la capacité de traitement temporel (figure 6.1). Cette taxonomie trouve son développement initial, et inachevé, dans [berthommier92].

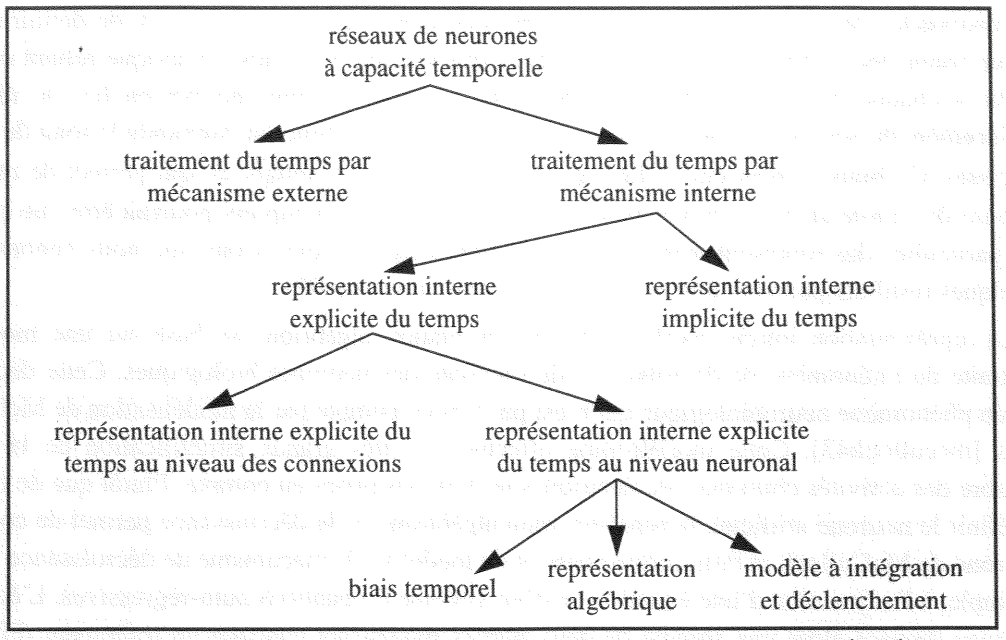


Figure 6.1 : Une classification possible des différents types de réseaux connexionnistes aptes aux traitements temporels (d'après [chappelier94]).

Les réseaux à représentation externes sont typiquement les TDNN, *Time Delay Neural Network* [waibel89]. Dans ces réseaux, le temps est spatialisé et aucune véritable capacité de mémorisation n'est implantée. Ces réseaux sont en fait des cas particuliers de la classe des réseaux statiques. D'autres réseaux, considérés comme étant dynamiques dans notre précédente taxonomie, peuvent cependant être considérés comme étant à représentation externe par certains points de leur architecture.

Les réseaux à représentation interne implicite du temps sont les réseaux totalement récurrents comme le modèle de Hopfield [hopfield82] et la machine de Boltzmann [hinton84], le traitement temporel étant une des capacités sous-jacentes de ce type d'architecture.

La représentation interne explicite du temps au niveau des connexions correspond au cas où le temps est modélisé par l'intermédiaire des connexions synaptiques. Ce type de représentation, assez utilisée, pose généralement le problème du nombre de degrés de liberté du réseau qui est au moins doublé puisque chaque connexion possède alors au moins deux degrés de liberté alors qu'une même capacité expressive peut être obtenue par d'autres modélisations. Il existe différents travaux se rattachant à cet axe : [chappell93], [kangas94].

La dernière sous-classe, la représentation interne explicite du temps au niveau des neurones, correspond aux cas où le temps est un phénomène au niveau du neurone et non plus du réseau ou des connexions. Trois possibilités existent pour mettre en place ce type de traitement local du temps. Un premier cas (cf. figure 6.1) correspond à l'utilisation d'un biais temporel. Le biais d'une unité neuronale correspond au terme supplémentaire (noté w_j dans l'équation 6.3) qu'il est possible d'ajouter à la somme pondérée.

$$y_{Nj} = f\left(w_j + \sum_{i=1}^{\text{nb(cellules de N-1)}} w_{ji} \times y_{N-1i} \right) \qquad \text{(Éq. 6.3)}$$

Le biais correspond en fait au poids synaptique d'une unité fictive dont l'activité serait constante et vaudrait 1. Ce biais permet normalement d'obtenir une meilleure résistance au "bruit" que peuvent être les données hors du corpus d'apprentissage. Les poids synaptiques sont en effet ajustés en fonction des erreurs observées en sortie du réseau à la présentation de chaque forme du corpus d'apprentissage. Une heuristique possible pour améliorer la résistance, et les capacités de

reconnaissance, du réseau face aux données hors du corpus d'apprentissage est de définir une unité fictive connectée à chaque neurone et dont la valeur vaut un. Le poids synaptique reliant cette unité fictive à chaque neurone est appelé le biais du neurone. Ce biais permet en fait de réaliser un déplacement du seuil de réponse de l'unité y_{Nj} en déplaçant la fonction sigmoïde le long de l'axe des abscisses. Ce biais w_j peut être défini comme une fonction du temps, ce qui permet de modifier la réponse de l'unité au cours du temps. Cette technique du biais temporel pourrait être vue comme un cas particulier des représentations internes explicites sur les connexions que nous venons de citer. Quelques résultats pourront être trouvés dans [horn91] ou [kim92].

La représentation interne explicite par représentation algébrique se base sur une modélisation abstraite du phénomène de décroissance de l'activité des neurones biologiques. Cette décroissance est un phénomène neurobiologique qui n'est pas pris en compte par la modélisation de McCulloch et Pitts [mcculloch43]. Cette modélisation effectue une très grande simplification de la réalité et nombre des activités chimiques et électriques ne sont pas prises en compte. Plutôt que de totalement redéfinir le neurone artificiel, la représentation algébrique de la décroissance permet de conserver le neurone de McCulloch et Pitts comme base et de modéliser le mécanisme de décroissance grâce, par exemple, à l'adjonction d'une équation ou d'un système d'équations auto-régressives. L'équation du neurone devient alors une somme de deux parties modélisant chacune un traitement différent. La première partie de la somme correspond à la somme pondérée des activités d'une partie ou de tous les autres neurones du réseau alors que la deuxième partie de la somme permet de modéliser la décroissance de l'activité du neurone considéré. L'équation résultante est de la forme de l'équation 6.4 ci-dessous.

$$y_{j,t} = \sum_{i=1}^{N} w_{ji} y_{i,t} + \sum_{j=1}^{T} a_j y_{j,t-j} \qquad \text{(Éq. 6.4)}$$

L'équation 6.4 est cependant incomplète pour obtenir la valeur d'activation exacte de l'unité $y_{j,t}$ puisqu'aucune fonction non linéaire n'y figure. Il est possible d'implanter cette fonction non linéaire de deux manières : en lui faisant englober ou non la partie auto-régressive. Ainsi, à partir de l'équation 6.4, il est possible d'obtenir deux modélisations données respectivement par l'équation 6.5 et par l'équation 6.6 : l'équation 6.5 modélise le temps de manière explicite puisque les valeurs d'autorégression sont séparées des valeurs d'entrée alors que l'équation 6.6 ne traite pas ces valeurs d'autorégression différemment des valeurs d'entrée. Ce dernier choix peut être vu comme un amoindrissement des capacités temporelles du neurone bien que cette option présente des avantages en terme de convergence et de contrôle des poids de feedback.

$$y_{j,t} = f\left(\sum_{i=1}^{N} w_{ji} y_{i,t}\right) + \sum_{j=1}^{T} a_j y_{j,t-j} \qquad \text{(Éq. 6.5)}$$

$$y_{j,t} = f\left(\sum_{i=1}^{N} w_{ji} y_{i,t} + \sum_{j=1}^{T} a_j y_{j,t-j}\right) \qquad \text{(Éq. 6.6)}$$

Enfin, alors que la représentation interne explicite par représentation algébrique se base sur une modélisation abstraite du phénomène de décroissance, il existe des modèles connexionnistes qui essaient de modéliser plus correctement les traitements chimiques et électriques effectués par les neurones biologiques. Il s'agit alors d'une représentation interne explicite du temps au niveau neuronal par un modèle à intégration et déclenchement, *integrate and fire* en anglais. Les modèles de ce type ont des définitions assez compliquées et sont généralement fondés sur un système d'équations différentielles. Quelques exemples de tels modèles peuvent être trouvés dans [abbott90] et [rinzel89]. Ces modèles, qu'il est difficile d'utiliser tels quels dans des applications d'ingénierie,

sont pourtant précieux pour les simplifications qui peuvent y être apportées en vue d'obtenir des modèles plus calculatoires. Certaines recherches ont été menées dans cette voie simplificatrice : [mcauley93].

6.1.3/ Taxonomie des unités à mémoire

Il existe une autre grande classification des réseaux par rapport à leurs capacités de mémorisation à court terme. Cette taxonomie prend en compte les facteurs de forme, de contenu et d'adaptabilité de la mémoire [mozer93]. Ici, l'architecture du neurone plus que de l'interconnexion des neurones entre eux prévaut comme critère de classification.

La forme de la mémoire correspond en fait à la seule architecture des neurones. L'architecture permet la conservation d'une activation d'une manière qui lui est propre, le traitement de l'information interne et/ou externe est donc différent selon l'architecture. La forme pourra être un délai simple (comme pour le TDNN), une mémoire à trace exponentielle, une mémoire gamma ou une mémoire gaussienne. Il existe par ailleurs d'autres formes de mémoires mais elles n'ont pas été prises en compte dans cette taxonomie. Les graphiques des réponses des architectures considérées sont donnés dans la figure 6.2 lorsqu'une impulsion unique est fournie en entrée de la ligne. Un exposé de certains de ces types de mémoires, et d'autres, est donné au paragraphe 6.4.

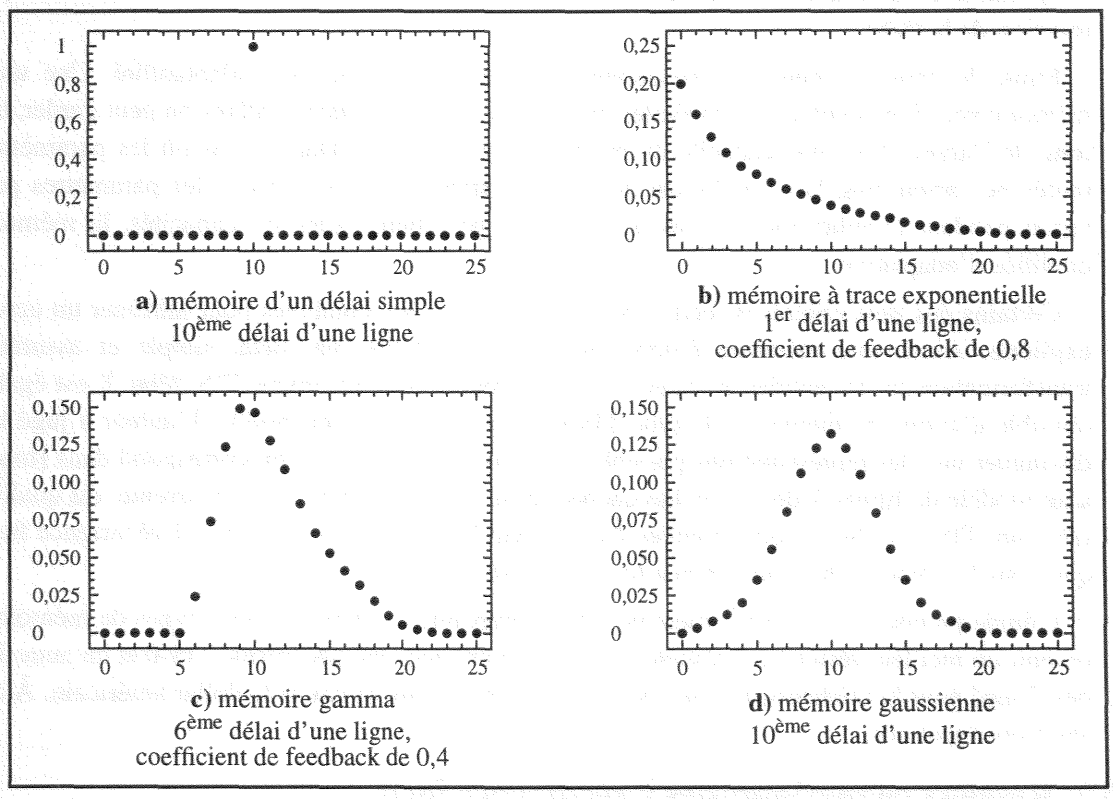


Figure 6.2 : Les réponses de différentes unités de mémorisation (d'après [mozer93]).

Ces différentes fonctions ne sont pas équivalentes dans leur mode de calcul puisque les mémoires gamma et les mémoires à trace exponentielle permettent de calculer de manière incrémentale la réponse d'une unité. La mémoire gaussienne requiert, elle, un recalcul complet de l'activité par convolution du noyau de la fonction de mémoire sur la totalité de la séquence d'entrée. C'est cependant ce désavantage qui permet à la mémoire gaussienne d'avoir une activité de mémorisation symétrique autour d'un certain point. Il est à noter que le mémoire à trace exponentielle est un cas particulier de la mémoire gamma puisque la réponse d'une unité gamma placée au début d'une ligne de délais suit le schéma type du graphe b de la figure 6.2. En effet, lorsqu'une impulsion unique est fournie en entrée d'une ligne de délais gamma, le calcul des activations des unités se trouvant du deuxième au dernier rang de la ligne prend en compte la valeur de l'unité précédente et la valeur de

l'unité courante alors que le calcul de l'activation de la première unité de la ligne ne prendra en compte, une fois l'impulsion initiale fournie, que la valeur rémanente de l'activation de cette dite unité. Ces deux modes de calcul expliquent la différence entre les graphiques b et c de la figure 6.2. De plus, l'impulsion est, au fur et à mesure de son parcours de la ligne de délais, répartie sur un nombre de délais plus ou moins important en fonction de la valeur du coefficient de feedback μ. Cette répartition explique la montée en puissance plus ou moins rapide de la valeur de l'activation d'un délai : il faudra ainsi 4 pas de temps pour qu'une unité gamma atteigne son maximum selon les conditions utilisées pour la réalisation du graphique c de la figure 6.2.

Le contenu de la mémoire est également un facteur important puisque plusieurs types de données peuvent y être stockées. Il peut s'agir d'une mémoire sur l'entrée (le sigle I, *input*, permet de la reconnaître). La mémoire peut également porter sur la sortie (qualificatif O, *output*) ou sur les états internes du réseau (qualificatif S, *state*). Les données peuvent également être modifiées par le biais d'une fonction non linéaire qui permet de ne pas conserver les données telles qu'elles ont été fournies ou telles qu'elles sont disponibles au sein du réseau. Il faut donc, dans ce cas, considérer la transformation (qualificatif T, *transformed*). Ces distinctions étant posées, il sera possible de définir de nombreux types de mémoire implantant une ou plusieurs des caractéristiques énoncées. Ainsi, il est possible de définir des mémoires I, TI, IS, TIS ou TOS, en choisissant le modèle adéquat en fonction de la tâche.

Enfin, le dernier critère pris en compte par cette taxonomie est l'adaptabilité. Une unité de mémoire peut faire évoluer ses capacités dans le temps selon certains critères ou peut garder, tout au long de l'application, une capacité de mémorisation identique. Dans le cas où les paramètres des unités ne varient pas, la mémoire est qualifiée de statique. A contrario, si les paramètres peuvent varier pendant l'application, et donc si une unité peut faire varier ses capacités, la mémoire est qualifiée d'adaptative.

Certains des trois critères de cette taxonomie peuvent être combinés pour attribuer un acronyme explicite. Ainsi, une mémoire fondée sur la représentation par délai simple et assurant une transformation sur les entrées et les états sera référencée par l'acronyme TIS-délai. Il est également possible d'avoir des mémoires de type TIS-gamma ou TOS-exponentielle. L'auteur a jugé bon de distinguer une des représentations possibles. Ce type d'architecture, qui correspond dans [mozer93] à un modèle de Elman à deux couches cachées dont seule la première est récurrente, est qualifiée de mémoire TIS-0. Celle-ci correspond en fait, de manière générale, aux réseaux à récurrence forte tels que nous les avons définis auparavant (cf. paragraphe 6.1.1).

L'étude précise qu'il est également possible d'implanter en parallèle deux types de mémoire pour obtenir un meilleur résultat et présente un réseau, combinant un mécanisme TIS-0 et un autre I-délai, développé pour la prédiction du taux de change entre le franc suisse et le dollar américain. À chacun ses préoccupations...

6.2/ Réseaux connexionnistes à récurrence forte

Une définition générale des réseaux récurrents étant donnée (cf. équation 6.2), la première architecture récurrente qui vient naturellement à l'esprit correspond à une interconnexion totale entre les cellules du réseau. Chaque cellule voit donc son activation calculée en fonction de la valeur de tous les neurones auxquels elle est connectée, étant entendu qu'elle peut théoriquement être connectée à elle-même aussi bien qu'à tous les autres neurones du réseau.

Ce schéma doit cependant être nuancé. L'architecture d'un réseau connexionniste à récurrence forte met très peu souvent en place de récurrences d'une cellule vers elle-même. Certains modèles, tel le modèle de Hopfield, peuvent aller jusqu'à interdire cette possibilité.

6.2.1/ Réseau de Hopfield

Le réseau de Hopfield [hopfield82] est un modèle inspiré d'un modèle de la physique théorique appelé modèle des verres de spins. Les spins sont des entités théoriques qui ne peuvent avoir que deux valeurs : -1 et +1. Les verres regroupent des spins et les relient les uns aux autres par des connexions valuées. La théorie a été développée à partir de la manière dont les spins interagissent les uns avec les autres et permet de comprendre comment l'énergie se stabilise au sein du verre après initialisation des spins avec des valeurs prédéfinies et après plusieurs itérations de recalcul des activités.

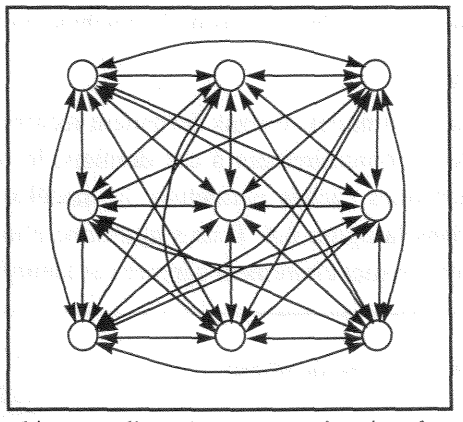


Figure 6.3 : Architecture d'un réseau connexionniste de type Hopfield.

Transposé dans le domaine du connexionnisme (voir la figure 6.3), ce modèle correspond à un réseau totalement récurrent, à l'auto-récurrence près puisque les neurones ne prennent pas leur propre valeur en compte lors du recalcul de l'activité (cf. équation 6.7). Chaque neurone du réseau est à la fois une entrée et une sortie, le réseau implantant ainsi une mémoire associative qui peut être mise en œuvre, par exemple, pour des tâches de reconnaissance de caractères [lippmann87]. Après initialisation des différents neurones aux valeurs de la donnée d'entrée, les valeurs possibles étant restreintes à 0 et 1 pour le modèle connexionniste, un processus d'itération est activé jusqu'à stabilisation des états des neurones. Ce processus itératif est stoppé lorsque la différence entre les états successifs des neurones sont minimes selon un critère qu'il faut définir. Lorsque la stabilisation est atteinte, l'énergie globale au sein du réseau est minimisée.

$$y_i = f\left(\sum_{j=1, j\neq i}^{N} w_{ji} y_j\right) \qquad \text{(Éq. 6.7)}$$

Les connexions synaptiques au sein du réseau sont symétriques et donc, si w_{ij} représente la connexion synaptique du neurone i au neurone j, l'égalité $w_{ij} = w_{ji}$ est vérifiée. Le processus d'apprentissage de ces connexions fait appel à la règle de Hebb [hebb49] qui renforce la connexion entre deux neurones s'ils sont actifs simultanément. Ce mode d'apprentissage est représenté formellement dans l'équation 6.8 où $\overline{y}_i$ représente la moyenne de l'activité du neurone i sur l'ensemble du corpus d'apprentissage. Ce processus d'apprentissage, qui correspond au calcul des moyennes d'activité des neurones, se réfère aux formes à apprendre et aucune itération n'est effectuée lors de cette phase.

$$w_{ij} = \overline{y}_i \times \overline{y}_j \qquad \text{(Éq. 6.8)}$$

La physique théorique ayant permis de définir le réseau de Hopfield, il est un des rares modèles connexionnistes à avoir des propriétés connues et démontrées. La plus intéressante de ces propriétés est la connaissance a priori du nombre de formes que le réseau pourra mémoriser. Un réseau de Hopfield de N neurones pourra ainsi mémoriser 0,14N formes distinctes sous forme de mémoire

associative.

Le réseau de Hopfield a, par ailleurs, servi de base de développement au modèle de réseau à neurone pulsé [derou94]. Ce type de réseau possède une architecture générale équivalente à la structure de réseau de Hopfield mais les neurones ont eux-même une architecture plus complexe qui utilise les paradigmes de récurrence locale et de fonction adiabatique. Nous reviendrons sur ces points ultérieurement.

Il faut enfin noter que le mécanisme de convergence du réseau ne garantit pas qu'un réseau quelconque converge, après apprentissage, vers un état stable pour toute forme qui lui sera présentée en entrée. Ce phénomène se retrouve dans le cas de la machine de Boltzmann.

6.2.2/ Machine de Boltzmann

Les machines de Boltzmann sont des réseaux fortement récurrents [hinton84], au même titre que les réseaux de Hopfield. Mais, contrairement à ces derniers, le réseau est divisé en trois parties : l'entrée, la sortie et les neurones cachés permettant la modélisation de la dynamique du réseau comme cela peut être vu dans la figure 6.4. Il ne s'agit donc plus d'une mémoire associative mais d'un réseau où entrées et sorties sont clairement séparées et identifiées.

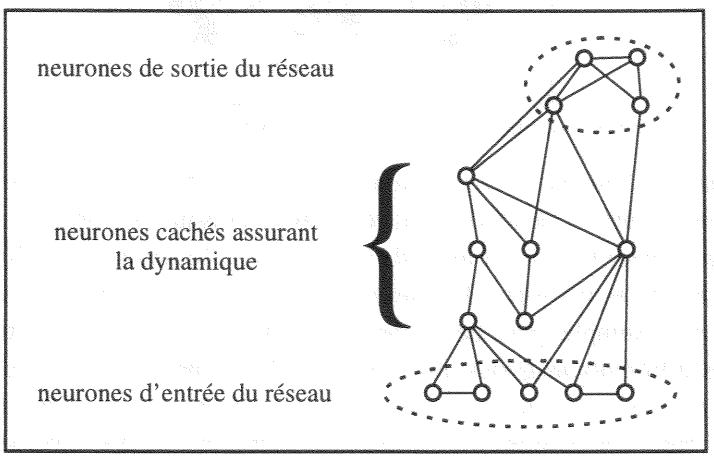


Figure 6.4 : Architecture d'un réseau connexionniste de type machine de Boltzmann.

Les machines de Boltzmann possèdent deux types de fonctionnement [azencott94] : une méthode synchrone et une méthode asynchrone (encore qualifiée de méthode séquentielle). Dans le cas d'une dynamique synchrone, les valeurs d'activation des neurones sont toutes recalculées simultanément et l'ensemble des neurones est donc remis à jour à chaque pas de temps. Ce mode de fonctionnement, identique à celui du réseau de Hopfield, peut, dans le cas d'une machine de Boltzmann, induire la non convergence du réseau avec certaines données initiales [azencott94]. La dynamique asynchrone fonctionne, elle, par mise à jour d'une sous-partie de l'ensemble des neurones du réseau. Les neurones à mettre à jour à un instant t sont tirés au sort selon une fonction plus ou moins restrictive mais assurant un parcours périodique de l'ensemble des neurones du réseau.

La stabilisation du réseau est différente selon que la dynamique est synchrone ou asynchrone. Dans le cas d'une dynamique asynchrone, le réseau doit être globalement immobile alors que dans le cas synchrone, les fréquences d'activité des neurones doivent être stabilisées.

Le calcul de l'activité d'un neurone se fait selon un schéma classique de somme pondérée, a_i étant le biais de l'unité i et f la fonction de Heaviside (cf. chapitre 2, paragraphe 2.2.1 : il s'agit de la fonction utilisée par McCulloch et Pitts). La formule est donc la suivante [azencott94] :

$$x_i = f(V_i) \qquad \text{(Éq. 6.9)}$$

$$V_i = \sum_j w_{ij} x_j - a_i$$

$$f(u) = \begin{cases} 1 \text{ si } u \geq 0 \\ 0 \text{ si } u < 0 \end{cases}$$

La fonction non linéaire de calcul de la valeur de sortie d'une unité est cependant vue par [azencott94] comme un processus stochastique :

$$P(x_i = 0) = \frac{1}{1 + \exp(V_i/T)} \qquad \text{(Éq. 6.10)}$$

Dans cette équation 6.10, T, la température du réseau, permet d'évaluer qualitativement l'agitation du réseau. Une température élevée permettra d'obtenir une quasi indépendance entre les différents neurones du réseau alors qu'une température basse permettra au réseau de se conduire de manière quasi déterministe. L'introduction de cette température abstraite permet d'obtenir de meilleurs résultats dans les phases d'apprentissage et de reconnaissance car elle évite au réseau de se retrouver bloqué dans des zones de minima locaux. Cette notion de température est inspirée des méthodes dites de recuit simulé de l'analyse numérique.

La différence entre les deux dynamiques, séquentielle et synchrone, doit également être faite au niveau de la loi d'équilibre du réseau, cette loi d'équilibre étant une fonction de l'énergie du réseau. Ce concept est très proche de celui utilisé dans les réseaux de Hopfield.

Le modèle de Boltzmann a été appliqué à de nombreux domaines de la reconnaissance de formes. Il est possible de trouver des applications en reconnaissance de la parole [prager86] ou en vision [azencott93]. Son utilisation reste cependant très limitée car ce réseau est très gourmand en puissance de calcul, tant à l'apprentissage qu'à l'utilisation, puisque le température doit être descendue progressivement pour que la convergence vers un état stable puisse être obtenue.

6.2.3/ Zipser short-term memory

Le modèle de mémoire active à court terme de Zipser a été présentée dans [zipser91]. Il s'agit d'un réseau récurrent dont l'architecture générale est assez similaire à une machine de Boltzmann. Le calcul des activations des neurones est cependant différent puisqu'il n'y a ici aucune référence au recuit simulé. Le calcul de l'activation d'un neurone fait apparaître trois valeurs qui sont spécifiques à ce modèle et qui sont référencées par x_s, x_c et $X_i(t)$ dans l'équation 6.11 où f est la fonction logistique. La variable x_s correspond à la valeur du stimulus qui est fournie en entrée de la cellule i pour le calcul de l'activation interne au pas de temps suivant. La variable x_c correspond, elle, à une valeur binaire qui permet de distinguer les phases d'apprentissage et de test. Cette variable est mise à 1 en phase d'apprentissage et à 0 en phase de test. Enfin, la variable $X_i(t)$ correspond elle à un terme de bruit gaussien. Cette dernière variable n'est active que lors des phases de tests pour simuler une activité neuronale connexe aléatoire. θ_i représente le biais de la cellule.

$$y_i(t+1) = f\left(\sum_j w_{ij} y_j(t) + w_{is} x_s + w_{ic} x_c + \theta_i + X_i(t) \right) \qquad \text{(Éq. 6.11)}$$

Ce modèle est intéressant dans les réponses qu'il est capable de fournir après apprentissage : la reconnaissance d'un stimulus connu provoquera un pic d'activité dans la cellule cible puis un maintien de l'activité pendant toute la présentation du stimulus, ce maintien précédant une décroissance de l'activité après le retrait du stimulus en entrée.

Une analyse du comportement de ce modèle de réseau et des effets provoqués par le bruit a été présentée dans [mcauley94].

6.2.4/ Réseaux duaux

Les réseaux duaux ont été présentés dans [azencott92c]. Ces réseaux sont également appelés machines de Boltzmann à cliques (figure 6.5 et figure 6.6). Ces cliques sont en fait de super unités au sein du réseau qui centralisent les activations et les décisions. La mise en place de ces cliques est une des conséquences des constatations faites par l'auteur de la lourdeur et, parfois, de la mauvaise qualité de l'apprentissage et de la convergence au sein des machines de Boltzmann.

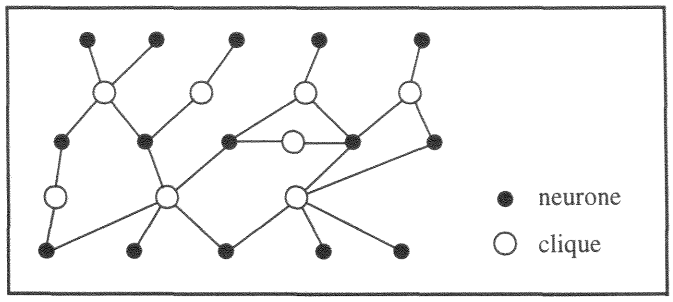


Figure 6.5 : Réseaux duaux (d'après [azencott94]).

L'introduction des cliques au sein des machines de Boltzmann permet d'introduire des contrôleurs locaux, calculant une énergie d'interaction locale. La mise en place de ces structures locales permet un meilleur apprentissage dans le mode séquentiel. Elles permettent de calculer des valeurs intermédiaires, à partir des neurones de la couche cachée, qui seront utilisées pour calculer les neurones de la couche de sortie. Par ailleurs, leur importance dans le calcul du résultat implique que les gradients d'erreur seront d'abord portés sur les cliques avant d'être répercutés sur les neurones cachés ([azencott92c] et figure 6.6) créant ainsi une structure en niveau au sein des machines de Boltzmann.

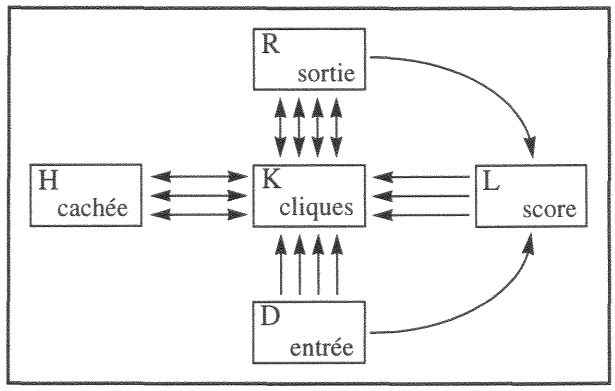


Figure 6.6 : Réseaux duaux (d'après [azencott92c]).

Le concept de machine de Boltzmann à cliques peut être retrouvé dans [saul95a] où les cliques sont qualifiées d'agrafes (*clamps*), cette notion d'agrafe ayant permis aux auteurs de faire ultérieurement le lien entre les machines de Boltzmann et les modèles de Markov [saul95b], la mise en évidence de ce lien ayant été entreprise dans [azencott92c].

Les cliques, ou agrafes, mettent cependant en œuvre des mécanismes tout à fait similaires aux neurones élémentaires eux-même et, bien qu'elles complexifient l'architecture du réseau, elles ne modifient en rien les concepts des unités de base.

6.2.5/ Modèle d'apprentissage par sélection

Certaines modifications des concepts de base des neurones apparaissent avec le modèle présenté par Dehæne et Changeux [dehæne87], [changeux89], [dehæne89]. Plusieurs idées ont été regroupées dans ce modèle pour faire émerger un modèle disposant de bonnes capacités d'apprentissage et de reconnaissance de séquences temporelles. Parmi toutes les caractéristiques regroupées ici se trouve la notion d'agrégat d'unités. Une partie des neurones du réseau sont regroupés dans différents

agrégats, chacun de ces agrégats ayant pour but de réaliser une tâche particulière. Il s'agit donc moins d'une mise en parallèle ou d'une répartition de neurones que d'un début de structuration des neurones en groupes distincts. Les neurones ne sont cependant pas tous réunis au sein de structures, baptisées ici grappes de neurones en synergie (voir la figure 6.7.a). D'autres structures élémentaires apparaissent. Le réseau comporte bien évidemment une couche d'entrée et une couche de sortie mais, en plus de celles-ci et de la couche des agrégats d'unités permettant la mémorisation, il existe une couche d'unités de codage de règles. Ces unités permettent de faire varier la prise en compte des entrées du réseau par la couche des unités de mémorisation. La prise en compte variable des entrées est assurée par le mécanisme de la triade synaptique (cf. figure 6.7.b) Ce mécanisme effectue une modulation du poids synaptique porté par une connexion de manière identique à un mécanisme dans le cerveau (cf. chapitre 2, figure 2.4). La figure 6.7.b montre ainsi une connexion allant d'un neurone A vers un neurone B dont l'efficacité est modulée par la valeur de l'unité C. Ce mécanisme, la triade synaptique, peut être vu comme une modification contextuelle, ou temporelle, des coefficients d'apprentissage.

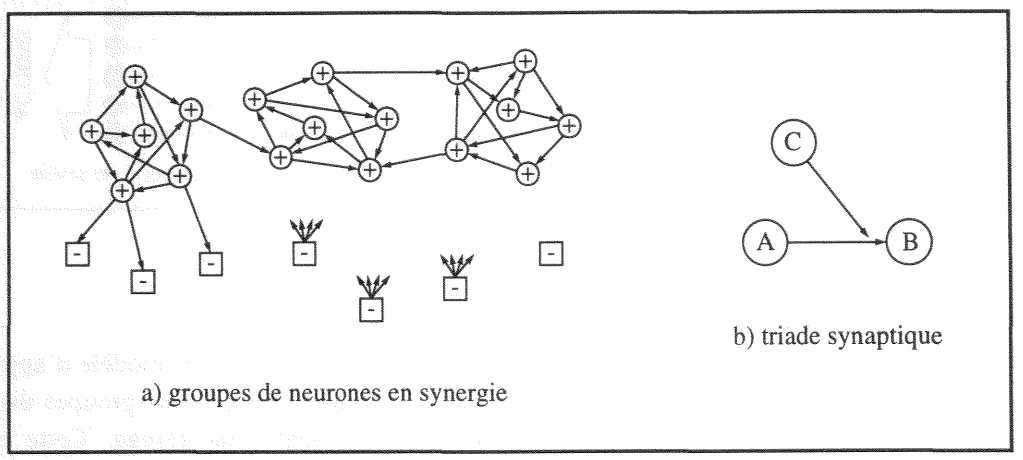


Figure 6.7 : Schémas de concepts utilisés par Dehæne et Changeux (d'après [changeux96]).

Les deux mécanismes principaux du modèle ayant été exposés, nous pouvons maintenant présenter le modèle en lui-même. Dans cette architecture, les unités de mémorisation assurent le codage des différentes séquences qui peuvent être observées en entrée. L'entrée est masquée par les unités de codage de règles qui pondèrent les valeurs des connexions allant des neurones d'entrée aux unités de mémorisation en fonction du codage interne des objets à un instant donné. Ce masquage permet de renforcer ou d'inhiber les traitements effectués dans les unités de mémorisation, les traitements étant répercutés en sortie du réseau (figure 6.8). La formalisation du processus d'apprentissage nécessaire à cette architecture pourra être trouvée dans [dehæne87]. Ce processus utilise la règle de Hebb pour diminuer ou renforcer les connexions au sein du réseau, ces connexions étant initialisées aléatoirement lors de la définition du réseau.

Le modèle d'apprentissage par sélection peut être appliqué à des tâches où se retrouve la notion d'arrivée séquentielle de l'information. Ce modèle a été principalement appliqué à une tâche de reconnaissance de chants d'oiseaux puisque l'étude neurobiologique de ce problème est à l'origine même du modèle [dehæne87]. Cette dernière tâche laisse entrevoir la possibilité d'appliquer ce modèle au domaine de la reconnaissance automatique de la parole ou à des domaines connexes mais ces études n'ont pas encore, à notre connaissance, été effectuées. D'autres tâches peuvent être résolues à l'aide de ce modèle. [dehæne96] présente ainsi des exemples d'apprentissage de règles comportementales ou d'apprentissage du test du Wisconsin sur le tri de cartes suivant des règles en constante évolution, test d'origine psychosociologique qui permet de détecter les lésions frontales. L'architecture du réseau utilisé dans ce dernier cas est cependant quelque peu différente de celle que nous avons présenté.

Ce type d'architecture fait explicitement référence au modèle de la colonne corticale [mountcastle78] ou au modèle similaire des groupe de neurones [edelman78]. La structure de colonne corticale a servi de base d'étude à des modèles connexionnistes s'en réclamant encore plus, comme nous allons le voir maintenant.

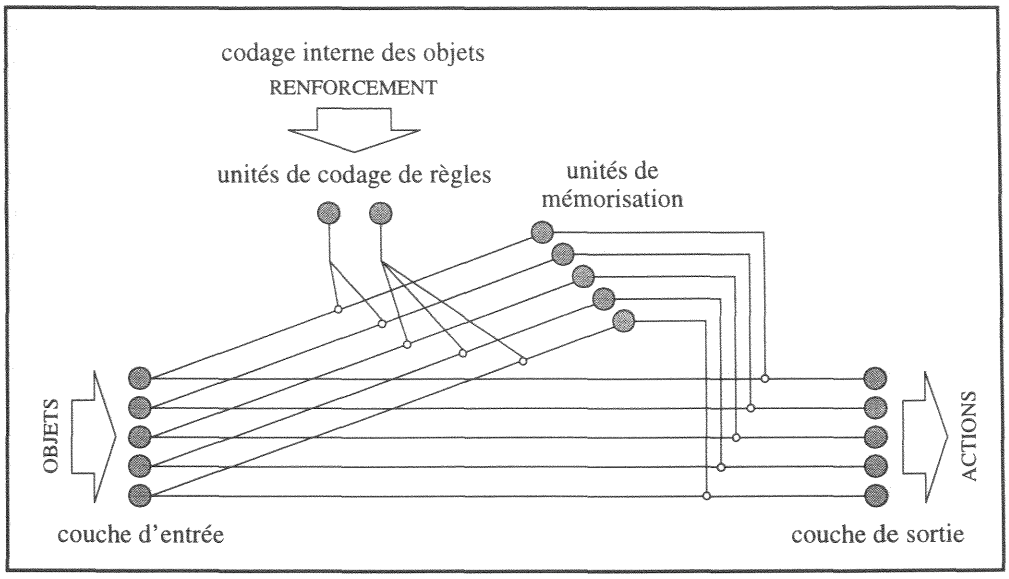


Figure 6.8 : Schéma du réseau de Dehæne et Changeux (d'après [changeux96]).

6.2.6/ Colonne corticale

On voit émerger, dans le modèle des réseaux duaux et plus encore dans le modèle d'apprentissage par sélection, une notion de segmentation de la totalité des neurones en sous-groupes distincts, les cliques synthétisant à un niveau local des activités neuronales du réseau. Cette notion de segmentation de la connaissance en petits groupes distincts de neurones peut être vue comme étant inspirée de la neurobiologie bien que les réseaux duaux aient, eux, été initialement élaborés dans un souci de stabilité de la dynamique.

Les neurones, au niveau individuel, stockent de l'information sous une forme qu'il est généralement difficile à comprendre et qui est très peu exploitable. Cette remarque est vraie quelle que soit la définition architecturale du neurone, que le champ d'étude soit la neurobiologie ou la modélisation neuromimétique. Les réseaux de neurones artificiels sont ainsi qualifiés de boites noires par certains. À l'inverse et grâce aux observations cliniques, la médecine générale puis la neurobiologie ont su distinguer dans le cerveau les différentes aires fonctionnelles et sensorielles existantes (cf. chapitre 2, paragraphe 2.2.3). Il est ainsi possible de localiser dans le cerveau les aires sensorimotrices, les aires associatives et les aires frontales.

À la croisée de ces différentes segmentations, microscopique pour le neurone et macroscopique pour les aires, il existe un niveau mésoscopique décrivant les interactions d'entités, baptisées colonnes corticales, au sein d'une même aire. Les colonnes corticales sont des regroupement de neurones concourrant à la réalisation d'une même tâche ou effectuant la reconnaissance d'un type restreint de phénomènes sensoriels. C'est à ce niveau que peut être retrouvé le principe de sonotopie pour la parole, ou de rétinotopie pour la vision qui est surtout modélisé, mathématiquement parlant, par l'intermédiaire des cartes de Kohonen [kohonen87], [kohonen88] et des modèles qui en découlent.

Le modèle de Kohonen n'est cependant pas un modèle de colonne corticale mais plutôt un modèle d'aire. La colonne corticale peut être formalisée, pour sa part, par définition d'une unité de traitement complexe, cette unité pouvant être agrégée avec d'autres pour former une aire. Nous allons maintenant décrire deux modèles de colonne corticale.

6.2.6.1/ Modèle de la colonne corticale de Burnod

Le modèle de la colonne corticale de Burnod [burnod88] définit formellement la colonne corticale. Ce modèle a été utilisé dans [alexandre90] et [guyot90] pour modéliser les aspects auditifs, visuels et moteurs du cortex.

Le modèle de la colonne corticale formalise les fonctions des différents neurones qui la composent. Ainsi, les neurones pyramidaux, en panier, en chandelier, en étoile, bipolaire et à double bouquet voient leurs fonctions regroupées et modélisées en liens lointains (liens avec des unités d'autres aires), liens locaux (liens avec des unités de la même aire), liens externes (liens avec le monde extérieur) et liens voisins (liens avec les unités voisines) [alexandre90]. Cette formalisation est décrite dans la figure 6.9 qui résume les différents types de connexions afférentes à un colonne en fonction de trois échelons formalisant, eux, les six niveaux de la colonne biologique (cf. chapitre 2, figure 2.11).

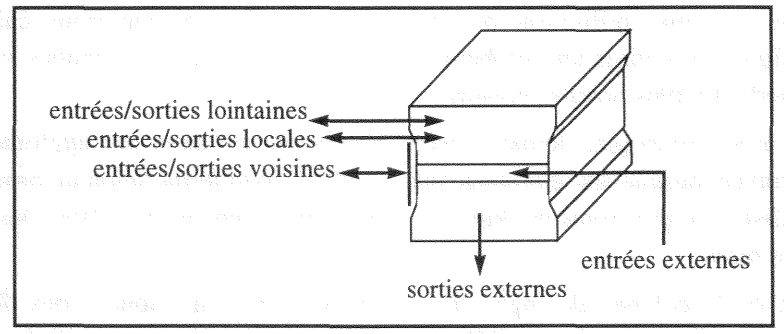


Figure 6.9 : Schématisation d'une colonne corticale (d'après [alexandre90]).

L'apprentissage dans ce modèle de la colonne permet de définir les valeurs des liens et, donc, du comportement de la colonne vis-à-vis de son entourage proche ou éloigné [alexandre90].

Ces colonnes se regroupent en aires. À l'intérieur de ces aires, chaque colonne effectue un traitement particulier qui peut correspondre, dans une tâche de la classification, à la reconnaissance d'un type particulier de formes d'entrée. Ce regroupement de colonnes est présenté en figure 6.10.

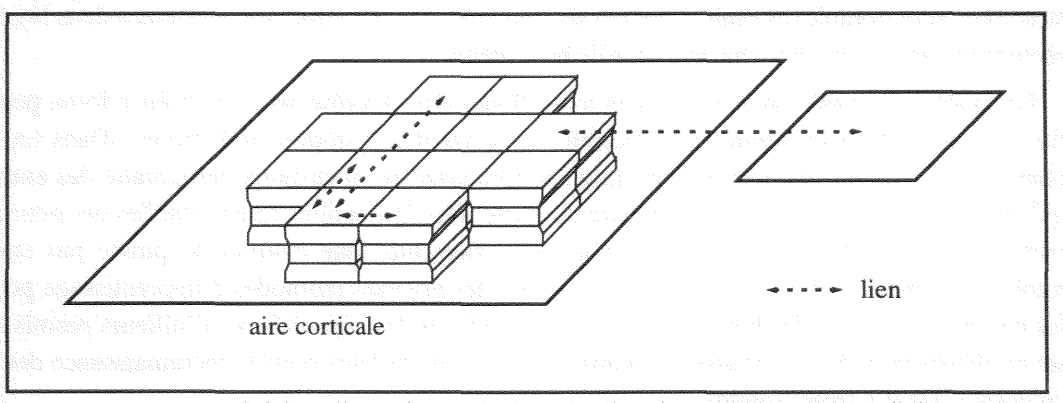


Figure 6.10 : Schéma d'une aire corticale et des liens internes et externes à l'aire (d'après [alexandre90]).

Des résultats présentés dans [alexandre90] permettent de constater que chaque colonne code un masque, correspondant à une forme ou à une partie d'une forme. Ces résultats peuvent être jugés meilleurs que ceux obtenus avec un modèle de Kohonen où la connaissance n'est pas représentée de manière hiérarchisée. Dans ce dernier type de modèle, le manque de hiérarchie et l'apprentissage non supervisé peuvent, en outre, interdire une exploitation correcte de la connaissance finalement représentée de manière parfois anarchique [buniet91].

D'autres modèles de colonnes corticales existent. Au rang de ceux-ci se trouve le modèle formel proposé par Ingber.

6.2.6.2/ Statistical Mechanics for Neocortical Interactions

Le modèle SMNI, acronyme de *Statistical Mechanics for Neocortical Interactions*, a pour la première fois été présenté dans [ingber81]. Il schématise également les liens entre différentes colonnes corticales d'aires du cerveau mais se veut être d'une approche beaucoup plus expérimentale puisqu'il a été établi à l'origine pour modéliser les phénomènes corticaux observables par électro-encéphalogramme (EEG). Ce modèle fait un usage très large de notions tirées de la mécanique statistique [ingber82].

Ce modèle formalise les interactions se déroulant dans le cerveau sur trois niveaux. Un premier niveau, le niveau microscopique, modélise ainsi le neurone de manière statistique en dégageant les probabilités de décharge des neurones, ceux-ci dépendant de processus gaussiens ou de processus de Poisson lors de la phase de réception de l'information.

Un deuxième niveau, les domaines mésoscopiques, formalisent les colonnes corticales. Ce niveau se caractérise par une utilisation de probabilités de bas niveau pour calculer une activité "mésocolonnique" et définir une probabilité d'interaction avec les colonnes voisines considérées selon la méthode des plus proches voisins.

Le niveau macroscopique, dernier niveau, représente les aires par agglomération de 10 à 30 colonnes au travers desquelles se propage une activité à court terme selon un processus markovien et donc probabiliste. Ce processus markovien est également supposé modéliser les interactions à long terme dans les aires.

Le modèle de la colonne de Ingber a été principalement appliqué à des tâches de simulation d'électro-encéphalogrammes [ingber95] mais reste cependant très peu utilisé par ailleurs bien que les résultats paraissent intéressants.

6.3/ Réseaux connexionnistes à récurrence par plaque

Les réseaux à récurrence par plaque se distinguent des réseaux à récurrence forte par la mise en place au sein du réseau de structures de grande taille, ces structures étant composées de plusieurs neurones. Mais, à la différence des réseaux duaux ou des modèles de colonnes corticales où les structures sont complexes mais n'ont pas de rôle prédéfini, les structures présentes dans les réseaux à récurrence par plaque ont chacune un rôle bien précis.

Les notions d'entrée ou de sortie sont assez floues dans les réseaux à récurrence forte, peut-être du fait de leur forte inspiration, ou proximité, vis-à-vis des systèmes biologiques. Dans un réseau à plaques, il existe plusieurs structures dont une sera affectée au stockage temporaire des entrées alors qu'une autre, totalement distincte, servira de sortie. Ces distinctions fonctionnelles ont pour avantage particulier de simplifier le processus d'apprentissage bien que celui-ci ne puisse pas encore être implanté de manière triviale. Certaines études sur les diverses méthodes d'apprentissage possibles et les optimisations, voir les heuristiques, qui en étendaient les capacités ont d'ailleurs permis d'obtenir les meilleurs taux de reconnaissance jamais obtenus par ailleurs pour la reconnaissance des voyelles du corpus TIMIT [robinson94].

Une première définition des structures exposées ci-après venant d'être donnée, il est également intéressant de signaler quelques extensions qui y ont été faites. Ce type d'architecture a en effet été largement étudié pour l'apprentissage et la modélisation de séquences abstraites (paragraphe 6.3.6). Ces études utilisent généralement des réseaux à plaque mais de récentes études utilisent des architectures qui, bien que différentes, s'en sont inspirées (voir, par exemple, le paragraphe 6.3.7 ou le paragraphe 6.3.9). Les architectures présentées dans ces derniers paragraphes nous permettrons, en outre, de faire le lien avec le paragraphe 6.4.

6.3.1/ Modèle de Jordan

Le premier modèle des réseaux à récurrence par plaque est aussi le plus simple. Il est également

l'initiateur de l'idée qui a conduit au développement de tous les autres. L'origine des réseaux de ce type peut être trouvée dans [jordan86]. Ces réseaux sont en fait des perceptrons multicouches qui ont une vue sur le contexte antérieur immédiat. Ce contexte, qui correspond à la mémoire des décisions antérieures du réseau, est fourni par simple recopie de la couche de sortie dans une partie de la couche d'entrée (figure 6.11).

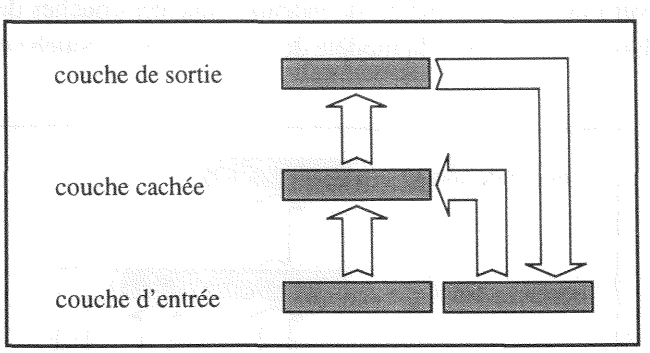


Figure 6.11 : Schéma de principe du modèle de Jordan.

La recopie simple de la couche de sortie en entrée du réseau n'était cependant pas suffisante pour l'objectif fixé dans [jordan86]. La tâche étudiée, la génération de plans en séquence, a été justifiée suffisamment complexe pour que l'auteur adjoigne à son réseau un mécanisme supplémentaire de recopie de la couche d'entrée vers elle-même (figure 6.12). Cette partie de la couche d'entrée en charge de la récurrence assure donc la véritable dynamique du réseau, la partie des neurones servant d'entrée au réseau étant positionnée à une valeur qui ne sera pas modifiée pendant le processus de génération de plan.

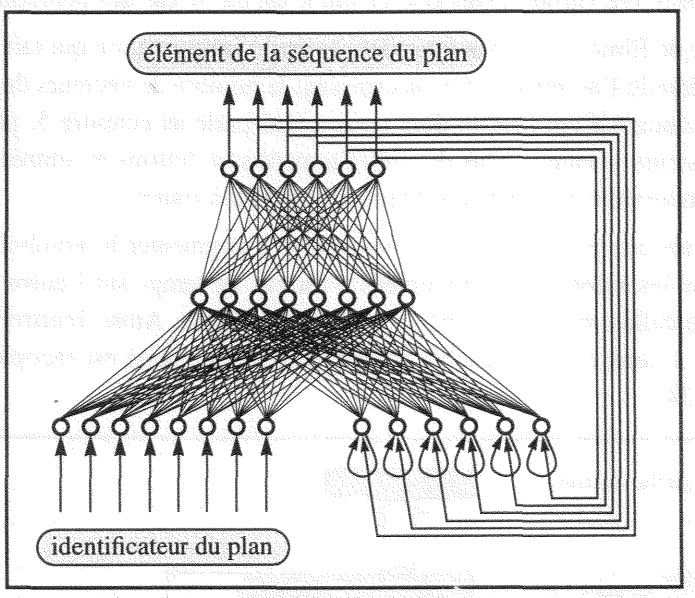


Figure 6.12 : Schéma calculatoire d'un réseau de Jordan (d'après [jordan86]).

Le réseau de Jordan a ainsi été appliqué à une tâche de production de la parole [lippmann89] mais, d'un point de vue psychologique, le modèle de Jordan peut être considéré comme une confirmation possible au problème général de l'associationnisme de Lashley [lashley51]. Lashley postulait que les mécanismes d'association simple ne permettaient pas de traiter et/ou de produire des structures séquentielles puisqu'ils leur étaient impossible de fournir un contexte, le contexte étant nécessaire à un séquencement correct de tâches comportementales ou à une analyse de chaînes grammaticales. Le modèle de Jordan confirme donc, à sa manière, l'hypothèse de Lashley en fournissant une structure mathématique apte à produire un plan sous la forme d'une séquence abstraite.

Le modèle de Jordan n'est cependant pas la seule architecture apte à produire des plans d'actions.

D'autres recherches, [thrun91], ont été menées avec succès sur ce problème en utilisant un modèle connexionniste différent : le modèle de Elman.

6.3.2/ Modèle de Elman

Le modèle de Elman [elman90] est postérieur au modèle de Jordan et y prend l'idée d'une recopie, d'un pas de temps sur l'autre, d'un vecteur de valeurs d'une des couches de niveau supérieur vers la couche d'entrée. Mais à la différence du modèle de Jordan, c'est la couche cachée qui est recopiée en entrée du réseau.

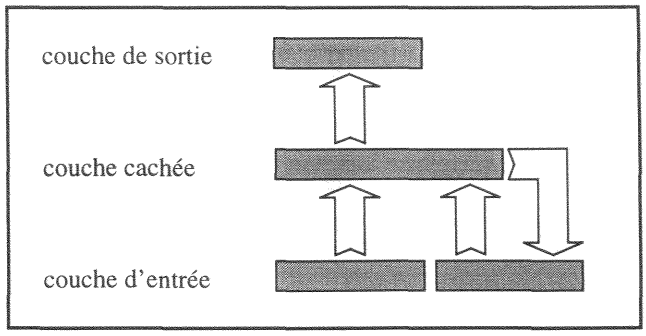


Figure 6.13 : Schéma général du modèle de Elman (d'après [elman90]).

Le modèle de Elman a été appliqué avec succès à des tâches de reconnaissance de la parole, prouvant, à chaque fois que la comparaison était faite, sa supériorité par rapport aux perceptrons multicouches [kumar91], [cottrell91]. L'architecture de Elman a par exemple prouvé qu'elle permettait d'obtenir de meilleurs résultats qu'un perceptron sur des tâches de production de phonèmes dont la durée variait fortement, ce qui n'est pas le cas des perceptrons [sejnowski87].

L'architecture de Elman n'est pas figée dans la seule configuration qui fait l'objet de la figure 6.13. [cho90] a ainsi étendu l'architecture en accroissant le nombre de vecteurs de données d'entrée ce qui correspond au principe d'agrégation dont nous avons parlé au chapitre 5, paragraphe 5.4.2.2. Selon ce principe, plusieurs vecteurs issus du prétraitement sont fournis en entrée du réseau, ces vecteurs étant calculés à intervalle de temps constant sur le signal à traiter.

D'autres auteurs ont par ailleurs jugé nécessaire d'augmenter le nombre de couches cachées de manière à ce que les informations recopiées d'un pas de temps sur l'autre soit d'un niveau encore plus abstrait que celles recopiées dans le modèle de Elman. Ainsi, [cottrell91] propose de réaliser une architecture à deux couches cachées dont seule la deuxième est recopié d'un pas de temps sur l'autre (figure 6.14).

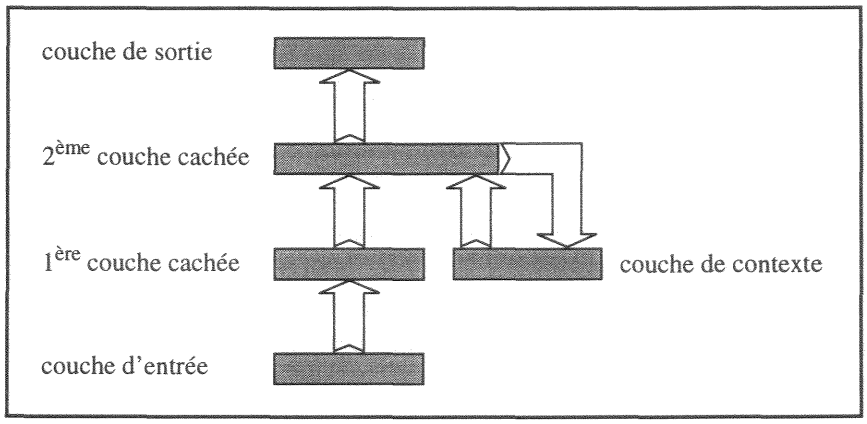


Figure 6.14 : Un développement du modèle de Elman effectué dans [cottrell91] pour une tâche de synthèse de parole à partir de mots.

Les modèles de Jordan et de Elman et les deux principes qui y sont associés étant présentés, nous allons maintenant voir des modèles qui en reprennent les idées à des degrés divers, en y ajoutant

parfois des principes d'apprentissage qui n'étaient pas présent dans les modèles originaux.

6.3.3/ Le réseau à information par état simple

Le réseau à information par état simple, *Simple State Information Network*, est un réseau associant les deux types de récurrence que nous venons de voir : le modèle de Jordan et le modèle de Elman. Il regroupe le principe de recopie de la couche de sortie en couche d'entrée avec le principe de recopie de la couche cachée en couche d'entrée. L'architecture générale est ainsi équivalente à celle présentée dans la figure 6.15. La recopie de la couche de sortie assure ici la mémorisation de la dernière prédiction faite alors que la recopie de la couche cachée permet de conserver la projection des dernières caractéristiques de l'environnement.

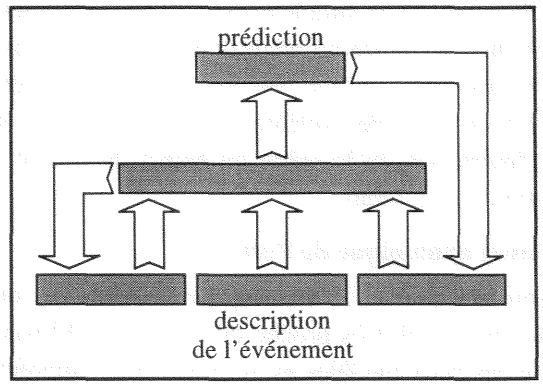


Figure 6.15 : Le modèle du Simple State Information in a Recurrent Network (d'après [hanson96]).

Ce modèle a été présenté dans [hanson96] pour tenter de modéliser le cycle perceptif de Neisser dont nous avons déjà parlé (cf. chapitre 5, paragraphe 5.2.3) sur deux expériences restreintes mais de haut niveau. Les auteurs de l'article présentent ainsi une vision des réseaux de neurones (figure 6.16) qui se veut la plus proche possible du cycle perceptif de Neisser (cf. chapitre 5, figure 5.6) en réinterprétant les fonctions des différentes composantes du réseau.

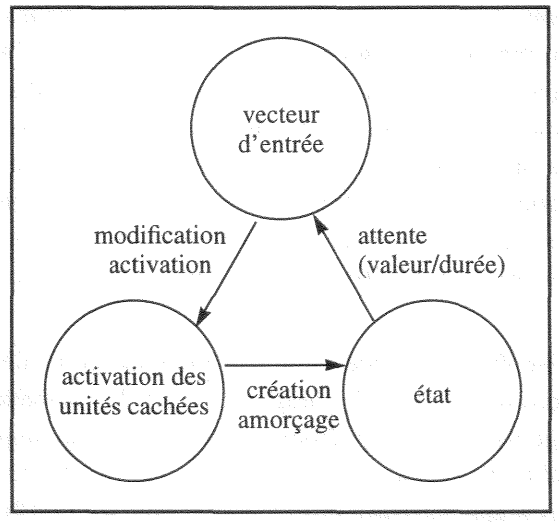


Figure 6.16 : Réinterprétation d'un réseau de neurone récurrent en fonction du cycle perceptif de Neisser (d'après [hanson96]).

Les expériences menées ont tout d'abord obligé les auteurs à décoder des bandes vidéos de comportements humains assez habituels : la première expérience portait sur une personne buvant un café au restaurant tout en lisant son journal alors que la deuxième expérience permettait de suivre deux personnes jouant une partie de Monopoly. Ces séquences vidéos ont été encodées par analyse de la scène toutes les secondes. L'analyse de la scène a permis d'isoler différentes actions de haut

niveau qui ont permis, une fois le corpus établi, d'établir la forme du vecteur d'entrée décrivant l'événement. Une partie de ce corpus a servi à l'apprentissage alors que l'autre servait à valider le réseau, cette validation étant qualifiée ici de vérification de la qualité du transfert puisqu'il s'agit de tâches comportementales de haut niveau.

Bien que les expériences utilisées pour valider l'architecture du *Simple State Information Network* n'aient que peu de rapport avec des tâches de perception auditive, la qualité de la prédiction obtenue est bonne, y compris lorsque la qualité du transfert d'une tâche est vérifiée sur l'autre tâche bien que les résultats soient évidemment moins bons.

Le point intéressant de ces expériences de transfert est la segmentation à intervalle constant du corpus, un événement n'étant pas ici un simple élément d'une grammaire de comportements de base. Les différents comportements ont, grâce à l'analyse des images toutes les secondes, une durée plus ou moins longue selon, par exemple, qu'une personne tourne une page d'un journal ou lit un article de ce même journal. Les durées des différents comportements semblent donc correctement modélisées bien que la description de la scène en entrée du réseau soit très probablement d'une grande aide pour l'obtention des résultats.

6.3.4/ Réseau à propagation dynamique de l'erreur

Le modèle de Robinson, le *Dynamic Error Propagation Network* ou *Error Propagation Network* [robinson89], présenté ici est en fait très proche du modèle de Elman (paragraphe 6.3.2). La seule véritable distinction entre ces deux modèles est le processus d'apprentissage qui est finalement très important au niveau de l'exploitation puisqu'il détermine les capacités du réseau.

Le principal domaine d'application du modèle de *DEPN* est la reconnaissance de la parole et il a permis à son auteur d'obtenir de très bons résultats puisque ce type de réseau parvient à obtenir des capacités de reconnaissance de l'ordre de 80% pour des tâches de reconnaissance de phonèmes sur le corpus TIMIT [robinson94].

Comme pour le modèle de Elman, ce réseau permet d'avoir un regard sur le passé grâce à la recopie des neurones de la couche cachée. Mais à la différence du modèle de Elman, la recopie est ici partielle comme le montre la figure 6.17. Les neurones de la couche cachée sont en fait divisés en deux sous-groupes. Un premier sous-groupe fournira les valeurs d'activation qui seront utilisées pour calculer les activations des neurones de la couche de sortie tandis que le deuxième sous-groupe ne fournira son vecteur d'activation qu'au seul mécanisme de recopie. Il s'effectue donc, au sein de la couche cachée, une spécialisation des unités en fonction du mécanisme de la recopie. Les capacités de représentation des informations antérieures ne sont cependant pas amoindries par ce mécanisme de recopie partielle puisque le nombre d'unités de la partie de la couche cachée attribuée au calcul des unités de sortie peut être augmenté pour répondre aux besoins de représentation [robinson91].

L'apprentissage utilisé dans ce modèle est une méthode équivalente à la méthode RTRL sur laquelle nous reviendrons (cf. chapitre 7, paragraphe 7.2.5.2). La méthode RTRL [williams89] et la méthode d'apprentissage dans les *DEPN* [robinson89] sont en fait apparues simultanément et seule la plus grande généralité théorique de la RTRL permet de la considérer comme la méthode principale. Le processus d'apprentissage mis en œuvre avec les DEPN est cependant beaucoup plus efficace aujourd'hui que le RTRL puisque qu'il a été étudié pendant de nombreuses années par le concepteur. Il est ainsi intéressant de voir que, sans aucune modification architecturale, l'auteur a su faire passer le taux de reconnaissance des voyelles de TIMIT de 60 ([fallside90], [robinson90a], [robinson90b]) à 80 pour cent ([robinson94]).

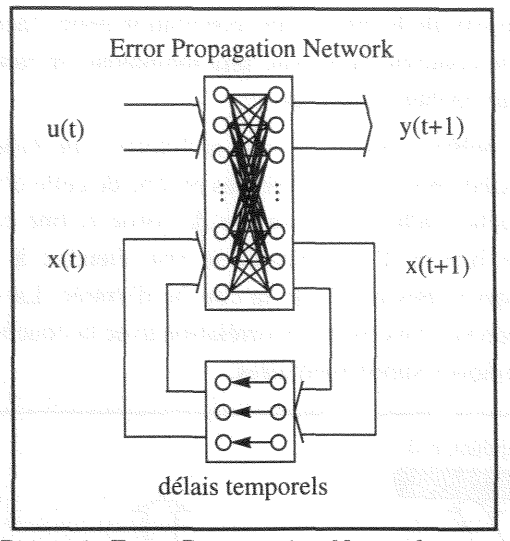


Figure 6.17 : Le *Dynamic Error Propagation Network* (d'après [robinson92]).

Le modèle de Robinson a fait l'objet de développements intéressants en adaptant le réseau à des techniques hors du domaine connexionniste. Il a ainsi été mis en œuvre avec des réseaux de Markov cachés [robinson92], à la manière de ce qui a été fait par ailleurs avec des perceptrons multicouches [bourlard90]. Les résultats obtenus sont cependant moins bons que d'autres obtenus par ailleurs [bourlard93] à la même époque.

Le modèle de Robinson a également été adapté à l'apprentissage par renforcement [kokar93], [jordan94a], sous le nom de *Reinforcement Driven Dynamic Network*. Ce type d'apprentissage est idéal pour résoudre des tâches pour lesquelles il est difficile de définir un vecteur de sorties désirées et où un apprentissage non supervisé est inapplicable. Ce type d'apprentissage est présenté dans [robinson89] pour une application au jeu du morpion, jeu se déroulant sur une matrice 3×3 où deux joueurs s'affrontent en inscrivant des ronds et des croix. La phase d'apprentissage a permis au réseau de passer de 30% de parties gagnées à 59%, l'opposant étant un algorithme ad-hoc basé sur un générateur de nombres aléatoires. Ce type de tâches peut s'apparenter aux tâches de reconnaissance de séquences définies à partir d'une grammaire.

6.3.5/ Réseaux récurrents hebbiens

L'architecture des réseaux récurrents hebbiens est également inspirée de l'architecture du modèle de Elman. Mais les différences architecturales permettent à ce modèle, selon son auteur [dennis94], d'implanter une mémoire à plus long terme que la mémoire effectivement implantée par les réseaux récurrents simples. De plus, à l'image des processus humains de mémorisation, ce modèle doit permettre :

- de créer des représentations sensibles aux caractéristiques de l'environnement,
- d'acquérir des critères de décision,
- d'acquérir des mécanismes de contrôle,
- d'éviter les problèmes d'interférence et de limitation des capacités de représentation,
- de généraliser des listes de longueur arbitraire définies sur de grands vocabulaires,
- de former rapidement des liaisons entre des représentations connues.

L'auteur estime que les modèles connexionnistes standards utilisant la rétropropagation du gradient d'erreur, dont le modèle de Elman, possèdent d'excellentes qualités d'interactivité à l'utilisation mais ne possèdent pas les caractéristiques de mémorisation qui viennent d'être énoncées. Il a donc étendu le modèle de Elman en lui ajoutant un mécanisme de mémorisation qui ne fasse plus seulement appel à la procédure de rétropropagation du gradient d'erreur mais qui fait également appel au mécanisme d'apprentissage hebbien qui permet, de manière habituelle, de créer, ou de

constater dans le cas de réseau de Hopfield, une corrélation entre l'activité de deux neurones. Ainsi, l'activité dans la plaque de contexte du réseau sera maintenue ou renforcée par une autocorrélation hebbienne depuis la couche cachée du réseau.

Les réseaux récurrents hebbien possèdent une architecture du type de celle schématisée dans la figure 6.18. Le réseau possède une architecture très proche de celle d'un modèle de Elman, avec une couche d'entrée, une couche cachée, une couche de sortie et une couche de contexte. Mais, à la différence du modèle de Elman, l'architecture se voit étendue à sa sortie par une couche qui correspond à la recopie stricte des unités de la couche d'entrée. La couche de contexte n'est, elle, plus définie par simple recopie mais par autocorrélation avec la couche cachée, le réseau utilise aussi un vecteur de poids synaptiques supplémentaires.

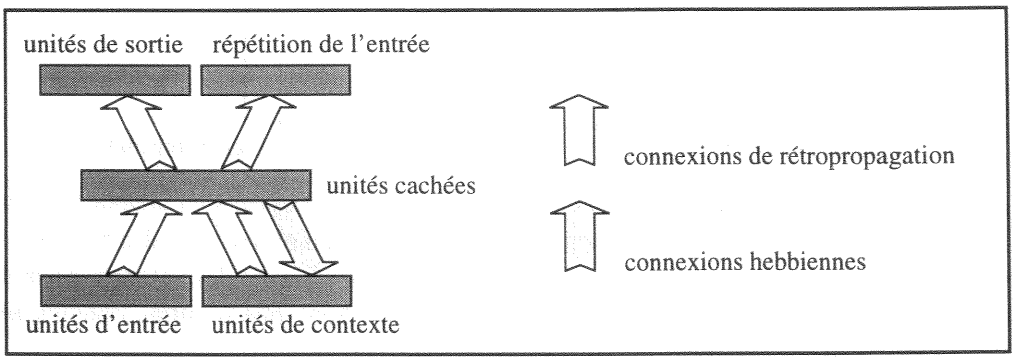


Figure 6.18 : Réseau récurrent hebbien. Schéma architectural (d'après [dennis94]).

L'adjonction d'une "seconde" couche de sortie, permettant d'obtenir en sortie l'image exacte de la couche d'entrée, permet en fait de définir, par apprentissage, un mécanisme de compression de l'entrée. La compression obtenue est identique à celles qui ont été réalisées dans les premiers temps du connexionnisme avec des réseaux *m-n-m*, *m* étant la taille d'un mot du vocabulaire et *n* étant la taille supposée d'une hypothétique ligne de communication, *n* étant plus petit que *m* et le plus proche possible de 1. Grâce à cette répétition de l'entrée en sortie, le réseau est obligé d'assurer une compression des données fournies en entrée avant de les restituer sur la couche de sortie dédiée, cette recopie de l'entrée en sortie n'étant pas la seule tâche assurée par le réseau qui doit également résoudre un problème vis-à-vis des unités de la couche de sortie effective. Cette compression permet de minimiser l'espace d'état occupé par le mot d'entrée courant, cette compression pouvant prendre en compte la taille du vocabulaire mais également la fréquence des mots d'entrée pour en assurer la représentation la plus compacte possible.

Les unités de contexte voient leurs valeurs calculées non par simple recopie mais par autocorrélation hebbienne. Cette recopie permet de moduler l'importance de certaines unités de la couche cachée et de ne prendre en compte que celles qui sont importantes pour l'application. Le processus de mise à jour des valeurs des unités de la couche de contexte respecte l'équation 6.12 suivante :

$$A_C(t+1) = f((A_H(t) \times W_{HC}(t))/\gamma) \qquad \text{(Éq. 6.12)}$$

Dans cette équation, A_C correspond au vecteur des activations de la couche de contexte, A_H correspond au vecteur des activations de la couche cachée et W_{HC} correspond aux valeurs des connexions synaptiques, hebbiennes dans ce cas, entre la couche cachée et la couche de contexte. La fonction sigmoïde f est définie selon l'équation 6.13 :

$$f(x) = \frac{2}{1+e^{-x}} - 1 \qquad \text{(Éq. 6.13)}$$

Un mécanisme supplémentaire est mis en œuvre dans la matrice des poids pour éviter la saturation

des unités de contexte. Ce mécanisme modifie les poids hebbiens tout au long du traitement d'une séquence d'entrée. Les poids W_{HC} sont ainsi recalculés à chaque pas de temps selon l'équation 6.14 où λ correspond au facteur de décroissance de la mémoire.

$$W_{HC}(t+1) = (1-\lambda) W_{HC}(t) + \lambda A_H(t) A_H^T(t) \qquad \text{(Éq. 6.14)}$$

Il est à noter que le vecteur de contexte et le vecteur des poids synaptiques de la couche cachée vers la couche de contexte sont remis à zéro avant chaque traitement de séquences. Ainsi $A_C(t = 0) = 0$ et $W_{HC}(t = 0) = 0$. Le schéma de principe d'un réseau récurrent hebbien est présenté dans la figure 6.19.

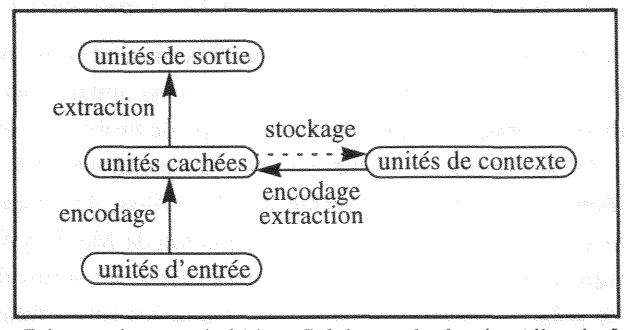


Figure 6.19 : Réseau récurrent hebbien. Schéma calculatoire (d'après [dennis94]).

Dans ce type de réseaux, le paramètre λ de décroissance de la mémoire n'est pas déterminé par apprentissage mais défini par le concepteur. L'étude menée dans [dennis94] sur la meilleure valeur possible de λ par rapport à une tâche de reconnaissance d'un élément d'une liste a permis de montrer que des valeurs basses, entre 0,2 et 0,4, étaient préférables à des valeurs plus importantes. Cette détermination est cependant remise en cause par d'autres études menées par l'auteur dans son mémoire. D'une manière générale, ce réseau a été appliqué aux seules tâches de reconnaissance de chaînes et de mots en fonction de leur fréquence dans un corpus, ce type de tâche étant également connu sous le terme de reconnaissance épisodique. Nous ne connaissons aucune application de ce réseau à des tâches de reconnaissance sensorielle de bas niveau.

6.3.6/ Encodage d'automates à états finis

6.3.6.1/ Présentation du problème

Nous avons, dans les paragraphes précédents, présenté quelques unes des applications possibles des réseaux connexionnistes à récurrence par plaques. Au rang de celles-ci se trouvent les applications de reconnaissance de la parole pour lesquelles ce type de réseaux a montré de bonnes capacités [robinson94]. Mais ces réseaux sont particulièrement bien adaptés à un domaine qui semble en totale adéquation avec les possibilités que procurent ce type d'architecture : la reconnaissance de séquences abstraites.

À la différence d'autres tâches où les séquencements ne sont pas toujours visibles de prime abord, la reconnaissance de séquences abstraites permet de se concentrer sur les capacités des modèles étudiés à résoudre des problèmes de reconnaissance d'ordonnancement. L'ordre est sous-jacent dans de nombreux problèmes de recherche, la reconnaissance de la parole étant au rang de ceux-ci. Et l'ordonnancement de formes abstraites conduit très naturellement à la notion plus générale de codage du temps. Certains problèmes de la RAP font explicitement appel à des notions d'ordre, tout particulièrement pour les problèmes de coarticulation. Ainsi, un phonème *P1* prononcé après un autre phonème *P2* pourra avoir un spectre assez différent de celui qu'il aurait eu s'il avait été prononcé après un phonème *P3* (cf. chapitre 1, paragraphe 1.6.2). Le phénomène de la coarticulation peut même avoir des incidences sur une étendue plus importante que deux phonèmes simplement contigus comme, par exemple, pour l'effet "carabine" où la liberté des articulateurs permet une anticipation et donc une déformation des formants des voyelles [pierrel91]. Ce problème de la

coarticulation peut entraîner des erreurs de classification des phonèmes lorsque les spectres sont classés dans l'absolu sans tenir compte du contexte et impose alors, pour améliorer les résultats, de mettre en place une grammaire permettant de connaître les réécritures possibles de certains phonèmes en d'autres dans une étape ultérieure à la classification.

Pour totalement isoler le problème de l'ordonnancement, certaines recherches ont été conduites sur des problèmes de reconnaissance de séquences ou de séries linéaires, permettant aux modèles étudiés de totalement s'abstraire de problèmes, annexes dans ce cas, tels que la classification de spectres. Ces recherches ont permis de tester les capacités de nombreux modèles connexionnistes. Les faibles capacités de certaines architectures et des algorithmes d'apprentissage associés ont également été démontrées grâce aux études expérimentales menées dans le domaine.

Le problème de la reconnaissance de séquences abstraites ne doit pas être confondu avec le problème de la prédiction. Il est possible de considérer la reconnaissance de séquences comme un cas dégénéré de prédiction mais, alors que la premier type de problèmes impose presque toujours de définir un automate, la prédiction ne l'impose pas. La prédiction est une tâche distincte de la reconnaissance de séquences abstraites puisque qu'elle utilise des données réelles, et donc bien souvent bruitée, telles que les séries produites par l'équation de MacKey-Glass [mackey77] qui a un comportement chaotique avec certaines valeurs de paramètres, les séries de coordonnées de courbes sinusoïdales, l'activité solaire, la consommation électrique, les séries financières et d'autres encore. Cette utilisation de données réelles, ou réalistes, implique immanquablement que la prédiction ne sera pas parfaite et qu'une marge d'erreur existera, quelque soit la qualité du modèle de prédiction. Une tâche de reconnaissance de séquences abstraites imposera, par contre, que l'automate soit parfaitement défini par le système chargé de l'implanter, le problème majeur étant, dans ce cas, que le passage d'un état à un autre ne soit pas dans un domaine trop flou.

Les problèmes de reconnaissance de séquences imposent, d'une manière générale, de définir un automate d'états fini de manière à vérifier l'exactitude de la séquence analysée par rapport aux séquences de référence. Cet automate permet de garder une mémoire, parfois approximative selon les grammaires, des événements antérieurs par la simple connaissance de l'automate et de l'état courant. Ce couple de données permet donc de réaliser une synthèse de l'information dans le plan temporel grâce, d'une part, à l'automate défini de manière fixe par l'intermédiaire des poids connexionnistes et, d'autre part, à l'état variable des unités connexionnistes. Cette synthèse de l'information permet d'éviter l'emploi d'une ligne de délais dont la profondeur devrait être égale à la longueur de la séquence à analyser. Ce dernier type de problème peut se retrouver à un niveau moindre dans les tâches de classification comme en RAP avec, par exemple, la définition du nombre correct de délais dans les modèle de type TDNN [waibel89], [bodenhausen91a] vis-à-vis de la tâche, cette adaptation permettant d'obtenir de meilleurs résultats puisqu'elle permet de disposer d'informations contextuelles de meilleure qualité [hermansky95b].

Des expériences ont par ailleurs été menées sur le problème dual qu'est la génération de séquences. Ces expériences ont déjà été mentionnées dans le paragraphe 6.3.1 et concernent le modèle de Jordan [jordan86]. Elles ont permis de vérifier la validité des architectures connexionnistes à récurrence par plaques pour ce type de problème et il semblait, dès le début des recherches dans le domaine de la reconnaissance de séquences, qu'une architecture duale à celle du modèle de Jordan permettrait de résoudre les problèmes de reconnaissance. Le modèle dual de celui de Jordan est le modèle de Elman dont nous avons parlé précédemment (paragraphe 6.3.2) mais d'autres modèles ont été utilisés pour tenter de découvrir la manière la plus efficace de coder un espace d'états en utilisant une architecture connexionniste.

6.3.6.2/ Encodage d'automates

L'encodage d'automates à l'aide de réseaux de neurones peut s'effectuer de différentes manières. La gestion d'un automate imposant d'avoir accès à l'entrée courante du système et à l'état courant de

l'automate, l'architecture qui semble naturellement adaptée comporte un rebouclage de la couche des états vers l'entrée du réseau, entrée qui se voit par ailleurs adjoindre une partie correspondant au codage des différents terminaux du langage. Nous avons schématisé cette architecture dans la figure 6.20. Mais cette architecture n'est évidemment pas la seule possible.

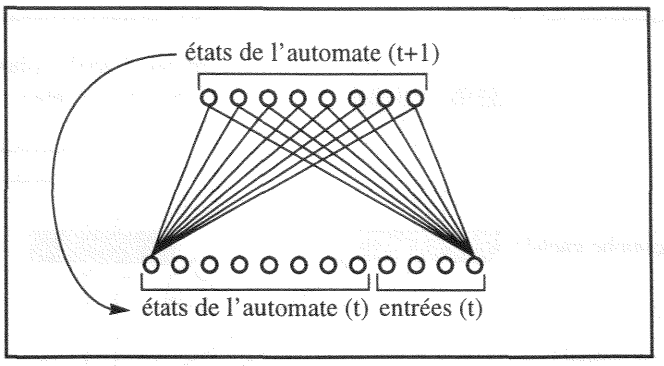


Figure 6.20 : Schéma général d'un réseau apte à l'encodage d'automates d'états fini.

Une première modification qui peut être apportée au réseau présenté dans la figure 6.20 est l'inclusion de la récurrence dans la couche de sortie. Le réseau ne ressemble alors plus à un réseau à récurrence par plaques mais plutôt à un réseau récurrent fortement connecté avec une couche d'entrée bien distincte du reste du réseau. Cette architecture a été étudiée, avec de bons résultats, dans [cummins93] et [cummins94]. La mise en place d'une couche totalement récurrente en couche de sortie peut encore être développée par l'adjonction de délais encastrés. La couche de sortie combine alors des mécanismes de représentation implicite et de représentation explicite du temps. Cette combinaison peut se voir critiquer et l'adjonction de délais pourra facilement être considérée comme une heuristique choisie à propos, mais ce choix permet d'obtenir une architecture qui a comme avantage de donner de très bons résultats dans certains cas [frasconi93]. Le mécanisme de la ligne de délais a été utilisé de manière encore plus intensive dans [giles94a] et [giles94b]. Dans ce cas, l'entrée du réseau est conservée pendant m pas de temps tandis que la sortie du réseau, qui est rebouclée en entrée, est conservée pendant n pas de temps. Le succès ou l'échec d'un tel réseau dans la phase d'apprentissage d'une tâche particulière dépendra bien sûr de l'adéquation entre les paramètres m et n et la tâche elle-même. Mais la découverte de la bonne adéquation permet d'obtenir, là encore, de très bons résultats. Ce type de mécanisme a, par ailleurs, été repris au niveau du neurone comme nous le verrons plus loin, au paragraphe 6.3.9.

Les premières architectures de la catégorie des réseaux à récurrence par plaque n'ont pas été oublié dans ces recherches. [zeng94] propose donc d'utiliser des réseaux ressemblant à ceux du modèles de Jordan pour résoudre les problèmes d'implantation d'automates dans les réseaux de neurones. La proximité entre le modèle de Jordan et le modèle exposé ici est cependant toute relative. La ressemblance entre ce dernier et une bascule RS, dont il est possible de trouver une définition dans tout bon cours de structure des ordinateurs, est d'ailleurs assez plaisante. Le modèle de Elman n'a pas non plus été oublié et son application à la reconnaissance de grammaires par implantation d'automates pourra être trouvée dans [servan91].

Le problème de l'implantation d'automates a également été l'occasion d'implanter de nouvelles architectures à récurrence par plaque. [tino95] propose ainsi une architecture qui utilise en fait deux réseaux : un premier réseau de traitement et un deuxième réseau d'arrière plan pour le stockage des états de l'automate. Alors que le premier réseau possède l'architecture générale d'un perceptron multicouche, le deuxième réseau, constitué de deux couches, permet de conserver un vecteur d'états par rebouclage. Un dépliage de la figure 6.21 supprimant les intersections de flots permet de voir que cette architecture crée en fait une temporisation d'un pas de temps entre la première et la deuxième couche du réseau principal. Le réseau secondaire tient cependant compte du vecteur de sortie du

réseau d'états pour calculer ses nouvelles valeurs d'activation, créant ainsi une mémoire à plus d'un délai de terme. Ce réseau a prouvé avoir de bonnes capacités de modélisation [tino95] et peut être considéré comme architecturalement proche des réseaux à propagation dynamique de l'erreur (paragraphe 6.3.4).

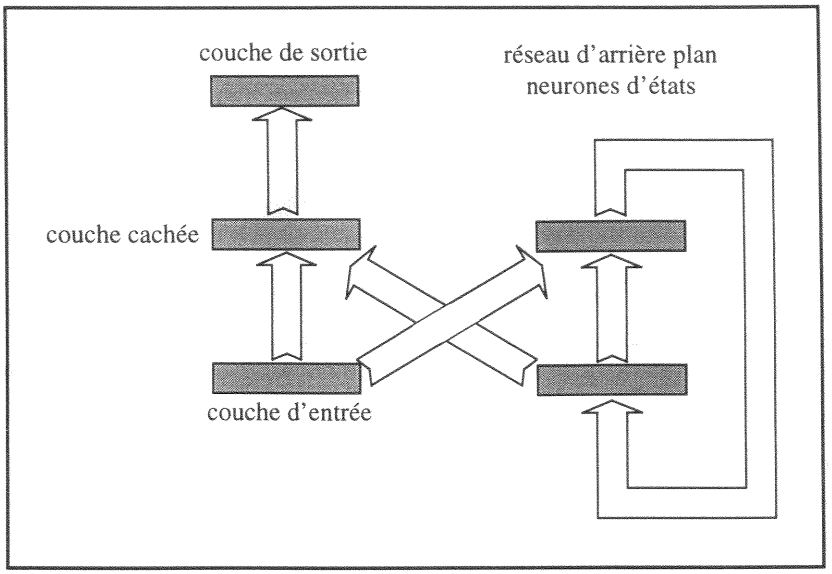


Figure 6.21 : Réseau à récurrence par plaque pour l'encodage d'automates (d'après [tino95]).

La compréhension des mécanismes utilisés par les réseaux pour implanter des automates n'est malheureusement pas aussi simple que avons bien voulu le faire croire au début de ce paragraphe. Si la gestion d'automates est, elle, très simple, l'implantation connexionniste d'automates s'avère en fait beaucoup plus problématique. Les réseaux connexionnistes récurrents sont, en effet, capables de simuler des systèmes dynamiques non linéaires. Mais cette modélisation est, de manière générale, imparfaite puisque l'initialisation aléatoire des poids connexionnistes du réseau et l'apprentissage d'une tâche par une méthode ne permettant pas d'atteindre l'optimum ne permettent pas d'obtenir un système final parfaitement stable.

Une étude du fonctionnement des réseaux connexionnistes récurrents pourra être trouvée dans [cummins93], [cummins94] et [casey95]. Ces études nous mènent très facilement sur des études ayant un rapport certain avec d'autres domaines tels que l'étude des systèmes oscillants avec les notions de pas de phase dynamique et de pas de phase géométrique [postma96]. Ces variations de représentation interne ont d'ailleurs poussé certaines équipes de recherche à s'orienter vers la définition d'un système à tolérance de fautes [omlin95a]. Une manière de comprendre le comportement d'un tel réseau après apprentissage peut également passer par une tentative d'extraction des règles qui ont été apprises [omlin95b]. Mais l'extraction de règles à partir d'un système connexionniste relève, nous semble-t-il, de la recherche d'un Graal bien difficile à atteindre.

Signalons enfin que certaines recherches [omlin94] ont prouvé l'importance du choix du codage des entrées, choix qui peut grandement influer sur les capacités d'encodage et, donc, de reconnaissance.

6.3.7/ Amélioration des capacités d'encodage d'automates

Les automates qu'il est possible de coder avec les modèles de réseaux récurrents présentés dans le paragraphe précédent ne permettent cependant pas de modéliser toutes les grammaires. Il est par exemple impossible de représenter des grammaires à contexte libre, telles que la grammaire de langages correctement parenthésés, de type 1^n0^n, ou la grammaire des palindromes. La modélisation de ces grammaires nécessite la mise en place d'une structure de file, premier entré - premier sorti, ou

d'une structure de pile, premier entré - dernier sorti, pour le stockage temporaire des niveaux d'imbrication. Il faut donc définir un système connexionniste capable de modéliser un automate mais qui utilise également un mécanisme similaire aux délais encastrés. De telles recherches ont été menées dans [sun95] et ont conduit au développement d'un modèle, le *Neural Network Pushdown Automata*, NNPDA, utilisant une pile de stockage en plus des concepts exposés précédemment (cf. figure 6.22). Cette pile est commandée par le réseau récurrent qui pourra la contrôler grâce à trois opérations : *push*, commande d'empilement d'une valeur, *pop*, commande de dépilement d'une valeur, et *nop*, *no operation*, pour ne pas modifier l'état de la pile. Une seule sortie du vecteur d'action de la sortie contrôle ces trois opérations qui peuvent par exemple être représentées par -1, 0 et 1. Les autres valeurs du vecteur d'action sont stockées sur la pile. La dernière valeur en date stockée sur la pile est fournie en entrée du réseau quelque soit la dernière opération effectuée. Ainsi, une même valeur sera présentée au réseau tant qu'aucune opération n'aura été effectuée.

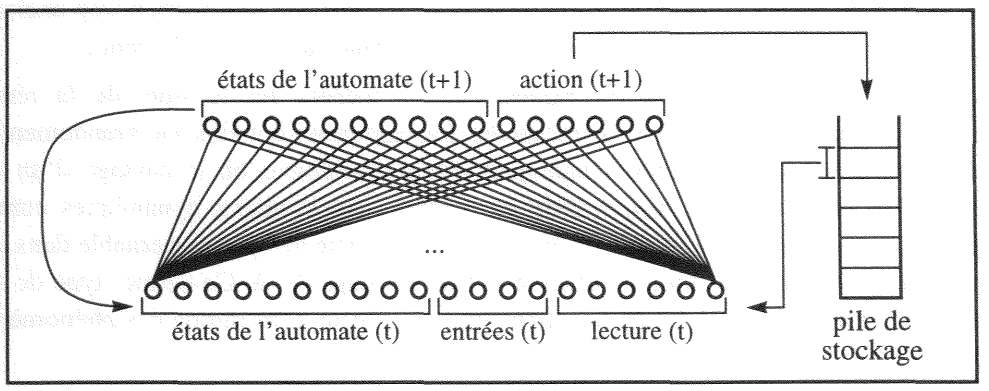


Figure 6.22 : Architecture d'un NNPDA, *Neural Network Push Down Automata* (d'après [sun95]).

Les réseaux NNPDA ont été testé avec succès sur des tâches nécessitant l'emploi conjoint des mécanismes d'automate et de pile. Ils ont ainsi été testés avec succès sur des grammaires de type 1^n0^n ou sur des grammaires de palindromes. À notre connaissance, aucune expérience supplémentaire n'a été menée avec ce modèle qui semble pourtant prometteur puisque la pile pourrait très bien être utilisée dans des tâches de modélisation temporelle, la durée d'un événement pouvant être marquée par un empilement plus ou moins important de marqueurs.

Un autre type de réseaux peut également être défini pour représenter une classe d'automates plus large que celle des simples automates déterministes : les automates flous. La définition des automates flous est un peu différente de celles des automates déterministes que nous avons vu au chapitre 5, paragraphe 5.2.2.2. La définition d'un automate flou ne comporte, par exemple, pas d'ensemble de nœuds d'acceptation, l'acceptation d'une séquence de symboles terminaux dépendant de la valeur maximale des dérivations de toutes les chaînes de production possibles, ces dérivations étant calculées en tenant compte du poids minimum de la règle de réécriture. Ces poids des règles de réécriture constituent la deuxième différence existant entre les automates déterministes et les automates flous. Dans les automates flous, les règles sont valuées d'un potentiel de transition compris entre 0 et 1. Ces valeurs ne sont pas, à proprement parler, des probabilités puisque la somme des valeurs des règles partant d'un nœud peut être supérieure à 1. Ces valeurs permettent donc d'avoir une indication, floue, sur les chances qu'a la règle d'être employée lors de l'analyse d'une séquence d'entrée.

Implanter des automates flous dans un réseau connexionniste nécessite l'utilisation d'un réseau divisé en deux parties, une première partie assurant le codage de l'automate comme auparavant tandis qu'une deuxième partie assure la classification floue à proprement parler [omlin96]. La seule application tentée avec ce type de réseaux n'est malheureusement pas très représentative de problèmes réels puisqu'un tel réseau a servi à la modélisation d'un automate flou représenté par un

corpus d'apprentissage de taille restreinte et généré aléatoirement.

6.3.8/ Les problèmes de l'apprentissage

Des études ont été menées pour étudier les capacités des réseaux connexionnistes récurrents à conserver des informations sur une longue période de temps. Ces études se rattachent au problème de la reconnaissance de séquences abstraites et ont permis de vérifier les lacunes des modèles connexionnistes dans ce domaine. Ces études justifient et valident d'une certaine manière les réalisations comme celle des réseaux aptes à simuler des automates à piles.

Les réseaux connexionnistes récurrents restent avant tout des réseaux connexionnistes. Lorsque le seul stockage effectué par le réseau l'est par l'intermédiaire de connexions récurrentes, l'information stockée depuis un certain temps finit par être totalement annihiler au profit de l'information plus récente. Bien que ce processus de disparition puisse être atténué par l'implantation de lignes de délais au sein du réseau, ce dernier ne pourra donc qu'oublier une information trop ancienne, un peu à la manière d'un être humain mais, dans tous les cas, beaucoup plus rapidement...

Diverses études, pratiques et théoriques, ont été menées sur le sujet de la rémanence de l'information par plusieurs équipes de recherche. Les automates ont ainsi été grandement étudiés car ils correspondent à des systèmes dynamiques discrets, systèmes où le passage d'un attracteur à l'autre se fait rapidement et franchement, au contraire des systèmes dynamiques continus, où le passage d'un état à l'autre se fait plus lentement et de manière très peu discernable dans le cas où les observations du système se font à un taux d'échantillonnage élevé. Ce dernier type de systèmes se retrouve, entre autre, dans le cas de la génération de la parole. L'évolution des phénomènes voisés y est par exemple assez lente.

Il faut cependant considérer cette distinction entre systèmes dynamiques discrets et systèmes dynamiques continus comme assez générale et insuffisante pour comprendre tous les problèmes posés dans ces systèmes. Ainsi, bien que le modèle de Jordan [jordan86] doive être considéré comme un modèle permettant d'implanter un automate, et donc un système dynamique discret, il est bon de noter, comme [pearlmutter89], que la notion temporelle dans ce modèle est beaucoup moins contraignante que dans d'autres. La génération de plan de séquences ne dépend pas, en effet, des valeurs d'entrée puisqu'une unique clé permet de générer l'ensemble de la séquence (cf. paragraphe 6.3.1). Le rythme d'échantillonnage des données en entrée étant un concept totalement absent, il n'y a pas, ici, de problème de perte de données par atténuation progressive puisque chaque pas de génération se fait au rythme voulu par le concepteur pour la génération de la séquence de sortie.

Trois types de problèmes permettent de juger des capacités véritablement temporelles d'une architecture connexionniste ont été donnés dans [bengio94a]. Un premier type de problème est appelé le problème de loquet, ou *latch problem*. Dans ce genre de problème, il est demandé au réseau de conserver une valeur, ou une décision prise d'après un ensemble de valeurs, pendant un laps de temps non nul sans se préoccuper des données qui sont observées par la suite en entrée. Ce type de problème fait donc appel à la modélisation d'au moins deux états, un d'acceptation et un de rejet, l'état choisi devant être conservé pendant un temps a priori non défini. Un deuxième type de problème concerne la classification de deux séquences différentes, c'est le *2-sequence problem*. Ce type de problème consiste à repérer les indices remarquables de chacune des deux séquences pour décider correctement à quel type de séquence appartiennent les entrées observées. Ce problème peut être défini avec une complexité variable, fonction du nombre d'indices remarquables et de la distance entre les deux séquences, que cette distance soit euclidienne, carrée ou de tout autre ordre. Ce problème peut bien évidemment se généraliser à un problème de classification de n séquences, plus ou moins semblables. Le troisième et dernier type de problème est une généralisation temporelle du problème de la parité qui est lui-même une généralisation du problème du XOR, problème qui a eu une importance non négligeable en connexionnisme [minsky69]. Ce type de problème a été baptisé le *parity problem* dans [bengio94a]. Il consiste en la simple transformation d'un vecteur

d'entrée dont la parité doit être découverte en une série temporelle dont la parité doit être donnée en cours ou en fin de traitement. Là encore, deux états au moins doivent être modélisés, l'un pour marquer la parité et l'autre pour marquer la non parité de la séquence dans le cas binaire. Le problème peut se résumer à la mise en place d'un moyen efficace de passage d'un état à l'autre en fonction de l'état courant de parité et de l'entrée observée.

Des études ont été faites dans [bengio94a] sur le problème du loquet avec un réseau de neurones dont l'architecture est donnée à la figure 6.23 (il ne s'agit effectivement pas d'un réseau à récurrence par plaque...). Ces études ont montré que la modélisation de tels problèmes avec pareille architecture était impossible. Cette impossibilité provient de deux raisons dont la première est liée à la deuxième. La première raison est que les connexions synaptiques ne permettent pas de modéliser des mécanismes de rétention de l'information et l'information la plus ancienne est donc de plus en plus diluée dans l'activité totale des unités. Ces connexions synaptiques sont, elles-même, définies par le processus itératif d'apprentissage utilisant la méthode de descente du gradient d'erreur. [bengio94a] prouve que cette méthode devient de plus en plus inefficace à mesure que la fenêtre temporelle nécessaire à la résolution du problème devient grande. Et cette inefficacité est malheureusement prouvée pour tout critère d'erreur puisque la preuve est indépendante de cette notion. Ceci prouve que le critère du moindre carré de l'erreur, communément utilisé, ne vaut pas la peine d'être changé dans l'espoir d'obtenir de meilleurs résultats. Si une solution existe au problème de l'apprentissage à long terme, il ne se trouve pas dans la modification du critère d'erreur mais certainement à un niveau beaucoup plus fondamental, le choix d'une autre méthode d'apprentissage pouvant être envisagé.

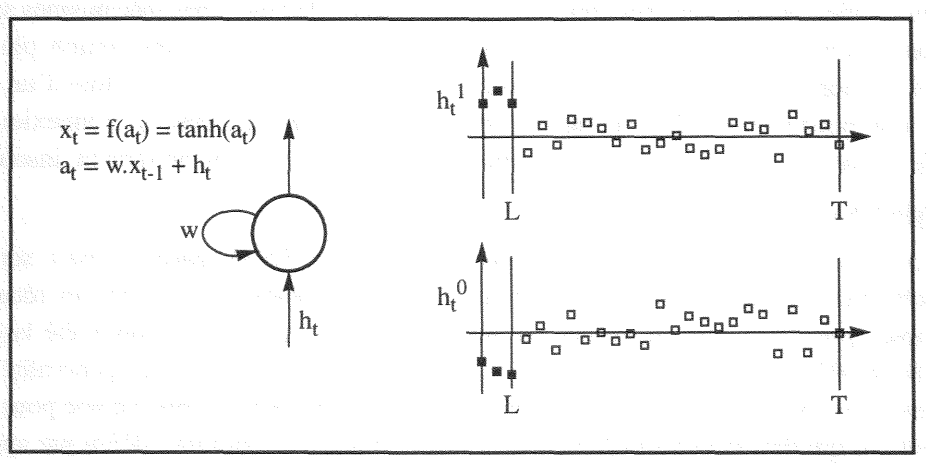


Figure 6.23 : Le neurone type et le problème du loquet étudiés dans [bengio94a].

Certaines recherches essaient donc de changer les concepts de base de l'apprentissage. [nerrand94] milite ainsi pour l'emploi de deux types d'apprentissage, chacun étant réservé à un domaine précis. La modélisation de systèmes dynamiques non bruités devrait se faire à l'aide d'un mode d'apprentissage différent de celui employé pour la modélisation des systèmes dynamiques bruités. [baldi95] rappelle cependant que la rétropropagation du gradient d'erreur est la meilleure méthode connue à ce jour et qu'elle surpasse en qualité d'autres méthodes d'apprentissage utilisables en modélisation de systèmes dynamiques. [baldi95] constate cependant qu'il n'existe encore aucun lien entre l'apprentissage de haut niveau, concernant la tâche, et l'apprentissage de bas niveau, concernant les modifications synaptiques effectuées par la méthode de rétropropagation du gradient d'erreur ou par la méthode d'apprentissage de Hebb. Or ce manque de connaissances pourrait très bien être la cause des problèmes rencontrés dans la modélisation de systèmes dynamiques lorsque la méthode de rétropropagation est généralisée de la dimension architecturale à une dimension à la fois architecturale et temporelle. Préalablement à ces constatations, [baldi94] avait effectué des études sur la modélisation de systèmes dynamiques à l'aide de modèles connexionnistes dont les neurones permettent de modéliser des délais. Les équations de tels neurones, donnés d'après [baldi94] sont :

$$\frac{du_i}{dt} = -\frac{u_i}{\tau_i} + \sum_j T_{ij} f_j(u_j) + I_i \qquad \text{(Éq. 6.15)}$$

cette première équation pouvant être généralisée pour donner l'équation 6.16 suivante :

$$\frac{du_i}{dt} = -\frac{u_i}{\tau_i} + \sum_j T_{ij} f_j(u_j(t-\tau_{ij})) + I_i \qquad \text{(Éq. 6.16)}$$

L'introduction de l'équation 6.15 comme définition du neurone influence énormément la stabilité du réseau. En effet, les réseaux, initialement stables avec la définition originale du neurone, peuvent devenir instables du fait des oscillations provoquées par les interactions entre délais. Ces délais permettent cependant de modéliser, dans les cas favorables, des oscillations stables par connexion réciproque de deux populations de neurones, l'une permettant l'excitation pendant que l'autre assure l'inhibition. [baldi94] constate qu'un accroissement du nombre des délais permet d'augmenter la période des oscillations d'un système et augmente ainsi le spectre des fréquences que le système peut représenter et donc l'étendue des problèmes que le système peut traiter. [baldi94] remarque enfin que peu d'études mettent en relation les réseaux de neurones aptes à la dynamique, tels que ceux adoptant l'équation 6.15 ou l'équation 6.16 comme définition du neurone, et la théorie du chaos, ce dernier champ d'études s'intéressant en effet à d'autres systèmes d'équations.

Il est cependant possible de modéliser efficacement certains types de systèmes dynamiques. Les modèles connexionnistes utilisés alors implantent bien évidemment des mécanismes spécifiques qui leurs permettent de résoudre des problèmes auxquels des modèles d'inspiration plus générale ne peuvent pas apporter de solution. Nous allons donc maintenant voir l'architecture d'un réseau dont le champ d'application est plus en rapport avec l'automatisme qu'avec le connexionnisme et qui possède de très bonnes capacités de résolution de problèmes qui sont, autrement, insolubles.

6.3.9/ Modèles NARX

Les réseaux NARX, *Non-linear Auto Regressive models with eXogenous inputs*, sont une réponse aux problèmes de l'apprentissage de séquences dans les réseaux connexionnistes récurrents qui ont été exposés précédemment, en particulier au problème du loquet [lin95] qui a été largement étudié dans [bengio93b] et [bengio94a]. Ce type de réseau n'a en fait pas la généralité des modèles auxquels il est comparé puisque l'architecture fait appel à des mécanismes ad hoc pour conserver des informations qui devraient normalement être conservées par un automate défini par apprentissage au sein du réseau. Ces mécanismes particuliers sont au nombre de deux et utilise chacun une ligne de délais encastrés pour stocker l'information nécessaire à la résolution du problème. Une première ligne de délais est implantée en entrée du réseau pour stocker les éléments successifs de la séquence à analyser. La longueur de cette ligne, notée n_u dans la figure 6.24, peut être ajustée en fonction du problème à résoudre et le nombre de délais peut très bien être égal au nombre d'éléments de la séquence, ce qui facilite grandement la mémorisation. Une deuxième ligne de délais permet de conserver la trace des réponses passées de réseau. La longueur de cette ligne de délais, notée n_y, est, là encore, ajustable en fonction du problème à résoudre. Un troisième paramètre architectural, H, défini le nombre de neurones en couche cachée.

Ces deux lignes de délais permettent ainsi de stocker l'information qu'il n'est plus besoin de stocker dans des états finement définis au sein d'une implantation d'automate. L'information n'est donc plus stockée par récurrence bien que l'architecture de ce type de réseau exploite une récurrence de la sortie vers l'entrée.

Malgré une utilisation judicieuse des mécanismes de lignes de délais, le réseau NARX ne possède pas une généralité suffisante pour remplacer, dans tous les cas, les réseaux à récurrence par plaque. Comme le souligne [siegelmann95], le réseau NARX ne peut pas implanter l'ensemble des

automates qu'un réseau à récurrence par plaque pourrait extraire d'un corpus d'apprentissage. Cet ensemble est en fait restreint aux seules machines à mémoire finie (*FMM, Finite Memory Machine* [siegelmann95]) contrairement aux réseaux récurrents qui peuvent implanter tout automate de n états dans un réseau à quatre couches. Ceci n'empêche pas les réseaux NARX de pouvoir parfaitement résoudre des problèmes de loquet ou de parité (cf. paragraphe 6.3.8).

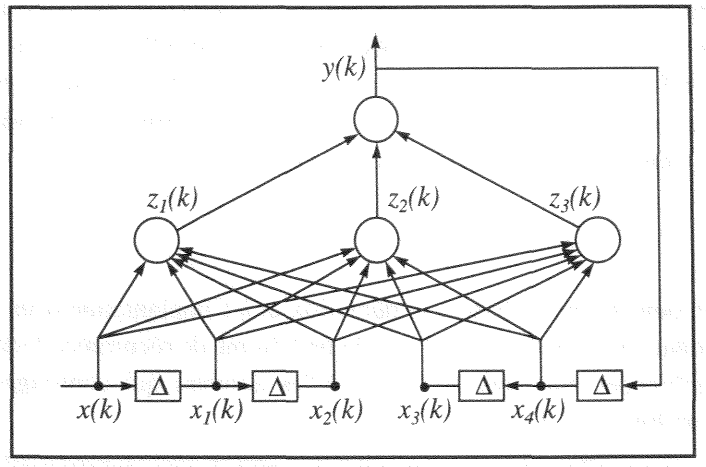


Figure 6.24 : Un réseau NARX de paramètres $n_u = 2$, $n_y = 2$, $H = 3$, (d'après [siegelmann95]).

Les réseaux NARX ne sont déjà plus tout à fait des réseaux de neurones à récurrence par plaque comme le montre la figure 6.24. La structure de plaque est, ici, plus diffuse que dans les modèles que nous avons vu jusqu'ici dans tout le paragraphe 6.3. Elle conduit presque naturellement aux architectures connexionnistes à récurrence locale que nous allons voir maintenant.

6.4/ Réseaux connexionnistes à récurrence locale

6.4.1/ Modèles de neurones formels

La très grande majorité des systèmes neuromimétiques, dont tous ceux que nous venons de voir, sont basés sur la modélisation formelle du neurone faite par McCulloch et Pitts en 1943 [mcculloch43]. L'équation de ce neurone peut être simplifiée pour montrer la simplicité du principe mis en œuvre. L'équation 6.17, reflet de l'équation originale, définit le principe de calcul d'une activation connexionniste par un premier calcul de l'activation interne du neurone, par sommation pondérée des entrées, cette activation étant ensuite transformée par la fonction signe, fonction non linéaire non continûment dérivable (cf. chapitre 2, paragraphe 2.2.1).

$$y_j = sign\left(\sum_i w_{ji} y_i\right) \qquad \text{(Éq. 6.17)}$$

Cette modélisation initiale a été reprise dans de nombreux modèles connexionnistes avec diverses variantes, la principale modification étant, aujourd'hui, l'adoption d'une fonction non linéaire continûment dérivable permettant un apprentissage par rétropropagation du gradient d'erreur sur plusieurs couches d'unités (cf. chapitre 2, paragraphe 2.4.4). Comme le montre l'équation 6.17, la seule référence temporelle existant dans cette modélisation est une référence à l'instant courant puisqu'aucune valeur d'un instant antérieur n'est présente. La seule technique possible pour prendre en compte des événements dont l'origine temporelle est différente est alors d'utiliser une récurrence à un niveau architectural global ce qui conduit aux architectures à récurrence forte (cf. para 6.2) ou aux architectures à récurrence par plaques (cf. para 6.3).

Ces deux solutions, faisant appel à une disposition particulière des connexions, ne sont pas les seules architectures récurrentes possibles. Une troisième possibilité, faisant appel à une récurrence locale au neurone lui-même, peut également être envisagée.

Un exemple de modélisation de haut niveau de ce type d'architecture est donnée dans [usher95]. Ce rapport technique, en plus d'énoncer certaines caractéristiques qu'il serait intéressant de trouver dans une modélisation connexionniste générale, présente une modélisation du neurone qui se veut plus proche de la réalité neurobiologique. Cette modélisation a comme particularité de mettre en place une récurrence locale au neurone pour modéliser l'évolution des phénomènes chimiques et électriques internes au neurone, évolution qui est progressive au contraire de la brutalité du changement d'activité qui caractérise la modélisation de McCulloch et Pitts. Le modèle de neurone définit dans [usher95] est redonné par l'équation 6.18 où I_i représente le potentiel d'activité reçu d'autres neurones, où k représente le taux de décroissance interne du potentiel et où ξ représente le bruit intrinsèque, considéré par les auteurs comme gaussien.

$$\tau \frac{\partial x_i}{\partial t} = I_i - kx_i + \xi \qquad \text{(Éq. 6.18)}$$

Cette équation peut servir de base à la définition et à l'implantation d'un réseau connexionniste. Elle n'est cependant pas la seule possible car le paradigme de récurrence locale au neurone peut être implanté de nombreuses manières avec des règles annexes qui contraignent plus ou moins les possibilités de l'architecture définie.

Un premier choix possible, qui est primordial du point de vue architectural, est la mise en place de la récurrence autour de la non linéarité ou en dehors de la non linéarité. La mise en place de la récurrence autour de la non linéarité signifie que la valeur de sortie du neurone, obtenue après modification de la valeur d'activation interne par la fonction non linéaire, sera rebouclée en entrée du neurone. La valeur de l'activation du neurone est donc fournie en entrée au même titre que les informations en provenance des unités afférentes. Ce type de récurrence pourrait, éventuellement, provenir de la sortie même de la phase de sommation des valeurs d'entrées avant toute modification par la fonction non linéaire. Une deuxième possibilité architecturale consiste à traiter la valeur issue de la transformation non linéaire de manière totalement indépendante vis-à-vis du mécanisme de sommation. La valeur d'activation est alors traitée par un mécanisme plus ou moins complexe qui constitue une phase de traitement postérieure aux traitements normalement effectués dans le neurone de McCulloch et Pitts. Le mécanisme utilisé est, le plus généralement, un filtre implantant un mécanisme de décroissance exponentielle de manière à simuler, ou plus vraisemblablement modéliser, les modifications de l'activation observables dans les cellules biologiques. Ces phénomènes biologiques sont la base des principes d'excitations ou d'inhibitions retardées qui permettent de modéliser une activité temporelle à un niveau local ou global.

Quelque soit le choix effectué dans le domaine architectural comme nous venons de le définir et qui reste assez général, il reste encore à préciser l'architecture effective qui sera utilisée. Au rang des paramètres à prendre en compte se trouve le nombre des délais qui seront utilisés pour la rétention de l'information, ce nombre pouvant être, au minimum, égal à un. Le nombre de délais à mettre en place n'est cependant pas le seul choix à effectuer. Il est également primordial de savoir comment sera traitée l'information mémorisée. Il est ainsi possible de décider de modifier l'information stockée par des poids synaptiques internes au mécanisme de récurrence, le terme de poids synaptique étant ici abusif, ou, à l'inverse, de laisser cette information disponible dans la forme sous laquelle elle a été prise en compte. Une conséquence du choix de l'implantation de poids synaptiques dans le mécanisme de mémorisation en dehors de la récurrence est la mise en place d'une éventuelle contrainte sur ces poids. En effet, ces poids, qui sont plus exactement des coefficients de régression, peuvent avoir une influence très forte sur les capacités de convergence, et donc d'apprentissage, du réseau. Certains choix architecturaux imposent donc de contraindre les coefficients sur certaines plages de valeurs possibles où la stabilité du filtre est prouvée mathématiquement. Mais il ne faut pas oublier qu'une preuve de stabilité d'un mécanisme autorégressif, ou plus généralement d'un filtre, n'est pas une preuve de convergence d'un réseau constitué d'un ensemble de filtres.

Enfin, l'usage de mécanismes tels que ceux exposés imposent parfois de réécrire un algorithme d'apprentissage prenant en compte la nouvelle architecture. Cela est particulièrement vrai dans le cas où le mécanisme de récurrence est placé après la fonction non linéaire. Une telle modification empêche en effet un calcul du gradient d'erreur selon une méthode aussi simple que celle employée habituellement [lecun85]. Le lecteur pourra aisément se rendre compte de ceci en consultant l'annexe 1 où sont présentées les équations développées pour notre modèle.

L'utilisation d'un mécanisme de récurrence locale n'impose pas d'implanter d'autres mécanismes de récurrence au sein du réseau. Il est ainsi possible de définir une architecture dont l'aspect général est équivalent à celui d'un perceptron multicouche, seule la définition locale des neurones variant par rapport à la définition de ceux employés dans un perceptron. Cette possibilité permet de définir une nouvelle classe de réseaux neuromimétiques qui sont baptisés *Locally Recurrent Globally Feedforward* [tsoi94] ou *Local Feedback Multilayered Networks* [frasconi92].

6.4.2/ Réseaux à récurrence locale et retour antérieur

La première catégorie de réseaux à récurrence locale dont nous allons parler brièvement regroupe les réseaux où le temps n'est pas singularisé par rapport aux entrées. La ou les valeurs de sorties de la fonction non linéaire ou de la somme pondérée sont rebouclées sur l'entrée du neurone. La connaissance temporelle est donc accessible de la même manière que l'information provenant des neurones afférents. Il est possible de considérer ce mécanisme comme une adaptation à une échelle moindre, un *downsizing* en quelque sorte, des mécanismes utilisés par les réseaux de Jordan, d'Elman ou par les réseaux NARX.

Le désavantage que nous voyons à cette méthode est justement sa définition architecturale. L'effet "boîte noire" du réseau de neurones ne porte plus uniquement sur la représentation des connaissances par poids connexionnistes entre les neurones du réseau mais également sur l'exploitation des informations temporelles au sein du neurone lui-même, le cas le plus problématique étant celui des réseaux de neurones à récurrence locale et retour antérieur puisque le neurone lui-même n'est pas capable de distinguer les deux types d'information. La juxtaposition sur un même plan d'informations actuelles et antérieures ne permet pas au réseau de les distinguer véritablement. Cette juxtaposition peut également empêcher une exploitation ultérieure des valeurs de ces connexions temporelles puisque leur fusion sémantique avec les poids connexionnistes normaux pourrait contrarier une éventuelle modification de ces valeurs en cours d'utilisation du réseau, modification dont nous reparlerons au chapitre 8.

Ce mécanisme est cependant avantageux par rapport aux réseaux à récurrence locale avec retour postérieur que nous verrons plus loin. Les informations temporelles étant présentées de la même manière que les informations venant des neurones afférents, aucune réécriture de la procédure d'apprentissage n'est nécessaire : l'apprentissage par rétropropagation du gradient d'erreur est dans ce cas totalement identique à ce qu'elle est dans les perceptrons multicouches. La seule modification véritable est l'étape de recopie des valeurs de sortie vers l'entrée du neurone comme cela se fait dans les réseaux à récurrence par plaques.

6.4.2.1/ Modèle de la rétropropagation pour les séquences

Le modèle de la rétropropagation pour les séquences est le premier modèle que nous présenterons ici au titre des récurrences locales à retour antérieur. Ce modèle a été baptisé BPS par ses concepteurs, *Back Propagation for Sequences*. Le neurone, défini dans [gori89], est d'une structure relativement simple puisqu'il ne s'agit que d'une simple ligne de délais encastrés rebouclant en entrée du neurone les activations passées de celui-ci sur m pas de temps antérieurs. Les afférences du neurone en provenance d'autres unités sont valuées, comme pour le modèle de McCulloch et Pitts, par des pondérations. Ces pondérations ont été notées w_i dans le schéma de la figure 6.25. Au même titre, les anciennes valeurs d'activation du neurone qui sont stockées dans la ligne de délais sont valuées par des poids connexionnistes notés k_i dans la figure 6.25. La méthode d'apprentissage n'est

que très peu modifiée par cette adjonction architecturale, les délais de la ligne pouvant, en fait, être considérés comme autant d'unités connexionnistes afférentes.

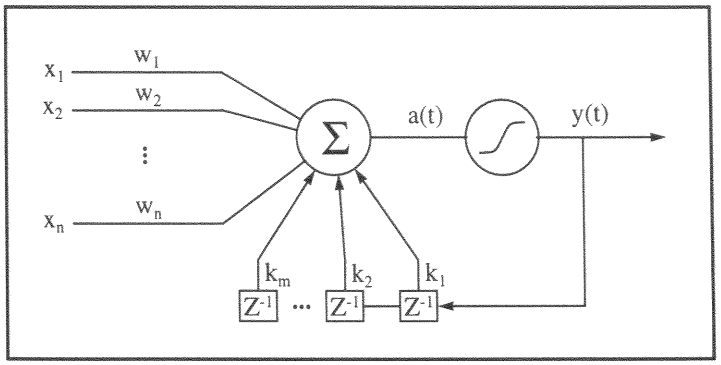


Figure 6.25 : Un neurone de *Back-Propagation for Sequence*.

Formellement, ce neurone se définit par l'équation 6.19 où le premier terme de la somme de droite représente la contribution du passé et où le deuxième terme représente la contribution du présent.

$$y(t) = f\left(\sum_{j=1}^{m} k_j y(t-j) + \sum_{i=0}^{n} w_i x_i(t) \right) \qquad \text{(Éq. 6.19)}$$

Comme son nom l'indique, ce neurone a été défini pour traiter des séquences. Le terme de séquences est ici à considérer de manière large puisqu'il peut aussi bien s'agir de séquences abstraites que de phénomènes physique comme la parole [bengio96].

[tsoi94] rappelle que cette architecture est générique et que la ligne de délais pourrait très bien être remplacée par un filtre beaucoup plus général comprenant plusieurs pôles et plusieurs zéros. Ce choix a été fait lors de certaines recherches, conduisant à d'autres architectures assez proches mais dont nous ne parlerons pas ici. Cependant, le lecteur pourra, par exemple, se reporter aux articles publiés sur le modèle du neurone à temps discret [bresslof93] ou certains systèmes utilisés pour l'identification de systèmes [burrows93].

Le principe de récurrence locale avec retour antérieur a également été appliqué au modèle du *Time Delay Neural Network* [waibel89]. Cette extension architecturale a permis d'obtenir de meilleurs résultats sur des tâches de reconnaissance de la parole, prouvant ainsi que ce type de mécanisme est capable d'apporter une capacité supplémentaire de modélisation à une architecture connexionniste dont les capacités initiales avaient déjà démontrées par un usage très répandu en reconnaissance automatique de la parole.

6.4.2.2/ TDNN récurrent

Le principe de récurrence locale a été appliqué au TDNN par [greco91]. Le réseau ainsi obtenu combine donc une représentation externe du temps à une représentation interne explicite à représentation algébrique. Le mécanisme de la plaque d'entrée est conservé ainsi que le principe de partage des poids. Ces deux caractéristiques permettent de conserver la possibilité d'un traitement géométrique du signal observé mais ne suppriment pas toutes les lacunes. [greco91] présente trois problèmes qui lui semble important à résoudre :

- l'information contextuelle n'est pas distribuée de manière uniforme. Cela nécessite de mettre en œuvre un mécanisme d'oubli progressif car l'utilisation d'une fenêtre d'entrée de grande taille peut entraîner des problèmes d'apprentissage,
- la vitesse d'élocution n'est pas uniforme sur l'ensemble d'un corpus et bien que ce fait n'ait aucune influence sur les systèmes utilisant des fenêtres de petite taille, les conséquences peuvent être dramatiques avec des fenêtres plus larges.
- l'agrandissement de la taille de la fenêtre impose d'utiliser plus de poids à l'apprentissage et

nécessite donc des corpus d'apprentissage de plus grande taille pour tenter d'éviter les problèmes.

C'est pour résoudre ces problèmes que le TDNN se voit étendu d'un mécanisme de récurrence local avec retour antérieur selon l'architecture présentée à la figure 6.26.

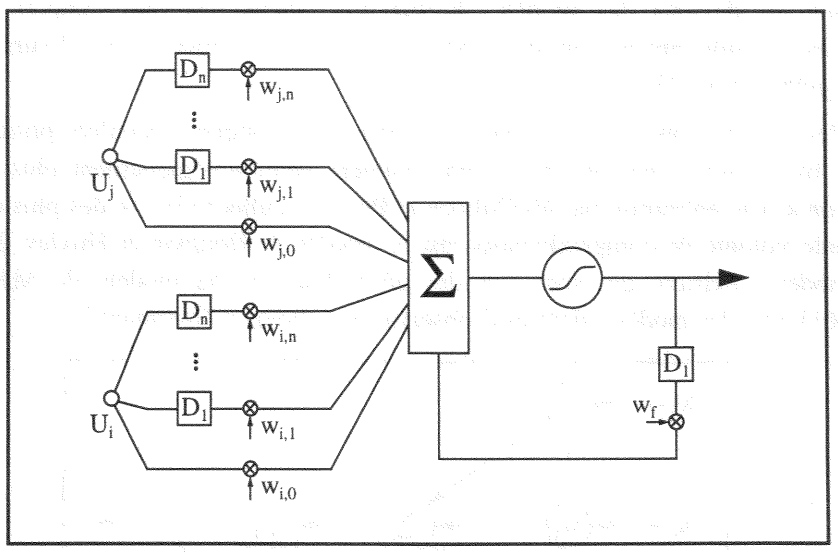


Figure 6.26 : Architecture d'un neurone d'un TDNN récurrent (d'après [greco91]).

L'application de ce modèle à la reconnaissance de la parole a permis d'obtenir de meilleurs résultats qu'avec l'architecture originale du TDNN ce qui, selon l'auteur [greco91], prouve la supériorité des systèmes récurrents pour le traitement de problèmes de nature séquentielle comme la parole. Le système récurrent choisi appartient cependant à une catégorie de systèmes à récurrence locale que nous considérons comme moins intéressante que celle des réseaux à récurrence locale et retour postérieur que nous allons voir maintenant.

6.4.3/ Réseaux à récurrence locale et retour postérieur

Cette catégorie de réseaux connexionnistes correspond aux cas où la récurrence locale est un mécanisme totalement indépendant de la phase d'intégration des connaissances par la somme pondérée à l'entrée du neurone. La récurrence locale se trouve dans ce cas après la fonction non linéaire et le traitement du temps se fait de manière totalement autonome.

Cette définition de la prise en compte du temps par récurrence locale correspond beaucoup plus au mécanisme défini dans [usher95] qui présente une des formalisations possibles des mécanismes neurobiologiques du cerveau (paragraphe 6.4.1).

Dans les modèles que nous allons présenter, le temps est un phénomène local au neurone, tout comme pour les modèles de neurones présentés au paragraphe 6.4.2. Il est cependant géré par un mécanisme indépendant de l'intégration connexionniste. Cette indépendance permet de mieux appréhender et, éventuellement, de mieux maîtriser le mécanisme en cause. Il serait donc possible de plus facilement modifier le comportement de cette mémoire en cours d'utilisation [pican95].

Nous avons également choisi de présenter des modèles d'architectures connexionnistes pour lesquels les traitements temporels sont assez complexes du fait du mode de calcul choisi. Nous présenterons aussi des modèles pour lesquels la représentation du temps n'est pas véritablement locale au neurone mais dont les principes peuvent être rapprochés de ceux des réseaux à récurrence locale avec retour postérieur comme, par exemple, les réseaux représentant le temps au niveau des connexions dont le lien avec des modèles plus représentatifs pourra être fait assez aisément.

6.4.3.1/ Réseaux de neurones chaotiques

Les neurones peuvent, eux aussi, être chaotiques. Certains modèles de réseaux connexionnistes se

réclament de la théorie du chaos mais d'un chaos relativement déterministe. Une des explications de ce choix pour le chaos semble cependant très intéressante. [zak90] estime en effet qu'une des limitations principales des réseaux connexionnistes est leur grande rigidité au niveau comportementale, une fois l'apprentissage effectué. Cette rigidité placerait d'emblée les modèles formels bien en dessous des modèles biologiques, même les plus simples, au niveau des performances. Ce militantisme pour des modèles chaotiques se retrouve par ailleurs à un niveau plus général [hogg90], [flake93].

Les modèles de neurones ou de réseaux de neurones chaotiques découlent principalement d'une volonté de formaliser le neurone réel d'une manière neurobiologiquement plus plausible que la formalisation stricte entreprise par McCulloch et Pitts [mcculloch43]. Un des plus célèbres modèles reflétant cette volonté de complexification est le modèle de Hodgkin et Huxley [hodgkin52] mais d'autres modèles existent qui sont soit des généralisations du modèle de McCulloch et Pitts [caianiello61], soit des modèles dérivés d'observations cliniques [freeman87].

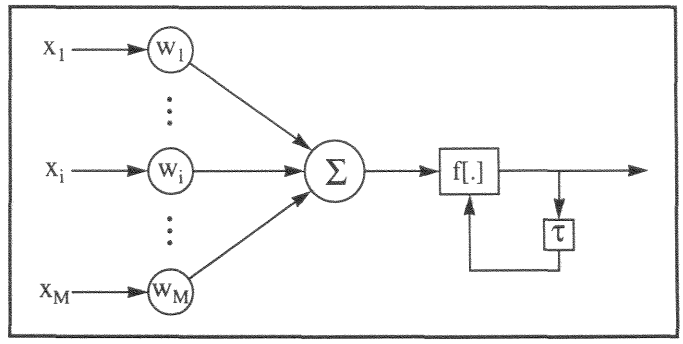


Figure 6.27 : Un neurone chaotique (d'après [dingle93], voir également l'équation 6.20).

Ainsi, [dingle93] définit un neurone chaotique au sens où son activation ne respecte pas une fonction dont la continuité est aussi simple que celle des fonctions sigmoïde ou logistique (cf. chapitre 2, paragraphe 2.2.1). Ce modèle, qui est à récurrence locale postérieure, utilise une fonction non linéaire *f* telle que définie dans l'équation 6.20. Cette équation est très proche de celle définie dans [freeman87] et deux types de comportements peuvent être obtenus selon la valeur de *net*. Si la valeur de *net* est comprise entre 0,25 et 1, le neurone aura un comportement similaire aux neurones formels tels que nous les connaissons. Si, par contre, la valeur de *net* est inférieure à 0,25 alors le neurone aura un comportement chaotique. [freeman87] avait préalablement obtenu des résultats similaires avec une fonction et des intervalles différents.

$$y(n+1) = 1 - 4(1 - net)\, y(n)\, (1 - y(n)) \qquad \text{où } net = \sum_{i=1}^{M} w_i x_i \qquad \text{(Éq. 6.20)}$$

Il sera très difficile d'interpréter la valeur d'un tel neurone pris isolément et l'auteur milite pour une mise en réseau de telles unités. Bien qu'il ne fournisse pas l'algorithme d'apprentissage associé à ces unités, il donne une règle d'initialisation aléatoire des poids w_j d'un neurone qui est fonction du nombre d'unités afférentes à ce neurone. Ce processus ne permet évidemment aucun apprentissage. Mais l'unité a été définie dans un but de modélisation de l'activité neuronale telle qu'elle peut être visualisée par électro-encéphalogramme et les résultats obtenus semblent tout à fait probants.

D'autres modèles de neurones chaotiques existent, dont certains sont à récurrence locale antérieure [aihara90], [kaneko90]. Cette dernière référence définit d'ailleurs un modèle de neurone possédant de nombreux feedbacks dont chacun possède sa propre ligne de délais. Ce modèle de neurone est très complexe et, là encore, aucune procédure d'apprentissage n'est proposée. Mais, après tout, donner une capacité d'apprentissage à tous ces modèles serait peut-être incohérent avec la philosophie dont ils se réclament...

Par ailleurs, tous les modèles formalisant une réalité neurobiologique forte ne sont pas chaotiques et certains modèles connexionnistes utilisant, par exemple, les principes du modèle de Hodgkin et Huxley ont des comportements rythmiques stables [mcauley93].

6.4.3.2/ Autoregressive Network

Le modèle de neurone autorégressif, également qualifié d'*Autoregressive Network*, a été présenté dans [leigthon91] par un des auteurs du système *Aspirin/Migraines* [leigthon92]. Ce réseau se caractérise par un mécanisme d'autorégression situé après la fonction non linéaire. Ce mécanisme permet donc de conserver les informations venant de la couche inférieure une fois qu'elles ont été traitées par le mécanisme de somme pondérée et par la fonction non linéaire. Le mécanisme d'autorégression permet ainsi de conserver les décisions prises par l'unité considérée pendant les N pas de temps précédents et de les réutiliser, par sommation, pour définir la valeur effective de l'activation de l'unité à un instant donné. Le nombre de pas de temps de cette mémoire autorégressive est prédéfini par le concepteur, selon la tâche à résoudre. Cette architecture peut être comparée avec l'architecture de la rétropropagation pour les séquences présentée au paragraphe 6.4.2.1. Cette comparaison permet de totalement appréhender la différence entre une récurrence locale à retour antérieur et une récurrence locale à retour postérieur.

Le modèle de neurone ainsi défini correspond à l'équation 6.21 composée de deux parties : une partie d'intégration neuronale (équation 6.22) et une partie autorégressive (équation 6.23). La composition bicéphale de l'architecture du neurone impose de totalement réécrire la procédure de mise à jour des poids par rétropropagation du gradient d'erreur en tenant compte des poids connexionnistes $w_{j,i}$ et des poids de l'autorégression $a_{i,n}$.

$$o_{i,t} = f(net_i(t)) + m_i(t) \qquad \text{(Éq. 6.21)}$$

$$net_i(t) = \text{biais}_i + \sum_{j=1}^{\text{nombre d'entrées (i)}} w_{j,i} o_j(t) \qquad \text{(Éq. 6.22)}$$

$$m_i(t) = \sum_{n=1}^{\text{ordre (i)}} a_{i,n} o_i(t-n) \qquad \text{(Éq. 6.23)}$$

La procédure d'apprentissage a été définie par les concepteurs [leigthon91] qui se base sur la généralisation de la rétropropagation aux réseaux récurrents faite par [pineda87]. Cette nouvelle procédure ne calculera donc plus seulement un gradient d'erreur relatif aux poids connexionnistes ($\Delta w_{j,i}$) mais calculera également un gradient d'erreur relatif aux coefficients d'autorégression ($\Delta a_{i,n}$).

Les pondérations de la mémoire autorégressive sont restreintes suivant un critère de stabilité de manière à ce que les pôles de la fonction de transfert soient situés à l'intérieur du cercle unitaire. Les tests de stabilité proposés par les auteurs suivent le critère de stabilité Routh-Hurtwitz défini dans [pandit83]. Ainsi, lorsque la fonction de feedback n'est constituée que d'un seul délai, la valeur de a_1 est limitée à l'intervalle [-1,1]. Ce critère de stabilité permet de garantir une stabilité asymptotique du mécanisme d'autorégression.

Une des applications possibles de ce neurone est la définition d'un filtre. Ce filtre reprend l'architecture générale de la figure 6.28 mais modifie la définition des entrées en les chaînant les unes aux autres. Cette modification architecturale permet d'obtenir un neurone qui n'effectue plus seulement une autorégression mais également une moyenne mobile puisque l'équation 6.21 se réécrit alors :

$$y_n = f(w_0 x_n + w_1 x_{n-1} + \ldots + w_M x_{n-M}) + a_1 y_{n-1} + a_2 y_{n-2} + \ldots + a_N y_{n-N} \qquad \text{(Éq. 6.24)}$$

Cette dernière équation se différencie d'une équation ARMA par la présence de la fonction non linéaire portant sur les valeurs de la moyenne mobile mais les auteurs précisent qu'il est possible de

retrouver une équation ARMA linéaire en choisissant correctement des coefficients modifiant la réponse de la fonction logistique qui est linéaire aux alentours de 0.

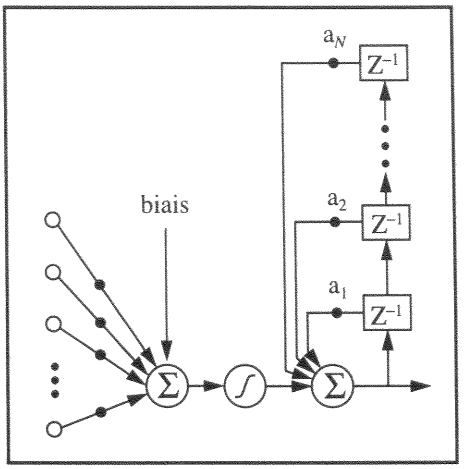


Figure 6.28 : L'architecture d'un *Autoregressive Network* (d'après [leigthon91]).

6.4.3.3/ Modèle de la Long Short Term Memory

Un des problèmes majeurs pour la mise au point des réseaux connexionnistes récurrents est la procédure d'apprentissage. Comme nous l'avons vu précédemment, certaines architectures récurrentes, telle que la machine de Boltzmann, posent des problèmes qui peuvent être considérés comme spécifiques à une classe. Cependant, deux faits majeurs, communs à tous les réseaux de neurones récurrents, permettent de comprendre un peu mieux les problèmes posés à un niveau calculatoire lors des phases d'apprentissage.

Un gradient d'erreur, rétropropagé sur plusieurs pas de temps avec une méthode inspirée de la BPTT [werbos90], peut ne plus être porteur de suffisamment d'informations pour que le réseau converge vers un état des poids satisfaisant. Deux cas peuvent être observés posant problème : le cas où les gradients deviennent trop forts (*exploding gradients*) et le cas où ceux-ci sont proches de 0 (*vanishing gradients*). Dans ces deux cas, aucune convergence de la phase d'apprentissage n'est possible. Ces problèmes sont liés aux poids du réseau, que ces poids assurent la récurrence ou non. Ils peuvent être facilement compris avec une étude simple de la fonction générale de rétropropagation du gradient d'erreur (équation 6.25).

$$\Delta w_{ij} = \eta \delta_j o_i \qquad \text{(Éq. 6.25)}$$

$$\delta_j = \begin{cases} f'_j(net_j)\,(t_j - o_j) \\ f'_j(net_j) \sum_k \delta_k w_{jk} \end{cases}$$

Au sein de cette équation, η représente le coefficient d'apprentissage et δ représente l'erreur à rétropropager. La première ligne de calcul de δ_j correspond au cas où l'erreur est calculée en couche de sortie. Cette erreur est simple à calculer puisque les valeurs de sortie effective o_j et de sortie désirée t_j sont connues. La deuxième partie de l'équation correspond au cas où le calcul de δ_j se fait en couche cachée. Dans ce cas, aucune valeur désirée n'est connue a priori et il faut donc déterminer l'erreur de manière indirecte. Cette deuxième partie de l'équation est celle qui a permis de relancer les études dans le domaine du connexionnisme, [mcculloch43], [minsky69], [lecun85]. Grâce à cette équation, le calcul local de l'erreur portée sur une unité cachée se fait grâce aux erreurs de la couche supérieure, qui sont préalablement calculées, et aux valeurs des poids synaptiques reliant l'unité considérée aux unités qui lui succèdent dans la couche supérieure. Ce mode de calcul est très efficace pour les perceptrons multicouches bien que de nombreuses heuristiques supplémentaires soient

utilisées pour faciliter la convergence. La présence de w_{jk} au sein de l'équation peut cependant entraîner des problèmes lorsque le gradient d'erreur est "descendu" sur de longues distances. Ces poids peuvent soient annihiler la valeur du gradient lorsqu'ils sont trop faibles, il s'agit alors de *vanishing gradients*, soient faire augmenter inconsidérément la valeur de l'erreur lorsqu'ils sont supérieurs à 1, il s'agit alors d'*exploding gradients*.

Pour palier à cet inconvénient majeur, qui est une limitation forte à un usage répandu et efficace des réseaux récurrents, il convient de définir et de mettre en place un mécanisme qui permet au réseau d'éviter ces problèmes pendant la phase d'apprentissage. Le meilleur de tous ces mécanismes serait une procédure d'apprentissage réellement adaptée à l'apprentissage de la récurrence mais cette procédure d'apprentissage n'existant pas, il convient de trouver un palliatif efficace. Une solution à ces problèmes d'apprentissage a été proposée dans [hochreiter95]. Cette solution est très critiquable car elle augmente énormément le nombre de connexions au sein du réseau. Elle permet néanmoins de contrôler efficacement le problème des gradients de trop forte valeur. Cette solution est inspirée du modèle sigma-pi [dehæne87] qui définit une forme plus complexe de connexion synaptique.

L'architecture d'une unité LSTM dénommée c est donnée dans la figure 6.29. Au sein de ce schéma, les poids des unités j vers l'unité i sont notés w_{ij} tandis la somme pondérée d'une unité i est notée net_i. Une unité LSTM est principalement caractérisée par la présence d'une seule récurrence non valuée, après le calcul de l'activation neuronale g, qui permet de calculer s_c, valeur d'activation autorégressive. Ce mécanisme est cependant plus simple que, par exemple, le mécanisme de l'*Autoregressive Network*. La valeur de sortie de cette récurrence, s_c, est pondérée par deux valeurs, y_{in} et y_{out}, les portes (*gating factors*). y_{in} paramètre l'entrée de la récurrence qui correspond à la valeur d'activation neuronale g tandis que y_{out} paramètre la sortie de la récurrence à l'activation h. Les valeurs y_{in} et y_{out} sont calculées à partir des valeurs d'activation d'autres neurones du réseau. La raison principale de la présence de ces valeurs est de faciliter le processus d'apprentissage en limitant la valeur des gradients puisque la fonction sigmoïde restreint la sortie à l'intervalle [0,1]. L'architecture du réseau lui-même est du type du perceptron multicouche mais toutes les unités de la couche cachée sont connectées entre elles par le biais du mécanisme des portes.

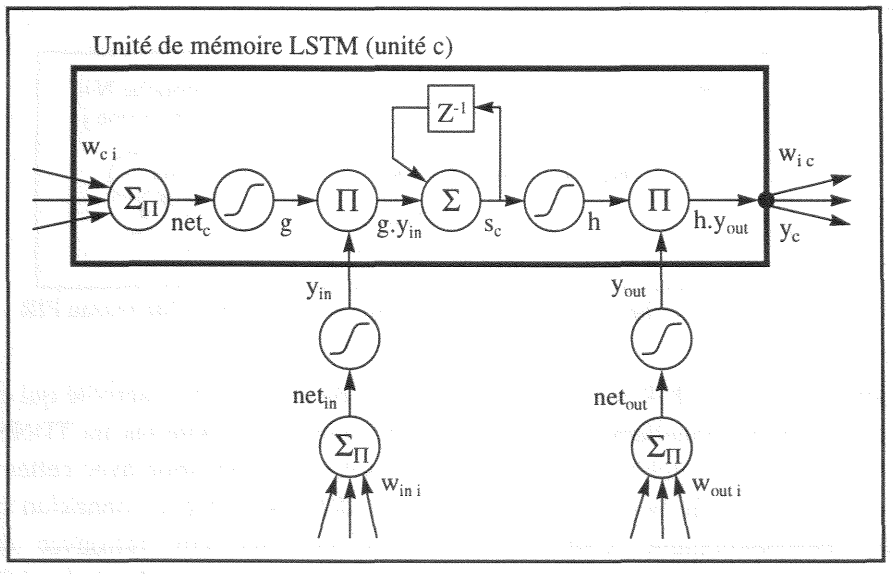


Figure 6.29 : Schéma d'une unité de mémoire LSTM (d'après [hochreiter95]).

Les seuls tests effectués sur cette architecture sont la prédiction de séquences ou d'une grammaire de Reber (voir [hochreiter95] pour quelques explications). Le test de prédiction de séquences a permis de valider très positivement cette architecture puisqu'il a été possible d'apprendre à prédire des séquences sur 100 pas de temps à l'avance, ce qu'aucune autre architecture n'a été capable de faire, et ce avec un temps d'apprentissage beaucoup plus court [hochreiter95]. Des résultats

sensiblement identiques ont été obtenus sur la tâche de prédiction d'une grammaire de Reber à un pas de temps à l'avance. Cette architecture semble donc permettre de résoudre quelques problèmes théoriques bien que les choix effectués ne soient pas généralisables.

Les concepteurs de cette architecture ont cependant proposé ultérieurement une méthode d'apprentissage qu'ils considèrent très performante, conseillant de n'employer le modèle LSTM que lorsque le nombre de poids est grand ou lorsque les poids nécessitent d'être déterminés avec une grande précision. Cette méthode, qu'ils ont baptisé *guessing* (littéralement "divination") et qu'ils limitent à des tâches d'apprentissage de séquences, consiste tout simplement à effectuer successivement plusieurs apprentissage avec un réseau dont les poids sont à chaque fois initialisés au hasard de manière différente et à conserver l'ensemble des poids qui aura eu les meilleurs résultats à la fin du processus d'apprentissage. De telles idées, qu'il est possible de rapprocher d'autres mises en œuvre avec les perceptrons multicouches [hansen90], prouvent à quel point le processus d'apprentissage dans les réseaux récurrents est encore mal maîtrisé.

6.4.3.4/ Réseaux FIR

Le réseau FIR, *Finite Impulse Response*, est un réseau à représentation interne explicite du temps au niveau des connexions selon la terminologie employée au paragraphe 6.1.2. Il a été présenté dans [back91] et [wan93]. Les connexions de ces réseaux sont en effet des filtres mémorisant les activations des neurones afférents. Toutes les étapes de mémorisation, assurée par les unités de la ligne de délais, sont fournies à l'unité de destination de la connexion synaptique et les poids présents sur la ligne sont donc spécifiques à un couple de neurones.

Cette architecture n'est pas, à proprement parler, récurrente. Les délais mis en œuvre par une connexion synaptique ne permettent en effet pas au neurone expéditeur de reprendre en compte les différentes niveaux d'activation qu'il a généré dans un passé proche. Il est cependant intéressant de considérer l'équivalence faible entre cette architecture et quelques unes des architectures que nous avons déjà présentées, tel que le modèle BPS (paragraphe 6.4.2.1), le TDNN récurrent (paragraphe 6.4.2.2) ou le modèle autorégressif (paragraphe 6.4.3.2). Au regard des architectures citées, le modèle FIR peut être considéré comme une simplification permettant de résoudre les problèmes d'apprentissage, par passage du récurrent au non récurrent.

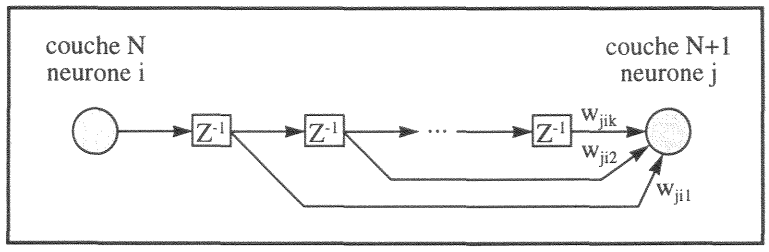


Figure 6.30 : Schéma d'une connexion entre deux neurones d'un réseau FIR (d'après [wan93]).

Le modèle de neurone FIR implante un mécanisme de rétention de l'activité qui est considéré par l'auteur comme fonctionnellement équivalent au mécanisme implanté par un TDNN [wan93]. Bien que ce type de structure architecturale ne semble pas être en adéquation avec celles présentées dans ce chapitre, le principe de la présence d'une ligne de délais sur chaque connexion synaptique a une plausibilité neurobiologique. Cette structure peut, en effet, être retrouvée dans le modèle connexionniste que nous allons voir maintenant et qui se veut très proche de la réalité chimique du cerveau.

6.4.3.5/ Modélisation de la chimie de la synapse

Le modèle FIR que nous venons d'étudier utilise une ligne de délais en lieu et place d'une simple connexion synaptique. Ce choix permet d'effectuer une première approximation du comportement de la synapse. D'autres recherches [kim92] ont tenté d'aller plus loin dans cette modélisation de la

synapse en utilisant trois concepts différents :

- Utilisation d'une modulation d'amplitude, W_{ji}. Il ne s'agit ni plus ni moins que de la connexion synaptique qui permet de prendre en compte la valeur d'une unité afférente de manière plus ou moins importante.
- Utilisation d'un délai postsynaptique, T_{ji}. Ce délai permet de modéliser un temps de rétention entre l'entrée du potentiel synaptique dans la synapse et sa mise à disposition en sortie pour le neurone de destination du potentiel. Ce délai est important puisqu'il contribue à la discrimination des différentes séquences observées en entrée [rall69].
- Utilisation d'un taux de décroissance de l'activité, D_{ji}. Ce taux de décroissance permet de faire disparaître le potentiel d'activité à un rythme prédéterminé. La valeur du taux de décroissance dépend, dans la réalité, des propriétés du soma et de la dendrite. Différents neurones peuvent, par conséquent, avoir différents taux de décroissance. La fonction de décroissance utilisée dans ce modèle a été définie par les auteurs comme étant de la forme $exp(-D_{ji}*t)$.

La fonction de calcul de la somme pondérée d'une entrée, prenant en compte les trois notions que nous venons de citer, est définie dans le cas continu comme suit dans l'équation 6.26.

$$S_{ji}(t+\Delta t) = I_{ji}(t+\Delta t - T_{ji})\, W_{ji} + S_{ji}(t)\, e^{-D_{ji}\Delta t} \qquad \text{(Éq. 6.26)}$$

L'utilisation simultanée de ces trois concepts permet donc d'étendre plus encore les mécanismes de gestion du potentiel synaptique par rapport à ceux qui sont utilisés dans le modèle FIR. Mais ces concepts ne sont pas encore suffisant pour décrire l'ensemble du modèle. Les auteurs ont également choisi d'adopter une fonction de transfert particulière. Le calcul de la valeur de sortie du neurone fait appel à la fonction de Heaviside que nous avons déjà mentionnée au chapitre 2, paragraphe 2.2.1. Mais l'application de cette fonction non linéaire ne s'effectue pas directement sur la somme pondérée de l'équation 6.26. Le mécanisme de sommation général, permettant de connaître l'état du potentiel interne à la cellule, prend en compte deux paramètre distincts pour le calcul de la sortie O_i de la cellule, sortie dont la valeur dépend du temps.

La valeur de O_i est obtenue par sommation de deux termes avant transformation par la fonction non linéaire. Le premier terme correspond à la somme des valeurs des différents potentiels synaptiques, comme pour tout neurone formel. Le deuxième terme est un biais dont la valeur sera modifiée au cours du traitement. Il s'agit donc d'un biais temporel. La formule de calcul de la valeur de sortie est donnée dans l'équation 6.27 où f est la fonction de Heaviside [kim92]. Le biais temporel correspond au terme noté H_i.

$$O_i(t+\Delta t) = f\left(\sum_j S_{ji}(t+\Delta t) - H_i(t+\Delta t)\right) \qquad \text{(Éq. 6.27)}$$

Le biais temporel H_i est lui-même calculé par sommation de sa valeur à l'instant t et d'un gradient, ce qui conduit à l'équation 6.28 suivante :

$$H_i(t+\Delta t) = H_i(t) + \Delta H_i(\Delta t) \qquad \text{(Éq. 6.28)}$$

La formule de calcul du gradient, donnée par l'équation 6.29, est une fonction du temps qui permet d'obtenir une fonction dont la courbe suit approximativement celle montrée à la figure 6.31.

$$\Delta H(t) = AmpH\left(\frac{1}{1+e^{-\alpha t}}\right) \qquad \text{(Éq. 6.29)}$$

Dans l'équation 6.29, *Amp* est une constante définissant la valeur de changement maximum du gradient tandis que α correspond à la constante définissant la longueur de la période réfractaire.

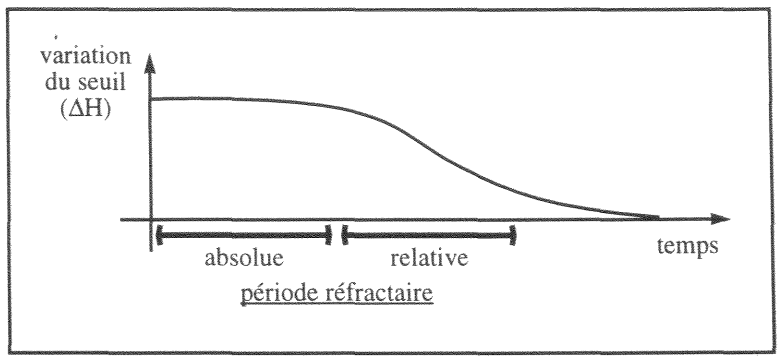


Figure 6.31 : Courbe de la fonction du biais temporel (d'après [kim92]).

Toutes les caractéristiques que nous venons d'énumérer permettent de définir une unité neuronale d'intégration temporelle dont nous donnons le schéma à la figure 6.32.

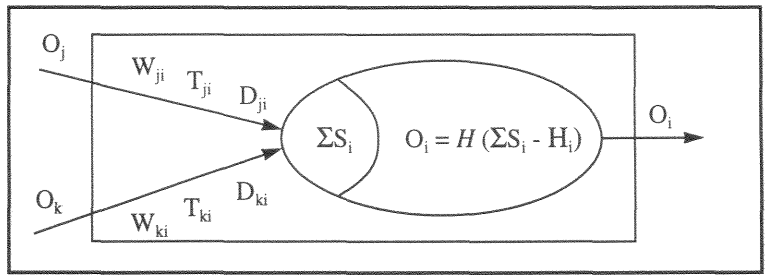


Figure 6.32 : Schéma d'un neurone d'intégration temporelle (d'après [kim92]).

Les expériences menées avec cette architecture n'ont pas été nombreuses malgré l'intérêt potentiel de cette unité connexionniste. La seule véritable expérience rapportée dans [kim92] concerne la reconnaissance de séquences binaires de trois éléments. Le problème posé par cette architecture est en fait son très grand nombre de paramètres qu'il faudrait définir par apprentissage. Ainsi, bien que les auteurs aient défini manuellement les valeurs des différents coefficients de délai, de décroissance synaptique et de décroissance neuronal du réseau, il semble évident qu'un problème complexe ne pourra être résolu qu'avec une phase d'apprentissage définissant tous les paramètres. Or cette méthode d'apprentissage est pour l'instant inexistante et sa mise au point sera sans doute fort complexe du fait du grand nombre d'interaction au sein d'une même unité. [kim92] entrevoit cependant une possibilité d'application très intéressante de ces neurones par fusion des mécanisme qui viennent d'être présentés avec le paradigme de carte auto-organisée [kohonen88].

6.4.3.6/ Le modèle de neurone à hystérésis

La mise en place d'une mémoire à court terme peut passer par des mécanismes encore plus complexes que ceux que nous venons de présenter. Ainsi, [tom95] présente une architecture utilisant un mécanisme d'hystérésis de manière à obtenir un système présentant, d'après les auteurs, des caractéristiques intéressantes pour la modélisation connexionniste :

- sensibilité et accoutumance aux phénomènes,
- nouvelle forme d'apprentissage non associatif,
- capacités de différenciation de formes spatio-temporelles présentées en ambiance bruitée,
- agrégation de séquences de longueur arbitraire dans une seule réponse,
- délai adaptable à la longueur temporelle de la forme.

L'hystérésis se modélise grâce à deux ensembles de fonctions : le premier ensemble regroupe la famille des fonctions ascendantes tandis que le deuxième ensemble regroupe la famille des fonctions descendantes. L'écart séparant le passage par zéro sur l'axe des x de chacune de ces courbes est noté H_c. Les deux familles de fonctions peuvent être modélisées par des fonctions hyperboliques du même type de celles qui sont utilisées pour définir les fonctions sigmoïdales. L'hystérésis est utilisé

en fonction de la variation de x. Lorsque x croît, y croît également selon la définition de l'ascendance donnée par la fonction ascendante. À l'inverse, lorsque x décroît, y décroît en suivant la fonction descendante.

Les fonctions ascendantes sont définies selon la formule donnée par l'équation 6.30. Cette équation définit en fait toute une famille de fonction selon la valeur de η.

$$y = \eta + (1 - \eta) \tanh (x - H_c) \qquad \text{(Éq. 6.30)}$$

η est lui-même définie par la résolution de l'équation 6.31 suivante :

$$y_0 = \eta + (1 - \eta) \tanh (x_0 - H_c) \qquad \text{(Éq. 6.31)}$$

qui permet de calculer la valeur de η par la formule de l'équation 6.32.

$$\eta = \frac{y_0 - \tanh (x_0 - H_c)}{1 - \tanh (x_0 - H_c)} \qquad \text{(Éq. 6.32)}$$

Les fonctions descendantes sont également définies selon l'équation 6.30 mais le calcul de η, nécessaire au calcul de la descente, est définie selon la formule de l'équation 6.33 suivante :

$$y_0 = -\eta + (1 - \eta) \tanh (x_0 + H_c) \qquad \text{(Éq. 6.33)}$$

La résolution de cette équation permet de calculer la valeur de η selon la formule :

$$\eta = \frac{y_0 - \tanh (x_0 + H_c)}{-1 - \tanh (x_0 + H_c)} \qquad \text{(Éq. 6.34)}$$

Le comportement des deux familles de courbes est schématisé selon dans la figure 6.33 qui permet de mieux appréhender le comportement du neurone à hystérésis.

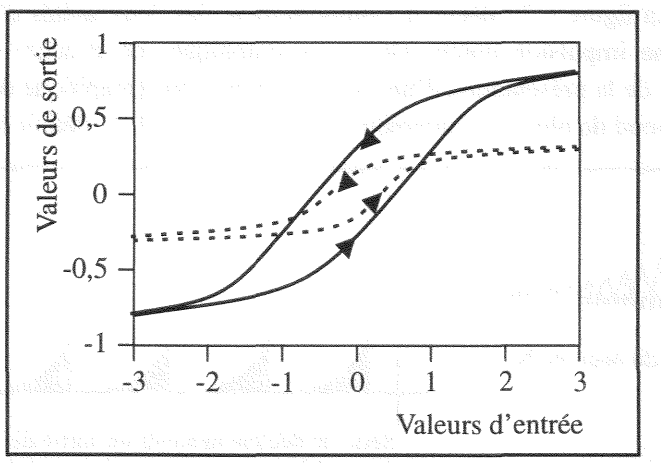


Figure 6.33 : Schémas de graphes de 2 fonctions ascendantes et de 2 fonctions descendantes du modèle *Hystery* (d'après [tom95]).

Il est possible d'appliquer cette architecture neuronale à des problèmes de classification temporelle. [tom95] étudie ainsi la différenciation par précédence temporelle et la classification de séquences mais les exemples étudiés sont, là encore, assez simplistes et ne permettent pas de juger de la qualité de la modélisation de problèmes complexes.

Le mécanisme de l'hystérésis n'en est pas moins intéressant puisqu'il permet de trouver une fonction qui, si elle suit la forme générale des fonctions sigmoïdales ou stochastiques (cf. chapitre 2, paragraphe 2.2.1), définit une méthode de traitement du potentiel neuronal qui est fonction de la variation du potentiel synaptique et donc, dans une certaine mesure, du temps. Ce type de travaux permet de définir des courbes qui vont au delà des études réalisées dans [amari92] et [geva92]. En

outre, ce type de mécanisme a par ailleurs été mis en œuvre avec succès dans de réseaux de Hopfield [derou94], prouvant que ce concept est potentiellement très intéressant.

6.4.3.7/ Neurones duaux

Les neurones duaux ne sont pas des extensions des réseaux duaux [azencott92c] dont nous avons parlé précédemment (paragraphe 6.2.4). Ils ont été définis dans [wang90] dans le but de réaliser un système connexionniste capable d'apprendre, de reconnaître et de reproduire des séquences temporelles.

L'architecture des neurones duaux permet d'obtenir une décroissance exponentielle de l'activité par simple passage du potentiel d'activité d'un neurone à l'autre. Un neurone dual est en fait composé de deux unités (figure 6.34) : une première unité, le neurone principal, reçoit l'information en entrée avant de la restituer en sortie tandis que la deuxième unité permet d'effectuer une mémorisation en arrière plan de l'information d'entrée avant de la restituer au neurone principal. Ce type d'architecture, avec les notions d'avant et d'arrière plan, peut être rapproché du modèle développé par [tino95] et présenté à la figure 6.21.

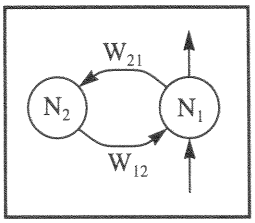


Figure 6.34 : Schéma d'un neurone dual (d'après [wang90]).

Ce mécanisme de boucle assure une mémorisation pendant une période plus ou moins longue selon les valeurs des poids W_{21} et W_{12}. Ce mécanisme de décroissance mis en place, un neurone dual pourra adopter un comportement simulant un déchargement de l'activité identique à ceux des neurones réels. La figure 6.35 décrit le comportement des deux unités d'un neurone dual après la présentation d'une impulsion unique (les deux graphiques de gauche) et le comportement d'un neurone dual lors de la présentation d'une série d'impulsions (graphique de droite). La décroissance exponentielle dépend de plusieurs paramètres qui permettent de la définir finement [wang90].

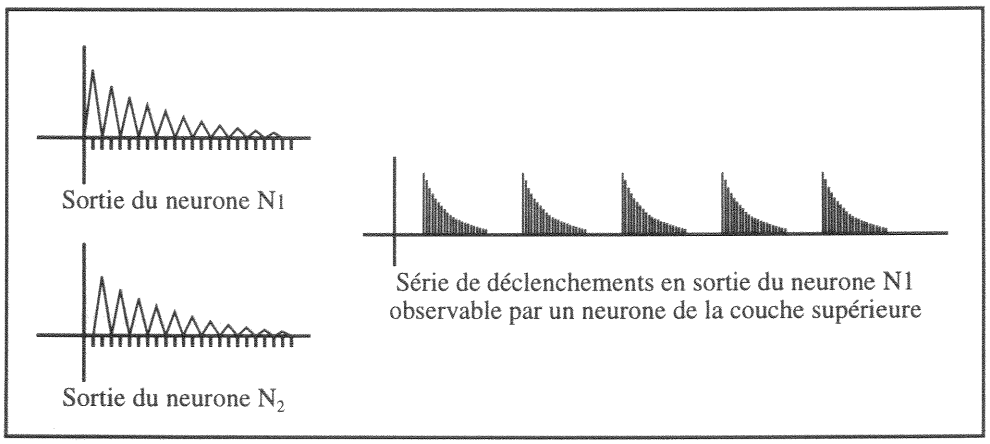


Figure 6.35 : Réponses type des neurones duaux après la présentation d'une impulsion (à gauche) et séquence de réponses du neurone dual (à droite) (d'après [wang90]).

Les neurones duaux peuvent être aussi bien utilisés pour la production que pour la reconnaissance de séquences. Les architectures à mettre en place pour ces deux types d'applications sont bien sûr différentes. Nous allons nous attarder sur l'architecture type de reconnaissance de séquences présentée dans [wang90] pour les différents concepts qu'elle met en œuvre. Le réseau de neurones dédié à la reconnaissance de séquences et présenté dans la figure 6.36 possède trois niveaux ayant chacun un rôle bien défini. La couche d'entrée assure la mémorisation des différents éléments de la

séquence, par décroissance exponentielle plus ou moins rapide de l'activité ; c'est donc ici que sont implantés les neurones duaux. Tous les neurones de cette première couche sont connectés aux neurones de la couche intermédiaire qui effectue la reconnaissance des séquences en émettant des valeurs pouvant être assimilées à des probabilités d'occurrence de la séquence représentée. Chaque séquence est représentée par un neurone dont la valeur est déterminée en fonction des valeurs des neurones de la couche d'entrée. Les neurones de cette deuxième couche sont eux-même connectés à autant de neurones de la couche de sortie, chacun de ces derniers neurones ne recevant de valeur que d'un seul neurone de la couche de reconnaissance des séquences. Les neurones de la couche de sortie sont tous reliés les uns aux autres mais les connexions ne sont pas ici véritablement récurrentes. Ces connexions permettent, en effet, de mettre en place un mécanisme de compétition entre les différents neurones de la couche de sortie.

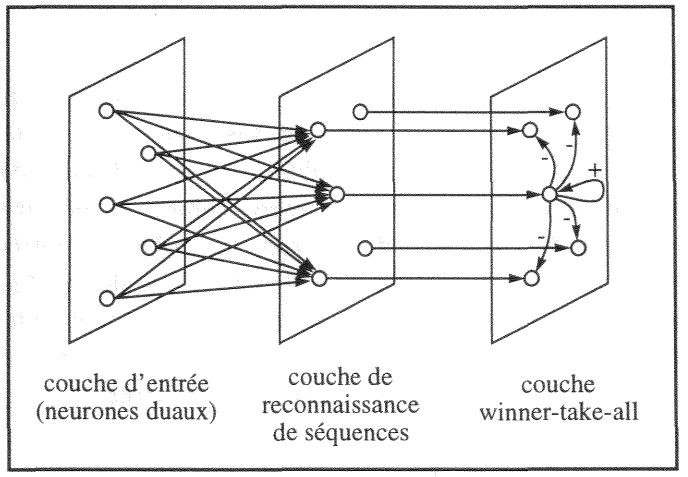


Figure 6.36 : Architecture type pour une tâche de reconnaissance de séquences temporelles (d'après [wang90]).

Le réseau de la figure 6.36 permet donc d'effectuer la reconnaissance de séquences en mettant trois mécanismes distincts :

- conservation de l'activité à plus ou moins long terme par mise en place d'une récurrence locale dans les neurones de la couche d'entrée,
- intégration de la connaissance temporelle à un niveau architectural distinct du niveau de la conservation et où chaque neurone représente un type de séquence possible,
- sélection de la séquence de plus forte probabilité à un niveau décisionnaire mettant en œuvre un mécanisme compétitif intégrant une connaissance de la probabilité d'apparition de chaque séquence, cette probabilité étant déterminée relativement au corpus d'apprentissage.

Les trois mécanismes que nous venons d'énumérer sont tous trois important dans des tâches de classification de séquences. Le mécanisme de sélection par compétition permet d'obtenir une classification qui reflète les données observées lors de la phase d'apprentissage. Mais le mécanisme de rétention de l'information par unités modélisant une décroissance exponentielle ne permet pas de traiter tous les types de séquences possibles. En effet, si une décroissance peut s'avérer nécessaire pour la prise en compte d'une information contextuelle dont l'antériorité est plus ou moins forte, elle ne permet pas de modéliser toutes les grammaires qui peuvent être rencontrées dans ce type de problème, au rang desquelles se trouvent les grammaires à contexte libre (cf. paragraphe 6.3.7 et paragraphe 6.3.8).

6.4.3.8/ Réseaux de neurones à mémoire

Les réseaux de neurones à mémoire ont été présentés dans [sastry94]. Ils possèdent une architecture assez proche de celle du neurone défini dans [wang90] quoique ce dernier soit plus simple (paragraphe 6.4.3.7). L'idée de base est ici de définir deux types de neurones distincts. Un

premier type assure, comme dans tous les réseaux de neurones, l'intégration des informations venant de la couche précédente. Ces neurones sont appelés neurones de réseau. Un deuxième type assure la mémorisation des informations traitées par le neurone de réseau. Ce type de neurone, le neurone de mémorisation, ne reçoit que deux entrées : une première entrée provient du neurone de réseau qui est associé au neurone de mémoire. Cette première entrée est pondérée par un coefficient α. La deuxième entrée permet de tenir compte de la valeur de sortie du neurone de mémorisation au pas de temps précédent, par le biais d'un feedback qui est lui aussi pondéré par un coefficient de valeur $1-\alpha$. Ce neurone de mémorisation utilise par ailleurs deux unités de délais, une première pour conserver la valeur de sortie du neurone de réseau, qui ne sera donc prise en compte qu'au pas de temps suivant, et une deuxième pour conserver sa valeur au pas de temps précédent. Le mécanisme de prise en compte des informations par le neurone de mémorisation est défini par l' équation 6.35 suivante :

$$z_i(t) = \alpha y(t-1) + (1-\alpha)\, z_i(t-1) \qquad \text{(Éq. 6.35)}$$

L'association des deux types de neurones permet d'obtenir une unité du réseau. Le neurone de réseau et le neurone de mémorisation constitue ainsi une seule unité composite avec deux mécanismes successifs et avec deux sorties visibles, une pour chaque neurone élémentaire. Cette unité composite peut même être définie avec plusieurs neurones de mémorisation successifs comme le montre la dernière couche du réseau de la figure 6.37 (d'après [sastry94]), l'article précisant cependant par ailleurs que cette possibilité est restreinte à la seule couche d'entrée du réseau. Les sorties des neurones constituent donc autant de sources d'information pour les neurones de réseaux de la couche en aval du neurone composite considéré. La prise en compte des potentiels d'activités par un neurone d'une couche est défini par l'équation 6.36 suivante :

$$y(t) = f\left(\sum_{i=0}^{n} w_i x_i(t) + \sum_{i=0}^{n} k_i z_i(t) \right) \qquad \text{(Éq. 6.36)}$$

L'unité composite, en plus des poids connexionnistes qui permettent de synthétiser les informations dans le neurone de réseau, utilise deux pondérations liées pour définir la valeur du neurone de mémorisation. Une première pondération permet de modifier la valeur de sortie du neurone de réseau avec une valeur α avant que l'activité soit stockée un premier délai. La deuxième pondération permet, elle, de modifier la valeur de sortie du neurone de mémorisation par $1-\alpha$ avant que cette valeur ne soit stockée dans un deuxième délai. Ces deux valeurs constituent les seuls poids du neurone de mémorisation.

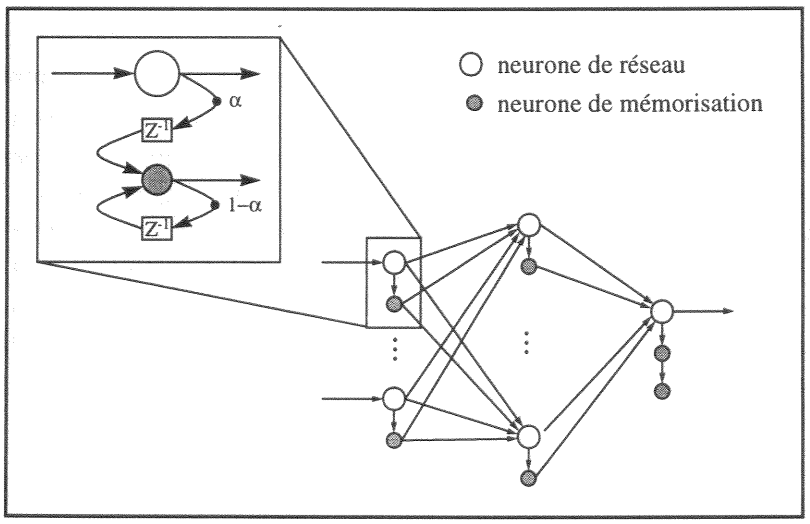


Figure 6.37 : Un réseau de neurones à mémoire (d'après [sastry94]).

L'apprentissage dans ces réseaux peut être effectué grâce à la méthode de l'apprentissage temps

réel récurrent (RTRL) ou grâce à des méthodes se voulant plus proches des méthodes de la théorie du contrôle des systèmes dynamiques comme dans [narenda90] par exemple. Les exemples d'apprentissage présenté dans [sastry94] concerne tous des systèmes dynamiques à une entrée et une sortie (*Single Input Single Output, SISO*) ce qui ne permet pas de véritablement appréhender les capacités de modèle pour des tâches plus complexes. De plus, les comparaisons fournies entre un réseau composé d'une couche cachée de trois neurones de réseau et un réseau composé d'une couche cachée de six neurones de réseau possédant chacun un neurone de mémoire ne permet pas d'observer une réelle différence au niveau des capacités de modélisation d'une tâche de type *SISO*. Il est donc tout à fait naturel de se demander si le réseau ne profite pas du décloisonnement entre les neurones effectuant la synthèse et ceux effectuant la mémorisation pour totalement occulter ces derniers.

Il est par ailleurs intéressant d'étudier la proximité des concepts mis en œuvre dans ce réseau avec ceux d'autres réseaux dont l'architecture est initialement très différente [harrison89], [vries90] ou même [feng91]. Nous allons maintenant exposer l'une d'entre elles.

6.5/ Neurone gamma

Le dernier modèle dont nous parlerons dans ce chapitre est le modèle qui nous a plus particulièrement intéressé lors de la deuxième partie de cette thèse. Cette architecture a pour la première fois été présentée dans [vries90].

Le modèle de neurone gamma correspond en fait à un filtre permettant de modéliser une décroissance plus ou moins rapide de l'activité dans les modèles connexionnistes. Sa première utilisation est en fait centrée sur la définition d'une unité apte à modéliser des signaux selon une profondeur variable. Ce problème n'est pas, a priori, en relation directe avec la représentation de la décroissance exponentielle mais le mécanisme développé par le modèle gamma permet tout autant de modéliser cette décroissance que de représenter des signaux de manière plus ou moins fine et à plus ou moins long terme.

Ainsi, le mécanisme du TDNN, qui possède une taille de mémoire finie dans sa plaque d'entrée tout autant que dans ses couches cachées, peut être amélioré par l'emploi des filtres gamma dans la plaque d'entrée, ces filtres permettant de remplacer tout autre paradigme de mémorisation. Le mécanisme ressemblant à une sorte de fenêtre glissante et reposant sur le principe des poids partagés devient alors inutile puisque le besoin de spatialisation est remplacé par un filtrage du signal dans la couche d'entrée. Un seul jeu de poids devient alors suffisant pour assurer le traitement et le paradigme de partage des poids peut être abandonné.

Les capacités du modèle gamma et les extensions qui y ont été apportées sont présentées dans le chapitre suivant.

CHAPITRE 7 : MISE EN ŒUVRE DES RÉSEAUX GAMMA

"La fin suggère les moyens"
George Polya
Comment poser et résoudre un problème

"The ability to reduce everything to simple fundamental laws does not imply an ability to start from these laws and reconstruct the universe"
P. Anderson

Résumé

Ce chapitre présente l'architecture gamma et les extensions que nous avons jugé bon d'y apporter vis-à-vis de nos connaissances en reconnaissance automatique de la parole et du problème que nous avions à résoudre. Ce chapitre s'attachera à énoncer les caractéristiques principales du modèle gamma et ses relations avec les systèmes dynamiques non linéaires ainsi que tous les problèmes d'apprentissage qui en découlent. Nous présenterons ensuite les extensions architecturales apportées au modèle. Deux séries de tests sont enfin présentées qui permettent de juger des qualités et des capacités d'un réseau connexionniste utilisant des filtres gamma, tant pour la reconnaissance de diverses séquences temporelles que pour la segmentation de la parole selon différentes classifications.

7.1/ Réseaux gamma

7.1.1/ Présentation

Le modèle gamma est une architecture assez récente bien qu'il puisse être très clairement apparenté à certains des modèles que nous venons de voir. Ainsi, les développements que nous allons entreprendre peuvent être facilement rapprochés de l'*Autoregressive Network* (cf. chapitre 6, paragraphe 6.4.3.2) bien qu'il soit plus complexe et que les contraintes imposées à ce modèle soient d'un autre ordre que celles imposées au modèle gamma. Le type d'architecture gamma permet de mettre en œuvre un réseau possédant deux types de mémoire du fait, d'une part, de l'organisation du réseau qui ressemble à un perceptron multicouche lorsqu'il est considéré à un niveau général, et, d'autre part, de l'organisation du neurone, qui ne respecte pas la définition habituelle donnée par [mcculloch43].

Il est donc possible de considérer qu'il existe deux mémoires dont l'une est statique et située au niveau des connexions entre les neurones. Cette mémoire permet de conserver la connaissance définie lors de la phase d'apprentissage et ne variera normalement pas en cours d'utilisation

quoiqu'il soit possible d'envisager une telle possibilité comme étant très intéressante. Une deuxième mémoire, dynamique, est implantée dans le réseau au niveau des récurrences locales de certains neurones. Cette récurrence va permettre de conserver l'information pendant un certain temps, modélisant ainsi la dynamique de l'analyse du signal et, d'une certaine manière, l'état courant du système observé.

Notre problème nous pousse à utiliser un système capable de modéliser à la fois des connaissances phonétiques instantanées, reposant sur les vecteurs d'entrée du réseau, et des connaissances phonétiques temporelles, reposant sur la durée antérieure de certains phénomènes. Nous avons donc décidé d'utiliser le mécanisme gamma comme base de notre architecture du fait de ses capacités et des connaissances mathématiques qui s'y rapportent.

7.1.2/ Architecture

Le modèle gamma part d'une idée simple pour tenter de modéliser la décroissance exponentielle de l'activité. Un filtre est défini selon l'équation 7.1 (d'après [vries90]). Dans ce filtre, un seul paramètre, μ, contrôle l'ensemble du comportement et permet d'obtenir différents types de réponses de la part du neurone selon l'intervalle dans lequel se trouve ce paramètre.

$$y_{i,t} = \mu y_{i-1,t} + (1-\mu)\, y_{i,t-1} \qquad \text{(Éq. 7.1)}$$

Ce paramètre de feedback, tout comme les paramètres des *Autoregresive Networks*, est limité sur un intervalle dans lequel la stabilité du filtre est démontrée. Mais contrairement aux $a_{i,j}$ de l'équation 6.23 du chapitre 6 concernant les *Autoregressive Networks*, les valeurs possibles du coefficient de régression sont, cette fois, limitées à l'intervalle]0,2[. Au sein de cet intervalle, le filtre pourra avoir plusieurs comportement en fonction de la valeur de μ. Ainsi, si μ est égal à 1, la deuxième partie de l'équation disparaît et le filtre est alors équivalent à un délai simple (cf. chapitre 6, figure 6.2.a). Les deux cas restants sont également très intéressants. Dans le cas où μ est compris entre 0 et 1, bornes exclues, le filtre gamma se comporte comme un filtre passe-bas. Ce cas permet de simuler une mémoire à décroissance exponentielle [principe93a] et c'est celui qui nous intéressera : seul l'intervalle]0,1] a, en effet, été étudié au cours de cette thèse. Le cas restant, cas où μ est compris dans l'intervalle]1,2[, permet au filtre de se comporter comme un filtre passe-haut.

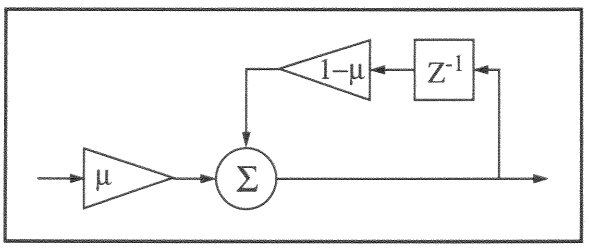


Figure 7.1 : Le schéma d'un filtre gamma (d'après [vries91a]).

Le filtre gamma, tel qu'il est défini par l'équation 7.1 et par la figure 7.1, n'est pas à proprement parler un neurone. Il s'agit d'un simple filtre dont l'utilisation première n'est pas pleinement de nature connexionniste. Il permet principalement de modéliser un système dynamique (paragraphe 7.2.1) bien que sa simplicité restreigne ses capacités modélisatrices. Ce filtre peut être utilisé comme mécanisme de mémorisation au sein d'un réseau de type *Autoregressive Network*, ce qui simplifierait l'équation et l'apprentissage dans ce dernier modèle. Mais le filtre gamma a tout d'abord été mis au point pour traiter et mémoriser (paragraphe 7.2.2) le signal à un bas niveau. Il est donc destiné, en tout premier lieu, à améliorer les capacités des *Time Delay Neural Networks* (paragraphe 7.2.3), ces réseaux étant très bien adaptés à la parole et largement répandus dans le domaine de la RAP. Le mécanisme du filtre gamma permet d'effectuer des traitements qui sont impossibles à réaliser avec un TDNN sans une étape de prétraitement supplémentaire (paragraphe 7.2.4). À ce titre, une architecture particulière a été développée pour étudier les capacités des filtres dans un contexte

limité. Cette architecture, le modèle *Focused Gamma Network* (figure 7.2) limite la présence des filtres gamma à la plaque d'entrée pour tenter d'obtenir de meilleurs résultats que ceux obtenus avec un TDNN.

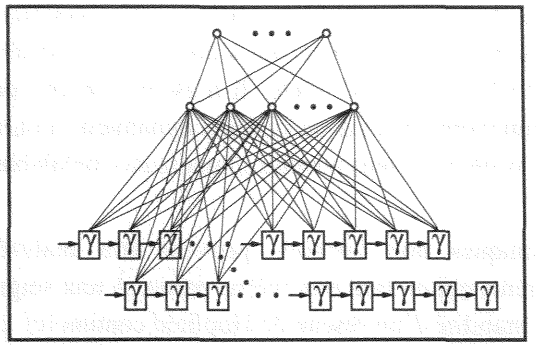


Figure 7.2 : Architecture du *Focused Gamma Network* (voir également la figure 7.5 d'après [principe93b]).

Avant toute étude des capacités théoriques du modèle gamma, il est important et intéressant d'étudier la manière par laquelle les performances maximales pourraient être atteintes avec un tel mécanisme. Les paramètres des filtres gamma permettent de modéliser le signal de manière théoriquement idéale par rapport à la tâche étudiée. Mais ces paramètres doivent, au préalable, être déterminés par apprentissage. Or la procédure d'apprentissage n'est pas véritablement définie et plusieurs méthodes ont été étudiées (paragraphe 7.2.5). Le filtre gamma peut également être analysé comme faisant partie d'une petite famille de modèles tentant de modéliser des systèmes dynamiques au rang desquels se trouve le signal de parole (paragraphe 7.2.6).

7.2/ Connaissance théorique du modèle gamma

7.2.1/ Les systèmes dynamiques

7.2.1.1/ Caractérisation des systèmes dynamiques

Un système dynamique autonome d'ordre d peut être représenté par un ensemble d'équations différentielles, ces équations étant celles des états du système représentés par $x(t)$. Ces équations différentielles caractérisent le système en temps continu et peuvent être modifiées pour modéliser ce même système en temps discret avant d'être utilisées informatiquement. La fonction f est la fonction de transfert, *vector field*, du système qui permet d'obtenir, à partir du vecteur des états du système, le vecteur des états à l'instant suivant.

$$\frac{d}{dt}x(t) = f(x(t)) \qquad \text{(Éq. 7.2)}$$

Tout système dynamique non linéaire peut être caractérisé par deux variables : la dimension du système et les exposants de Lyapounov qui y sont associés. La dimension du système correspond en fait au nombre de degrés de liberté du système, notion qui a été définie par [whitney36], et qui est aujourd'hui qualifiée de dimension d'enchâssement, ou *embedding dimension*, [sauer91], [casdagli91], [abarbanel93]. Cette dimension d permet d'étudier un système spécifique sur $\Re^d$ même si ces dimensions ne sont pas directement perceptibles dans un signal. Les autres valeurs permettant de caractériser un système sont les exposants de Lyapounov de la série temporelle constituée par les données du système observé [eckmann85], [eckmann86]. Plusieurs exposants de Lyapounov peuvent être calculés sur une série. La valeur de ces exposants donne une mesure du taux de création de l'information pour chacun des degrés de liberté du système. Lorsque λ, l'exposant de Lyapounov, est nul, c'est à dire lorsque e^{λ} égal 1, le taux de création de l'information est nul. Un système à temps continu possède toujours au moins un exposant de Lyapounov égal à 0 [casdagli91]. Un exposant inférieur à 0 permet de savoir que l'information disparaît pour le degré considéré. Enfin, si au moins

un des exposants de Lyapounov est supérieur à 0 alors le système dynamique observé peut être considéré comme chaotique.

Toutes les recherches menées actuellement ne permettent cependant pas de totalement appréhender les systèmes dynamiques. La caractérisation d'un système, grâce à sa dimension et à ses exposants de Lyapounov, permet de le rattacher à une classe de systèmes mais ne permet pas de le modéliser formellement d'une manière fidèle. Il n'est pas encore possible, à l'heure actuelle, de retrouver les règles de fonctionnement d'un système dynamique à partir de la simple observation du processus généré par ce système et sans aucune connaissance préalable des règles de fonctionnement qui lui sont propres.

Ces systèmes dynamiques non linéaires peuvent être analysés par le biais des réseaux neuromimétiques récurrents qui en font eux-même partie. À tout seigneur, tout honneur, [marcus91] étudie les problèmes de stabilité d'un réseau de Hopfield continu (cf. chapitre 6, paragraphe 6.2.1) et fait remarquer l'importance du gain de la fonction non linéaire des neurones pour la stabilité générale du réseau, prouvant ainsi l'inadaptation de la fonction de Heaviside (cf. chapitre 2, figure 2.7). [zbikowski94] et [sjoberg95] étudient les liens existants entre les réseaux connexionnistes récurrents et l'état actuel de la théorie du contrôle. Enfin, [kolen94] tente de schématiser ces réseaux sous forme de systèmes de fonctions itérées, *iterated function systems*, et étudie, entre autres choses, le problème de l'exploration de l'ensemble de l'espace de représentation par le réseau et le problème de la formation de *clusters*.

Les réseaux connexionnistes permettent, par ailleurs, de tester les notions théoriques relatives à la reconstruction de la dynamique d'un système, tant pour la théorie du contrôle que pour la simple modélisation.

7.2.1.2/ Reconstruction de la dynamique d'un système

La reconstruction de la dynamique d'un système est un problème ardu. Ce problème peut cependant être résolu dans certains cas. Un système dont la dynamique est linéaire pourra, ainsi, être assez facilement modélisé à l'aide d'une moyenne mobile ou d'un mécanisme autorégressif qui sont regroupés sous le qualificatif d'ARMA, *Auto Regressive Moving Average*. Il en est tout autrement pour un système dont la dynamique est non linéaire et qui fait donc intervenir une fonction supplémentaire, non linéaire, du type de celles que nous avons présentées au chapitre 2, paragraphe 2.2.1. La théorie développée à ce sujet est balbutiante et ne permet pas de trouver un modèle pour tout système étudié.

Certaines recherches essaient cependant de donner une méthodologie à la modélisation. [takens81] propose ainsi un cadre général pour modéliser un système dynamique non linéaire déterministe à partir d'observations. Il s'intéresse plus particulièrement au problème de la prédiction en fonction d'observations du système à modéliser. Or la classification peut être considérée comme un cas particulier de prédiction.

Soit $\{x_1, x_2, \ldots x_n\}$ une série de valeurs issues d'un système dynamique déterministe à partir desquelles il faut prédire $\{x_{n+1}, x_{n+2}, \ldots\}$, Takens montre que toute prédiction peut être effectuée avec une fonction f non linéaire, la valeur d de la dimension intrinsèque du système et une valeur τ d'un délai arbitraire selon la formule :

$$x_t = f(x_{t-\tau}, x_{t-2\tau}, \ldots, x_{t-d\tau}) \qquad \text{(Éq. 7.3)}$$

[takens81] prouve ainsi que tout système dynamique non linéaire déterministe peut être modélisé fidèlement. Malheureusement, cette équation et la démonstration associée sont purement théoriques et s'il est possible de retrouver la dimension d'un système à partir d'observations, il est, encore aujourd'hui, impossible de déterminer le paramètre τ nécessaire à la mise en œuvre de cette équation bien que certains aient développé plus avant la théorie de Takens [fraser86], [casdagli89], [sauer91] et que d'autres aient déjà découvert des heuristiques spécifiques à certains domaines d'études

[abarbanel93]. Ces heuristiques pourraient cependant être dangereuses si elles étaient de trop mauvaise qualité car, comme le remarque [bengio94a], un attracteur périodique peut être transformé en point fixe par simple sous échantillonnage du signal à une fréquence égale à la période de l'attracteur. Ceci signifie que le choix d'une mauvaise période d'échantillonnage peut totalement faire disparaître à la vue de l'algorithme d'approximation l'ensemble des variations permettant d'observer au moins un des degrés de liberté du système, rendant la modélisation incorrecte.

Le problème étudié ici doit être considéré comme un développement très important d'autres études ayant eu, dans le passé, une grande importance en parole comme le codage prédictif linéaire de la parole déjà présentée au chapitre 1, paragraphe 1.7.2.2. Cependant, il s'agit cette fois de ne plus se contenter d'une moyenne mobile et d'un mécanisme autorégressif pour modéliser et donc pouvoir prédire le signal observé mais d'inclure les capacités des fonctions non linéaires pour approcher plus encore la réalité. Il est, de plus, souhaitable de supprimer les termes d'erreur qui étaient la règle pour le codage prédictif linéaire (cf. chapitre 1, équation 1.1) et qui ont justifié l'apparition de la méthode RELP qui considérait cette erreur comme faisant partie de la modélisation.

Ce domaine de recherche est donc d'une extrême importance du fait des changements dont il est porteur. Mais il faudra, avant tout, pouvoir déterminer le nombre de dimensions du système et, surtout, la fréquence d'observation du signal qui serait fonction de τ et ne serait plus simplement égale à 1 comme, par exemple, pour la méthode LPC. Cette utilisation d'un coefficient τ permet de mettre en œuvre un horizon d'observation plus adapté au signal mais nécessitera, en contrepartie, que cet horizon soit, au préalable, défini par la théorie ou, à défaut, par une heuristique. Cet horizon d'observation est, pour l'instant, déterminé de manière totalement heuristique.

7.2.2/ Horizon temporel de la plaque d'entrée

7.2.2.1/ Plaque d'entrée à horizon fixe

Un des modèles neuromimétiques ayant aujourd'hui le plus de succès dans le domaine de la reconnaissance automatique de la parole est le TDNN, *Time Delay Neural Network*, présenté dans [waibel89]. Ce modèle a eu de nombreux développements mais l'architecture du réseau et l'implantation des poids neuromimétiques sont assez spécifiques pour permettre de facilement distinguer cette famille connexionniste parmi toutes les autres présentes dans le domaine de la RAP.

L'idée de base de ce type de réseaux est de fournir une représentation spatiale du signal temporel traité. La couche, ou plaque, d'entrée d'un TDNN est constituée de lignes de délais encastrés, les *tapped delay lines*. Ces différentes lignes de délais sont chargées d'effectuer une conversion temporo-spatiale sur le signal d'entrée et permettent ainsi d'obtenir une représentation temps-fréquence. Un vecteur de données issu du prétraitement est calculé à intervalle constant et est ensuite fourni à la plaque d'entrée du réseau qui en tiendra compte pendant une période de temps variable et dépendante de la taille de la dite plaque. Mais la partie du signal visible par le TDNN est généralement très limitée dans le temps. Enfin, les données des vecteurs de prétraitement ne seront à aucun moment modifiées par le mécanisme de stockage.

Dans le cas de la parole, un TDNN sera donc en mesure d'effectuer des tâches de classification sur des phonèmes et sera en mesure, si la plaque d'entrée est assez longue, de prendre en compte les contextes gauche et droit. La nature restreinte de la période de temps observée par le TDNN ne permet cependant pas d'effectuer des traitements sur de longs intervalles qui pourraient permettre de considérer le TDNN comme faisant partie des méthodes globales. Les traitements de nature suprasegmentale sont donc à exclure avec un TDNN tel que définit par [waibel89]. Ce dernier type de traitement a cependant été abordé dans [haffner92a] et [haffner92b].

Chaque unité de la couche cachée d'un TDNN a une vue sur l'ensemble des lignes de la couche inférieure (cf. figure 7.3) mais ne prend pas en compte l'ensemble des délais de chaque ligne. Ainsi, hormis pour les unités de la couche de sortie, les unités synthétisent localement l'information de la

couche amont en effectuant un traitement qui peut être considéré comme équivalent à un masquage.

Une des spécificités du TDNN est le partage des poids synaptiques entre différentes unités d'une même couche. Cette spécificité conduit à la différence qui existe entre l'architecture globale d'un TDNN, architecture implantée lors de la phase d'apprentissage, et le schéma calculatoire du TDNN qui est utilisé une fois les poids synaptiques correctement définis (figure 7.3). Le partage effectif des poids au sein du réseau intervient lors de la phase d'apprentissage du TDNN. À ce stade de la définition du réseau, seule la plaque d'entrée implante des délais alors que les unités des couches cachées calculent toutes leurs activations. Lors de toute rétropropagation du gradient d'erreur ayant lieu pendant la phase d'apprentissage, les poids synaptiques sont modifiés en fonction de l'erreur puis une moyenne est calculée pour respecter le principe selon lequel toute unité d'une ligne de délais d'une couche cachée représente un même traitement sur les données amont. Ainsi, une fois le réseau définit correctement par rapport à la tâche, il est possible d'optimiser les traitements selon le schéma de droite de la figure 7.3, le partage des poids devenant alors sous-jacent.

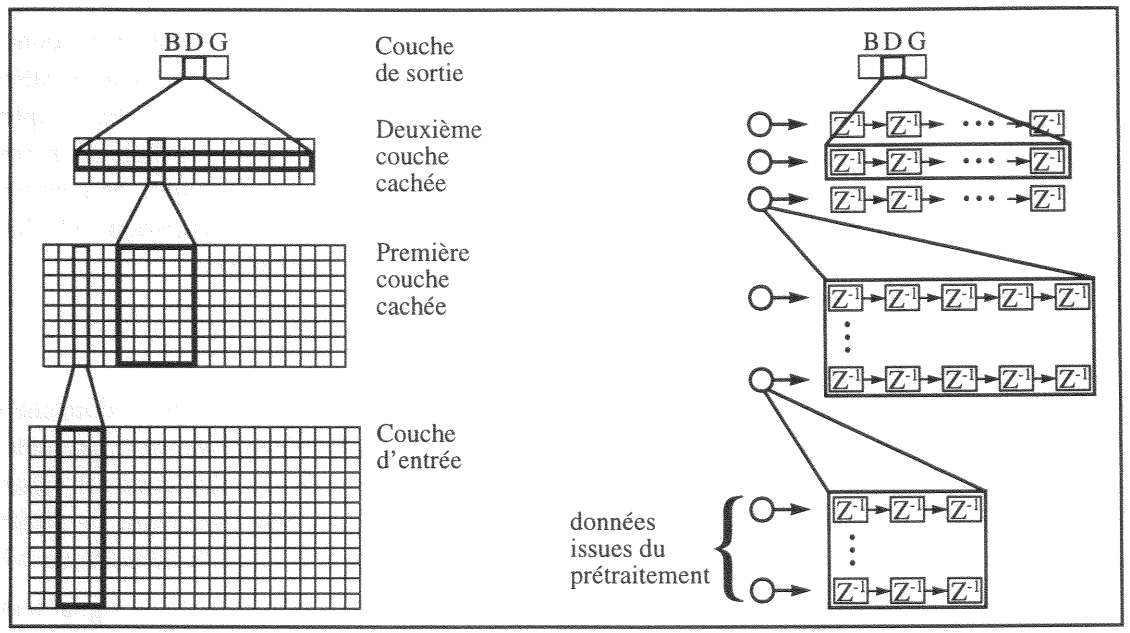


Figure 7.3 : Architecture globale (à gauche) et schéma calculatoire (à droite) d'un TDNN (d'après [waibel89]).

Ce type de traitement n'est cependant pas spécifique au TDNN. La technique du calcul de la moyenne des poids qui vient d'être exposée était à l'origine dédiée aux problèmes de reconnaissance d'images. Grâce à la contrainte de partage des poids synaptiques, il est possible de définir plusieurs ensembles de neurones, chaque ensemble effectuant un seul type de traitement sur toute l'image à analyser. Par spécialisation au cours de l'apprentissage, certains neurones reconnaîtront par exemple les verticales présentent dans une sous-partie de l'image tandis que d'autres reconnaîtront les horizontales présentent dans cette même sous-partie. Cette spécialisation est connue en neurobiologie sous le nom de rétinotopie. Dans le cas de la vision, les études en neurobiologie ont pu prouver que les informations reçues dans la rétine pour la vision centrale étaient projetées dans le cortex visuel primaire de manière à conserver la disposition, ou topologie, de l'image dans la rétine [imbert83]. Cette idée a d'ailleurs été appliquée à la reconnaissance de chiffres manuscrits avec un réseau dont l'architecture est, justement, semblable à celle du TDNN [lecun89a].

Le TDNN agit donc de la même manière que les systèmes neuromimétiques dédiés à la vision. Il est ainsi possible de mettre en parallèle les traitements effectués par ce type de réseaux et l'analyse visuelle de spectrogrammes faites par des experts en phonétique [lonchamp90]. Un TDNN pourra effectuer des traitements pour trouver les transitions dans le signal ou les tenues de formants. Il agit, comme tout expert en phonétique, en système d'analyse et de décodage d'un graphique, que les

données fournies par le prétraitement soient de type spectrographique et donc obtenues par transformée de Fourier, ou d'un type plus dédié à l'analyse de la parole comme les MFCC, *Mel Filter Cepstral Coefficients*.

Le mimétisme entre le TDNN et les systèmes de reconnaissance d'images peut encore être accentué. Certaines architectures dérivées du TDNN ont été proposées pour accroître la localité des traitements effectués. Il a par exemple été proposé de réduire la taille du masque de manière à ce que les unités des couches supérieures aient non seulement une vue partielle sur l'ensemble des délais d'une ligne mais aient également une vue partielle des différentes lignes de délais qui composent la couche amont. [sawai91] a ainsi proposé deux modèles, le *Frequency Time Shift Invariant TDNN* et le *Block Windowed Neural Network*, qui n'ont rien à envier aux modèles dédiés à la reconnaissance d'images. À travers de tels modèles, il est possible de voir apparaître des notions d'échelles différentes. Il est en effet envisageable d'utiliser des masques de tailles différentes, ces différents masques permettant alors d'observer des événements de durée variable à chaque niveau de traitement. La définition des différentes échelles restent cependant la tâche du concepteur qui doit trouver le meilleur compromis lors de la création du réseau.

Le TDNN reste avant tout un réseau appliquant des masques. Il met en œuvre une mémoire de taille limitée de manière à observer l'évolution de certains phénomènes dans une fenêtre étroite mais ne permet pas d'implanter une mémoire comme il est possible d'en trouver dans les systèmes biologiques. De plus, la définition du TDNN telle que nous l'avons présentée suppose que toutes les lignes de délais soient constituées à l'identique et avec une taille unique, interdisant par là même des traitements d'échelles différentes selon les fréquence alors que la parole est connue pour la lenteur relative des événements en basses fréquences par rapport à certains événements présents dans de plus hautes plages fréquentielles.

Quelques études ont été faites sur la manière d'adapter le mieux possible les délais dans un TDNN. En effet, l'architecture, chaque fois qu'elle est définie par le concepteur du réseau, répond a priori à de nombreuses hypothèses qu'il aurait peut-être été bon de résoudre par apprentissage. La taille de la plaque d'entrée, la taille des différentes couches cachées et même l'intervalle, ou pas de temps, entre le calcul de deux vecteurs de données, le *shift*, sont autant de choix à faire qui peuvent avoir une très forte influence sur les résultats. Une plaque d'entrée de trop peu de délais ou un pas de temps trop long peuvent entraîner une incapacité à apprendre une tâche particulière. Il peut donc être intéressant de déterminer au moins une partie de ces paramètres par apprentissage. [rander92] a proposé une méthode pour déterminer le nombre de délais de chaque ligne, c'est à dire déterminer la taille du contexte des unités des couches supérieures, ainsi que les instants de début et de fin des différentes lignes par rapport au temps courant, ce qui correspond à la position du contexte qui peut être plus ou moins loin dans le passé par rapport à l'instant t où se prend la décision. Cette méthode doit en fait être vue comme une méthode simplificatrice puisque l'auteur propose de définir un réseau initial de grande taille et de réduire artificiellement le nombre de délais en utilisant une gaussienne qui caractérise le masque de la ligne de délais elle-même (cf. chapitre 6, figure 6.2.d). Une autre méthode, proposée dans [bodenhausen91a] et baptisée Tempo 2, est plus ambitieuse encore puisqu'elle cherche non seulement à définir la taille et la place du contexte mais également le nombre optimal de poids synaptiques en opérant d'éventuelles fusions. Tempo 2 essaie donc de trouver la meilleure répartition temporelle des délais mais essaie également de minimiser l'espace des paramètres libres du réseau pour améliorer les qualités et la rapidité de l'apprentissage.

7.2.2.2/ Critique de l'horizon fixe

Un TDNN peut être adapté à des tâches de durées variables, grâce à des algorithmes comme Tempo 2. Mais il n'est pas possible de faire cette adaptation sans changer l'architecture du réseau au cours de la phase d'apprentissage, ce processus pouvant en fin de compte devenir dangereux vis-à-vis de la qualité finale. Cet apprentissage doit déterminer le nombre de neurones, leur

localisation temporelle et les poids qui y sont associés. L'architecture ainsi définie doit être assez souple pour prendre en compte les éventuelles variations pouvant exister dans le signal à traiter. Quoi qu'il en soit et quelque soit la manière adoptée pour améliorer le mécanisme mis en œuvre par le TDNN, "l'horizon temporel d'un système [...] défini à partir d'une mémoire explicite est précisément déterminé par le délai de cette mémoire" [berthommier92]. À notre avis, cette phrase souligne très bien le manque de souplesse des architectures fondées sur de simples lignes de délais encastrés. Cette critique n'est d'ailleurs pas isolée [principe95a]. L'architecture du TDNN est adaptée à la parole au sens où elle prend en compte des éléments de nature temporelle, les spatialise et permet ainsi de traiter la parole à la manière d'un expert en phonétique analysant un spectrogramme. Mais les événements se produisant en parole peuvent être très éloignés les uns des autres aussi bien sur l'échelle temporelle que sur l'échelle fréquentielle.

Si le problème à résoudre fait appel à des événements de types différents ou si une architecture est conçue a priori pour résoudre des problèmes de nature variée, il est nécessaire de définir des mécanismes d'apprentissage capables d'adapter l'architecture donnée à des tâches où les périodes à considérer seront durées variables et où les plages fréquentielles à observer seront d'étendues variables.

Le besoin d'un traitement d'événements temporels de nature différente est aujourd'hui renforcé par certains courants de recherche tentant de remettre en cause les choix effectués par l'ensemble de la communauté. Plus particulièrement, certains pensent qu'il faut aller au delà de périodes de 10 à 20 millisecondes actuellement utilisées pour observer le signal et s'orienter vers des durées 10 fois plus longues [hermansky95a]. Le même auteur remarque par ailleurs que l'étroitesse de la fenêtre d'analyse, héritage des pionniers du codage de la parole, rend les systèmes de RAP très sensibles au bruit car cette étroitesse ne permet pas de distinguer l'information utile du bruit ambiant tandis qu'une grande fenêtre temporelle permet d'effectuer un lissage [hermansky95b].

La figure 7.4 présente les différences pouvant exister entre un spectrogramme calculé avec une fenêtre temporelle de 16 *ms* et un autre calculé avec une fenêtre de 128 *ms*. Le signal temporel correspond au fichier *si1039.wav* du corpus *timit/test/dr1/mdab*. Comme cela peut facilement se voir, les informations contenues dans le spectrogramme calculé avec un fenêtre large sont beaucoup moins nombreuses. Le calcul d'une transformée de Fourier avec de telles largeurs de fenêtre permet, en fait, d'effectuer un lissage du signal temporel comme le fait remarquer [hermansky95b], ce lissage ayant tendance à supprimer toutes les informations relatives à des signaux non stationnaires. Nous avons par ailleurs pu remarquer que les fricatives, encore visibles dans le spectrogramme de droite de la figure 7.4, résistaient très mal à une transformation de Fourier avec une fenêtre temporelle de 256 *ms*, seules les voyelles restant alors visibles. Cette fenêtre temporelle large permet, en tout état de cause, d'améliorer la résolution fréquentielle au détriment de la résolution temporelle.

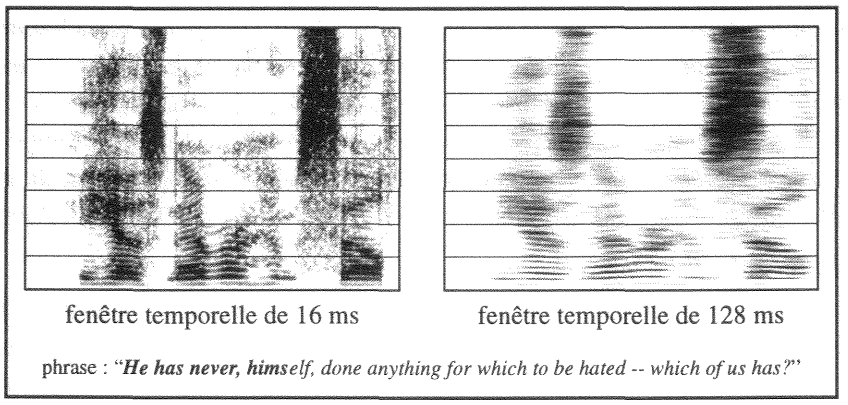


Figure 7.4 : Deux spectrogrammes bande étroite calculés avec deux fenêtres temporelles de tailles différentes.

Ce besoin d'une fenêtre d'analyse plus large est également cité dans [bourlard95a], [bourlard96a] et [bourlard96b] qui préconise d'effectuer des analyses du signal de parole avec des fenêtres temporelles de 30 à 200 *ms* plutôt qu'avec des fenêtres de 10 *ms*. L'auteur y voit plusieurs avantages indépendants. Le premier avantage est lié à la réduction du nombre d'observations effectuées, réduction qui permettrait d'améliorer les performances de systèmes tels que les réseaux de Markov. Un autre avantage vient de la plausibilité biologique d'un tel mécanisme qui permettrait de simuler la mémoire à court terme de l'oreille interne. Le dernier avantage vient du fait que ce type d'observations permettrait d'obtenir un débruitage naturel du signal avec certaines méthodes de traitement telle que la transformée de Fourier, des fenêtres temporelles de longueur importante étant d'ores et déjà employées dans des techniques telles que la soustraction spectrale (cf. chapitre 1, paragraphe 1.7.4) ou la méthode RASTA-PLP (cf. chapitre 1, paragraphe 1.7.2.5) puisque de telles fenêtres permettent d'éliminer le bruit non stationnaire.

Mais ce besoin pour une fenêtre d'analyse temporelle plus large ne permet pas pour autant de totalement s'abstraire d'une analyse à court terme, nécessaire à la reconnaissance de certains phénomènes. Il semble donc souhaitable d'analyser le signal de parole avec une méthode permettant d'ajuster les fréquences d'échantillonnage et donc de représentation du signal. De telles méthodes existent et permettent d'analyser le signal en fonction d'une erreur représentant, en partie, l'erreur d'échantillonnage.

7.2.2.3/ Horizon variable

Le modèle gamma, initialement présenté dans [vries90], a vu sa première application présentée dans [principe93b], par le biais du *Focused Gamma Network* (figure 7.5). Ce réseau est composé d'une plaque d'entrée constituée de lignes de délais encastrés dont les unités ne sont plus de simples délais mais des unités gamma. Cette plaque d'entrée est surmontée d'un perceptron multicouche qui permet d'effectuer la tâche de classification désirée. Ce perceptron n'agit pas à la manière d'un TDNN puisqu'aucune procédure de partage des poids synaptiques n'est mise en place. Seule la plaque d'entrée et son organisation reste donc dans la lignée du paradigme défini par le TDNN et c'est justement par cette plaque d'entrée d'un nouveau type que [vries90] espère obtenir de meilleurs résultats que ceux obtenus avec un TDNN.

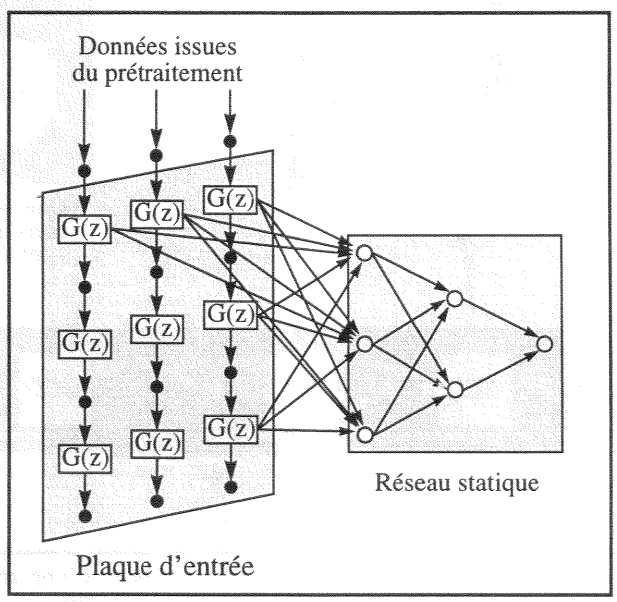


Figure 7.5 : Architecture du *Focused Gamma Network* (d'après [principe93b]).

Cette plaque d'entrée, composée d'unités gamma, mémorise le signal analysé de manière optimale par rapport à la tâche à apprendre. Le point optimal peut être atteint par apprentissage des différents coefficients de feedback du réseau, cet apprentissage particulier pouvant se faire pendant la phase

d'apprentissage général du réseau. Bien qu'aucune procédure de partage des poids ne soit mise en place au niveau des connexions synaptiques, une règle importante de ce réseau est la définition d'un unique paramètre μ pour l'ensemble de la plaque d'entrée. Ainsi, toutes les unités gamma implantent le même filtre et réalise la même fonction de mémorisation. Ce choix d'implantation peut paraître réducteur mais peut être justifié vis-à-vis de la procédure d'apprentissage, que nous verrons plus loin, car certaines études ont postulé de l'impossibilité d'implanter plusieurs types de mémoire concurrents au sein d'une même structure [kuo93a]. Malheureusement, ce choix pour un unique paramètre n'a pas permis de lever les problèmes d'apprentissage existants encore aujourd'hui.

Le point optimal étant défini par le biais d'un unique coefficient μ, il est possible de déterminer un certain nombre de valeurs permettant d'avoir des indications sur le processus appris par le réseau.

La fonction de transfert de l'unité gamma de rang k dans la ligne est donnée par l'équation 7.4 où μ est un paramètre local. Cette fonction de transfert a un pôle unique en 1-μ. La stabilité de cette fonction de transfert est démontrée pour des valeurs de μ comprises entre 0 et 2 [vries92].

$$G_k(z) = \left(\frac{\mu}{z-(1-\mu)}\right)^k \qquad \text{(Éq. 7.4)}$$

Outre cette stabilité dans l'intervalle]0,2[, il est possible de montrer les différences de comportement du filtre gamma pour des valeurs comprises dans cet intervalle. Le cas trivial est celui où μ est égal à 1. Dans ce cas, l'équation $y_{i,t} = \mu \cdot y_{i-1,t} + (1 - \mu) \cdot y_{i,t-1}$ se réécrit simplement en $y_{i,t} = y_{i-1,t}$. Nous nous retrouvons alors dans le cas d'une ligne de délais simples, identiques à ceux implantés dans un TDNN. Un deuxième cas correspond à des valeurs de μ comprises entre 1 et 2, bornes exclues. Le filtre gamma est alors un filtre passe-haut. Comme nous l'avons vu au chapitre 4, paragraphe 4.1.2, un filtre passe-haut permet de supprimer les composantes stationnaires d'un bruit. Enfin, le troisième cas correspond à des valeurs de μ comprises entre 0 et 1, bornes exclues. Le filtre gamma réagit alors comme un filtre passe-bas et est capable d'implanter de la mémoire par conservation de son activité, celle-ci diminuant de manière exponentielle. C'est ce dernier cas qui nous intéresse tout particulièrement.

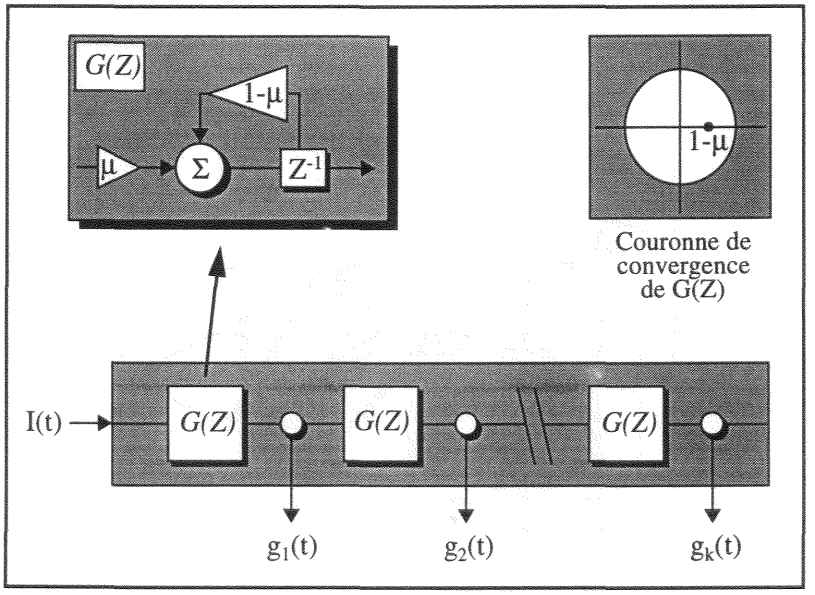


Figure 7.6 : Schéma d'une ligne de délais gamma, schéma d'une unité gamma et couronne de convergence de la transmittance du filtre gamma.

Ayant défini, dans l'équation 7.4, la fonction de transfert du $k^{\text{ième}}$ délai d'une ligne de délais gamma, nous allons maintenant voir la réponse impulsionnelle dans le cas continu qui est donnée par l'équation 7.5.

$$g_k(t) = \frac{\mu^k}{(k-1)!} t^{k-1} e^{-\mu t} \qquad k = 1, \ldots, K \quad \mu > 0 \qquad \text{(Éq. 7.5)}$$

L'équation 7.6, calculée à partir de l'équation 7.5, donne le temps d'échantillonnage moyen pour le $k^{\text{ième}}$ délai de la ligne. Ce temps d'échantillonnage moyen correspond au temps d'arrivée du maximum de l'impulsion dans l'unité.

$$n_k \equiv \sum_{t=0}^{\infty} t g_k(t) = -z \frac{d}{dz} G(z) \Big|_{z=1} = \frac{k}{\mu} \qquad \text{(Éq. 7.6)}$$

L'équation 7.7, qui repose sur la valeur du temps d'échantillonnage moyen défini par l'équation 7.6, donne la période moyenne d'échantillonnage pour le $k^{\text{ième}}$ délai de la ligne de délais.

$$\Delta n_k = n_k - n_{k-1} = \frac{1}{\mu} \qquad \text{(Éq. 7.7)}$$

Il est possible, à partir de la période moyenne d'échantillonnage, de calculer deux valeurs distinctes donnant des indications intéressantes sur la mémoire mise en œuvre par la ligne de délais étudiée dans le cas où les filtres gamma de la plaque d'entrée partagent tous la même valeur de μ. Les calculs qui suivent sont valables pour la plaque d'entrée dans son ensemble et non plus seulement pour chacune des lignes de délais prises séparément.

Une première valeur intéressante est tout simplement la valeur de μ [principe93e]. Cette valeur peut en effet être considérée comme représentant la résolution fournit par les filtres gamma pour la représentation du signal traité. Cette résolution peut être calculée selon l'équation 7.8 qui fait intervenir la période moyenne d'échantillonnage. En ne considérant que le cas où μ est compris dans l'intervalle]0,1], un μ proche de 1 offrira une bonne résolution sur le signal d'entrée, celui-ci restant presqu'intact, alors qu'un μ proche de 0 offrira une très mauvaise résolution, le signal d'entrée étant fortement confondu avec la trace du signal pris en compte aux pas de temps précédents. La résolution est un concept qui doit donc être opposée au concept de mémoire, l'amélioration de l'une se faisant au détriment de l'autre.

$$R_k = \frac{1}{\Delta n_k} = \mu \qquad \text{(Éq. 7.8)}$$

Une seconde valeur intéressante concerne la profondeur effective de la plaque d'entrée. Cette profondeur abstraite correspond au nombre effectif de délais simples, de type z^{-1}, qui auraient dû être implantés pour qu'une mémoire équivalente soit mise en place. La profondeur permet donc de connaître la mémoire implantée par la ligne de délais gamma connaissant le nombre de filtres gamma mis en œuvre et la valeur de μ. Cette profondeur, comme le montre l'équation 7.9, peut également être calculée à partir de la période moyenne d'échantillonnage.

$$D_k \equiv \sum_{i=1}^{K} \Delta n_i = \frac{K}{\mu} \qquad \text{(Éq. 7.9)}$$

L'équation 7.10, qui permet d'apprécier la profondeur effective d'une ligne de délai gamma, est obtenue en généralisant les deux variables D_k et R_k de profondeur et de résolution à n'importe quel rang dans la ligne de délai. Cette équation permet de connaître le nombre K de délais, la profondeur D et la résolution R étant donné. Ainsi, après apprentissage et obtention d'une valeur de μ, il est possible de connaître le nombre effectif de délais implantés par une ligne de K filtres gamma selon la formule :

$$K = D \times R \qquad \text{(Éq. 7.10)}$$

L'équation 7.10 est une formule générique et peut très facilement être réécrite pour donner l'équation 7.11. Cette dernière équation est également très intéressante puisqu'elle permet de connaître le nombre effectif de délais implanté par chaque ligne de la plaque d'entrée à partir du nombre réel de délais et de la valeur de μ. Cette équation n'est que la simple généralisation de l'équation 7.9.

$$D = K/R \qquad \text{(Éq. 7.11)}$$

Les deux dernières équations permettent donc de connaître, a posteriori, le nombre de délais qu'il a été nécessaire d'implanter pour apprendre la tâche présentée en apprentissage et donc le nombre optimal de délais qu'il aurait fallu choisir pour résoudre cette même tâche avec un TDNN.

Une dernière formule permet de mesurer la vitesse de propagation de la valeur maximale d'une impulsion fournie en entrée d'une ligne de filtres gamma partageant tous la même valeur de coefficient de feedback [vries91a]. Ainsi, le $k^{\text{ième}}$ filtre d'une ligne verra le pic de l'impulsion passer au temps t_p qui est défini par l'équation 7.12 suivante :

$$t_p = \frac{k-1}{\mu} \qquad \text{(Éq. 7.12)}$$

Cette propriété pourra être retrouvée visuellement par le lecteur dans l'annexe 2, tant dans les graphiques du paragraphe A2.2 que dans les graphiques du paragraphe A2.3.

7.2.3/ Étude comparée du modèle gamma et du TDNN

Le modèle gamma est avant tout défini pour améliorer les capacités, ou pallier les insuffisances, des réseaux neuronaux comportant des lignes de délais encastrés. Le modèle connexionniste le plus représentatif de cette classe est le TDNN, très largement employé en reconnaissance automatique de la parole.

La comparaison des capacités d'un réseau gamma et d'un TDNN à résoudre les même tâches semble donc une étape nécessaire dans l'établissement de la supériorité de l'un par rapport à l'autre. Ce travail a fait l'objet d'une thèse [kuo93a] qui s'est surtout centrée sur l'étude des moyens de prédiction de systèmes dynamiques non linéaires, le problème justement résolu de manière théorique par [takens81]. Le travail effectué dans cette thèse montre, lorsque la comparaison est faite, la supériorité du modèle gamma, ou même d'un TDNN récurrent, sur l'architecture du TDNN en temps que prédicteur de la valeur suivante d'une série temporelle. Les séries étudiées sont cependant d'une catégorie assez particulière puisqu'il s'agit de séries dites chaotiques étudiées dans les cas où des paramètres spécifiques les rendent déterministes. La relation avec la reconnaissance de la parole est donc assez difficile à faire.

Une comparaison en relation avec le monde de la RAP a été faite par [lawrence96]. Cette article semble, a priori, ambitieux. Il compare en effet l'architecture gamma au réseau FIR ([back91], cf. chapitre6, paragraphe 6.4.3.4), au TDNN et à la méthode des k plus proches voisins, ou *k Nearest Neighbour*, k-NN, avec approximation locale, k-NNLA. Les résultats obtenus permettent, encore une fois, de prouver la supériorité de modèle gamma sur le TDNN mais aussi, cette fois, sur d'autres méthodes. Malheureusement, la tâche étudiée ne correspond qu'à une tâche d'identification du phonème /A/ vis-à-vis du phonème /S/ dans le corpus TIMIT. Ainsi, bien que cette tâche soit relative au domaine de la RAP, il faut reconnaître toutes les limites d'une telle comparaison.

Malgré la comparaison positive qui vient d'être faite, il faut reconnaître que le modèle gamma n'est pas plus adaptable à tous les problèmes de la parole que ne l'était le TDNN. S'il est permis d'espérer que le mécanisme du filtre et la procédure d'apprentissage qui y est associée permettront de définir un réseau gamma plus correctement que ne l'aurait été un TDNN, [bengio93] note cependant que le modèle gamma n'est pas plus adapté aux séquences élastiques que ne l'était le TDNN. Il est donc vain d'espérer être en mesure de traiter des cas se trouvant à une extrémité du

spectre du problème et qui réclament une mémoire de taille supérieure à celle qui est implantée ou implantable, pour le modèle gamma tout comme pour le TDNN. Au mieux, un réseau gamma permettra de traiter plus de cas que ne pouvait en traiter un TDNN.

7.2.4/ Rétention des moments de Poisson du signal

Le mécanisme mis en œuvre par un réseau gamma permet de conserver en mémoire une partie de la trace du signal. Ce mécanisme permet donc de transformer le signal perçu d'une manière dépendant du ou des coefficients de régression. Cette trace de signal peut être qualifiée de moment de Poisson du signal et a été étudiée tant dans [celebi95a] que dans [principe95]. [celebi95a] étudie ce phénomène de manière tant théorique que pratique en fournissant une étude approfondie, reprenant des travaux entrepris par ailleurs [saha82]. Une première présentation des travaux cités pourra être trouvée dans [celebi94].

[principe95] étudie ce phénomène de rétention d'une partie du signal de manière intéressante. Il propose ainsi la définition d'une nouvelle méthode de prétraitement du signal utilisant des filtres gamma. Cette méthode repose sur l'utilisation des valeurs d'un spectre extrait du signal par transformée de Fourier, valeurs qui sont ensuite traitées par un banc de filtres gamma permettant de conserver les traces du signal observé. La figure 7.7 montre ainsi deux représentations spectrographiques différentes d'un même extrait du corpus TIMIT : le mot *greasy* de la phrase *sa1.wav*. Le graphique de gauche présente la représentation spectrographique conventionnelle obtenue par transformée de Fourier, les fréquences étant ici placées sur l'axe des abscisses tandis que l'ordonnée représente le temps. Le graphique de droite présente une nouvelle représentation temps-fréquence permettant de visualiser les moments reconstruits du même signal à l'aide de filtres gamma.

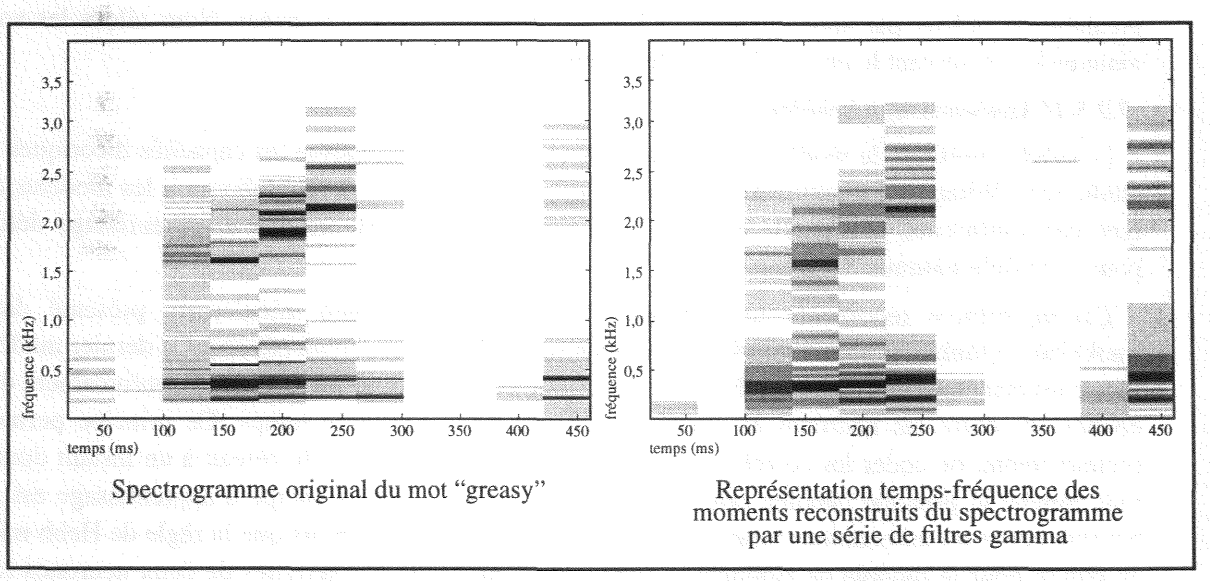


Figure 7.7 : Spectrogramme et représentation temps-fréquence des moments reconstruits par une série de filtres gamma (d'après [principe95a]).

Il est possible de considérer cette représentation comme similaire à un modèle d'audition (cf. chapitre 1, paragraphe 1.7.2.6) puisqu'elle est capable de simuler l'amortissement progressif de l'excitation que produirait le signal à l'intérieur de l'oreille.

7.2.5/ Apprentissage

L'apprentissage est le grand problème posé par le modèle gamma et tous les modèles connexionnistes récurrents de la même classe. Bien que certains autres modèles connexionnistes ([leigthon91], [sastry94]) s'en rapprochent très fortement et aient donc, eux aussi, besoin d'une procédure particulière, aucune méthode d'apprentissage efficace n'a, jusqu'à présent, été trouvée.

Certains modèles ont bien été définis pour pallier le manque théorique par des astuces architecturales mais le problème de l'apprentissage de la dynamique reste encore à résoudre.

L'apprentissage a toujours été le problème majeur qui a défini les limites des recherches en connexionnisme. Après les premières études, qui ont été très porteuses et capables de générer un engouement certain [mcculloch43], [hebb49], la mise en lumière des limites des modèles de l'époque [minsky69] a porté un coup fatal à toutes les recherches qui étaient alors entreprises. Il a alors fallu attendre une décennie et demie pour voir apparaître un nouvel algorithme d'apprentissage [lecun85] capable de résoudre les problèmes qui avaient été soulevés.

Le modèle gamma et les modèles qui peuvent lui être associés posent à leur tour un problème d'apprentissage. La mise en place d'un nouveau concept issu de la neurobiologie, la décroissance de l'activité au sein de la cellule même, pose un problème d'adaptation de l'algorithme d'apprentissage original utilisé dans les perceptrons multicouches et dans tous les modèles qui en sont, de près ou de loin, dérivés. Ce problème d'apprentissage est généralisable à d'autres modèles récurrents pour des problèmes similaires que nous avons déjà exposés (cf. chapitre 6, paragraphes 6.3.8 et 6.4.3.3). La question se pose même de savoir si ce manque théorique ne sera pas la cause d'une mise en sommeil de ce domaine de recherche particulier. Les études théoriques en cours sur les systèmes dynamiques non linéaires permettent, heureusement, de préserver une lueur d'espoir (cf. paragraphe 7.2.1). Les résultats de ces recherches permettront probablement de généraliser le paradigme de l'apprentissage connexionniste à des architectures plus complexes que celles qui étaient initialement envisagées dans le domaine.

Le modèle gamma doit, tout comme d'autres modèles récurrents, déterminer les valeurs optimales des coefficients de feedback pour résoudre du mieux possible une tâche particulière qu'il faut au préalable modéliser par apprentissage. Différentes méthodes ont été essayées. Nous allons les voir maintenant en suivant la genèse du modèle gamma.

7.2.5.1/ Apprentissage hebbien

[vries91a] présente le modèle d'un point de vue théorique, en analyse les capacités théoriques et étudie son affiliation à d'autres modèles tels que le modèle additif de Grossberg ou les modèles de type convolutionnel. Cet article présente également le premier algorithme d'apprentissage défini pour le modèle gamma.

Cet algorithme repose sur un principe d'apprentissage hebbien initialement présenté dans [tank87a] et [tank87b] pour le modèle de *Concentration In Time Network (CITN)*. Ce dernier modèle utilise les principes du modèle de Hopfield mais propose de calculer les valeurs de renforcement des connexions entre les neurones en fonction de l'activité à travers le temps. Ce principe permet, premièrement, de coder les corrélations d'activités entre les neurones du réseau à un instant donné tout comme le principe hebbien original. L'autre capacité d'un tel principe d'apprentissage est de pouvoir coder les associations temporelles entre événements. Ainsi, alors que la règle de Hebb mise en œuvre pour le modèle de Hopfield se fonde sur la moyenne des activités de deux neurones (cf. chapitre 6, équation 6.8), le principe est ici modifié pour prendre en compte une séquence d'activations de deux neurones. La mise à jour des poids n'est plus aussi directe que par l'équation 6.8 du chapitre 6 mais fait intervenir un gradient de modification calculé à partir des vecteurs d'activation de deux neurones, le processus de mise à jour des poids correspondant à :

$$\frac{dw_{ij}}{dt} = \eta y_1(t)\,(y_2(t))^T \qquad \text{(Éq. 7.13)}$$

Dans cette équation, η représente le coefficient d'apprentissage et $y_i(t)$ représente la séquence des activations du neurone i sur l'intervalle t. Ce principe général a été adapté au modèle gamma en considérant la récurrence gamma comme une séparation entre deux valeurs distinctes : la valeur de sortie effective de l'unité et l'activité neuronale proprement dite. Cette dernière activité correspond

en fait à la valeur d'entrée du neurone puisque, comme nous l'avons vu au paragraphe 7.1.2, le mécanisme de filtre est implanté dans la plaque d'entrée mais pas dans les couches supérieures. La formule de mise à jour des valeurs de μ est donc donnée par :

$$\frac{d\mu_k}{dt} = \eta x_k(t)\ (y_k(t))^T \qquad \text{(Éq. 7.14)}$$

Dans cette équation, $x_k(t)$ correspond à la séquence des valeurs d'entrée du filtre gamma et $y_k(t)$ correspond à la séquence des valeurs de sortie. Cette équation reste locale à une unité gamma dans le temps et l'espace.

Cette méthode d'apprentissage souffre d'une gros désavantage : elle est incapable de faire le lien entre l'apprentissage non supervisé de la valeur de μ et l'apprentissage supervisé des poids d'un réseau utilisant la rétropropagation du gradient d'erreur, cette rétropropagation permettant d'apprendre des tâches de classification. Si cette méthode avait été implantée dans un même réseau et si les valeurs des connexions de la plaque d'entrée, composée de filtres gamma, avaient été mises à jour selon l'équation 7.14, l'apprentissage de la tâche par le perceptron multicouche aurait été totalement indépendante de l'apprentissage des caractéristiques du signal par la plaque d'entrée.

Cet algorithme d'apprentissage, tel qu'il était présenté dans [vries91a], n'a pas eu de suite.

7.2.5.2/ Première mise en œuvre de l'apprentissage récurrent

Un autre méthode d'apprentissage a été présentée dans [vries92] et [principe93a]. Cette méthode est beaucoup plus pratique puisqu'elle repose sur une architecture baptisée *Focused Gamma Network* (cf. figure 7.5). Le FGN est un réseau mettant en œuvre un perceptron multicouche et une plaque d'entrée composée de filtres gamma. L'algorithme d'apprentissage prend en compte une erreur sur la couche de sortie et ajuste les connexions synaptiques grâce à la méthode de rétropropagation du gradient d'erreur comme pour toute tâche de classification. La nature récursive de la plaque d'entrée, où se trouvent les filtres gamma impose de revoir l'algorithme dans un souci de complexification : il faut tenir compte de la nouvelle architecture de certains neurones. La présence du filtre dans la seule couche d'entrée permet, a contrario, de simplifier la méthode d'apprentissage des connexions récursives. L'algorithme choisi pour adapter les poids de feedback est la méthode RTRL, *Real Time Recurrent Learning* [williams89a], car elle est, selon l'auteur [vries92], bien plus adaptée à de petits réseaux dynamiques que ne l'est la méthode BPTT, *Back Propagation Through Time* [werbos90].

La mise à jour des connexions, récursives ou non, du réseau se fait selon la valeur de l'erreur à la sortie du réseau mais cette mise à jour se fait de deux manières distinctes : une première partie du processus d'apprentissage ajuste les poids synaptiques du perceptron tandis que la deuxième partie du processus ajuste la valeur du feedback des unités de la plaque d'entrée. La première partie du processus exploite l'erreur de sortie E telle que définie dans l'équation 7.17 où intervient une notion temporelle. Le gradient d'erreur des poids synaptiques est calculé comme pour tout perceptron multicouche selon la formule de l'équation 7.18 alors que le gradient d'erreur pour les coefficients de régression est donné par l'équation 7.19.

$$\Delta w_{ijk} = -\eta \frac{\partial E}{\partial w_{ijk}} \qquad \text{(Éq. 7.15)}$$

$$\Delta \mu = -\eta \frac{\partial E}{\partial \mu_i} \qquad \text{(Éq. 7.16)}$$

$$E = \frac{1}{2}\sum_t \sum_i [e_i(t)]^2 = \frac{1}{2}\sum_t \sum_i [d_i(t) - x_i(t)]^2 \qquad \text{(Éq. 7.17)}$$

$$\frac{\partial E}{\partial w_{ijk}}(t) = \frac{\partial E}{\partial w_k}(t) \times \frac{\partial}{\partial w_k}(t) \qquad \text{(Éq. 7.18)}$$

$$\frac{\partial E}{\partial \mu_i}(t) = \sum_m \frac{\partial E}{\partial x_m(t)} \times \frac{\partial x_m(t)}{\partial x_{ik}(t)} \times \frac{\partial x_{ik}(t)}{\partial \mu_i} \qquad \text{(Éq. 7.19)}$$

$$= -\sum_m e_m(t)\, \sigma'_m(net_m(t)) \sum_k w_{mik} \alpha_i^k(t)$$

Le coefficient α de l'équation 7.19, qui permet de connaître le gradient d'erreur de μ, se calcule récursivement selon l'équation 7.21, le premier pas de l'induction étant donné par l'équation 7.20. Ce coefficient quantifie des variations du signal d'entrée dans l'unité gamma considérée.

$$\alpha_i^0(t) = 0 \qquad \text{(Éq. 7.20)}$$

$$\alpha_i^k(t) = (1-\mu_i)\,\alpha_i^k(t-1) + \mu_i \alpha_i^{k-1}(t-1) + [x_{i,k-1}(t-1) - x_{i,k}(t-1)] \qquad \text{(Éq. 7.21)}$$

La méthode d'apprentissage qui est exposée ici peut être retrouvée dans beaucoup des articles de recherche écrits par les concepteurs du modèle gamma et ceux qui l'ont repris à l'identique.

7.2.5.3/ Deuxième approche récurrente

Une deuxième approche de l'apprentissage récurrent a été testée par les créateurs du modèle devant les difficultés de convergence qui s'étaient faites jour avec la méthode précédemment exposée. Deux grandes classes d'algorithmes ont été définies pour l'apprentissage dans les modèles récurrents : le RTRL, qui a servi de pierre d'achoppement au premier algorithme d'apprentissage récurrent dans le modèle gamma, et la BPTT, *Back Propagation Through Time*, qui avait été considérée comme inadéquate dans [vries92].

La méthode BPTT a été étudiée par la suite et ceci s'est révélé profitable. Un premier article, [lefebvre93], a étudié la possibilité de mettre en œuvre ce paradigme avec le modèle gamma. Cette tentative s'est révélé fructueuse du fait des concepts qui ont prévalus à la définition de la BPTT. Dans cette méthode, le gradient calculé sur plusieurs pas de temps est exact car l'état du réseau est conservé par duplication de tous les neurones sur la totalité des pas de temps calculés. Si le RTRL est un algorithme temps réel, comme le précise son nom, c'est parce que le gradient d'erreur peut être rétropropagé à tout moment car il est approximé à chaque pas de temps en fonction de sa valeur au pas de temps précédent et de sa valeur courante. Cette approximation se retrouve dans l'équation 7.21.

Contrairement au RTRL, la méthode BPTT met en place une duplication du réseau à chaque pas de temps jusqu'à ce qu'une erreur puisse être rétropropagée. Dans ce cas, le calcul du gradient se fait en rétropropageant l'erreur à travers le réseau et sur tous les pas de temps, les connexions récurrentes servant de point de passage d'un pas de temps à l'autre, c'est à dire d'une copie du réseau à l'autre. Après mise à jour des connexions avec leurs gradients, une moyenne est effectuée sur tous les poids dupliqués pour obtenir la nouvelle valeur de la connexion. Cette procédure est donc nettement plus lourde que le RTRL puisque la rétropropagation du gradient se fait désormais dans les deux dimensions que sont le réseau au dernier pas de temps et ses duplications aux pas de temps précédents, accroissant d'autant le besoin en puissance de stockage et de calcul.

Les équations de mise à jour des poids ne sont pas fondamentalement différentes de celles présentées pour la méthode RTRL puisque la principale différence se fait au niveau de l'implantation algorithmique.

Malgré l'exposé que nous venons d'en faire, cette méthode d'apprentissage par BPTT ne permet pas d'obtenir une absolue fiabilité de la convergence : les apprentissages non convergents ne sont pas

totalement éliminés. Or les deux méthodes d'apprentissage que nous venons de voir sont issues des deux grands paradigmes existants dans le domaine de l'apprentissage dans les réseaux récurrents. Il semble donc nécessaire, à ce stade, d'essayer de définir une procédure d'apprentissage différente et innovante.

7.2.5.4/ Approche actuelle

L'approche actuellement étudiée dans certaines recherches tente de revenir à la première méthode d'apprentissage proposée. L'idée majeure est de combiner un processus de rétropropagation du gradient d'erreur à un processus d'apprentissage hebbien. Il s'agit de trouver une méthode efficace alliant les capacités de l'apprentissage supervisé dans les modèles récurrents, nécessaire à la mise en place de toute tâche de classification ou d'identification de système, et la facilité de mise en œuvre de l'apprentissage non supervisé à travers le temps comme cela avait été initialement envisagé. La fusion de ces deux types d'apprentissage n'est cependant pas triviale puisque la première méthode repose sur un calcul de gradient d'erreur alors que la deuxième se fonde sur une moyenne des activations neuronales, ces deux notions semblant, a priori, antagonistes.

Il existe pourtant un lien entre ces deux méthodes d'apprentissage. [wang95a] montre ainsi que l'apprentissage supervisé par la méthode des moindres carrés devient équivalent à un apprentissage non supervisé lorsque l'information mutuelle entre l'entrée et la sortie désirée atteint un maximum ou un minimum. Mais cette constatation est loin d'être suffisante pour définir une nouvelle procédure d'apprentissage.

Certaines tentatives ont cependant été entreprises dans les domaines où elles étaient réalisables. [rigoll92] propose ainsi d'utiliser un apprentissage non supervisé pour définir les valeurs des poids connexionnistes d'un perceptron. La procédure de calcul de l'erreur est, bien sûr, modifiée. Elle permet de définir un réseau effectuant une tâche de quantification vectorielle. Cependant, cet apprentissage non supervisé permet d'atteindre des performances globales satisfaisantes car les sorties du réseau sont exploitées par plusieurs réseaux de Markov qui permettent donc de redéfinir par l'architecture les classes qui n'ont pas pu être définies par l'apprentissage non supervisé. Le perceptron a ainsi été détourné de sa fonction première tandis que la compréhension de ses sorties est rendue plus difficile. Cette technique d'apprentissage non supervisé est donc à proscrire dans le cas de tâches de classification, lorsque la sémantique des sorties est primordiale. La méthode qui vient d'être présentée ne doit pas non plus être confondue avec la méthode mise en œuvre par [bourlard90a], [bourlard90b] qui utilise un réseau connexionniste pour calculer les probabilités des événements observés, probabilités utilisées par la suite par des réseaux de Markov.

Le besoin de définir une procédure d'apprentissage prenant en compte les paradigmes d'apprentissage supervisé et non supervisé pourra ressembler, aux yeux de certains, à une tentative désespérée pour sauver les architectures à récurrence locale. Ce jugement négatif est cependant un peu brutal car le problème de l'apprentissage dans les modèles connexionnistes n'est peut être pas encore totalement cerné. L'actuelle rétropropagation du gradient d'erreur doit être considérée comme parfaitement adaptée à l'architecture pour laquelle elle a été définie : le perceptron. La qualité de modélisation qu'elle a permis d'atteindre avec cette architecture neuromimétique ne doit cependant pas nous obliger à la considérer comme une solution applicable à tous les modèles connexionnistes, surtout lorsque ceux-ci s'éloignent de l'architecture originale. Les machines de Boltzmann sont un bon exemple de cette apparente inaptitude. Les réseaux connexionnistes à récurrence locale doivent également être considéré de la même manière puisqu'ils intègrent, en plus du mécanisme d'intégration des connaissances, un mécanisme biologiquement plausible de décroissance de l'activité interne. Ce mécanisme de décroissance de l'activité peut se retrouver dans le cortex au niveau des connexions synaptiques, comme [kim92] l'a bien défini dans son modèle (cf. chapitre 6, paragraphe 6.4.3.5). Ce mécanisme a également été étudié par [bell93] mais, cette fois, au niveau biologique. Ce dernier article étudie la transmission de l'information électrique dans le cerveau au

niveau neuronal et présente une modélisation de la répartition des activités à l'intérieur d'une membrane vis-à-vis des afférences selon une loi anti-hebbienne. Tout comme [hebb49] qui a présenté le principe de la règle de Hebb, [bell93] semble renforcer l'importance d'un tel principe mais cette fois au niveau du neurone lui-même, ce principe n'étant malheureusement pas pris en compte dans la rétropropagation du gradient d'erreur.

L'apprentissage purement hebbien de l'association temporelle a été étudiée de manière approfondie dans [zhang94] avec des techniques malheureusement assez peu généralisable aux perceptrons. La recherche d'une méthode d'apprentissage adaptée a donc pris d'autres voies, avec des résultats qui ne semblent pas avoir atteint une qualité ou une généralité suffisante pour être applicables de manière générale. Il semble d'abord souhaitable de rappeler que [bengio94a] a prouvé que les problèmes d'apprentissage observés ne peuvent pas être résolus par une redéfinition de l'erreur de sortie et que c'est donc le processus interne de mise à jour des coefficients qui doit être modifié. [tsung90] propose ainsi de modifier le processus de calcul des deltas de la méthode RTRL en reprenant des notions de la version continue de cet algorithme. Diverses méthodes de mise à jour des poids ont également été étudiées par [srinivasan94] qui prône pour un apprentissage des poids en mode *batch*. D'autres technique plus novatrices ont également été développées telle que la prédéfinition de certains poids du réseau [omlin92] qui correspond, en quelque sorte, à la conservation des valeurs d'initialisation bien que cette technique soit présentée avec un aspect d'ingénierie des connaissances. [kehagias91] a, pour sa part, tenté d'adapter l'algorithme *forward-backward* de Baum [baum67] à la définition des valeurs synaptiques en modifiant le mode de calcul local du delta. Il est également possible d'utiliser le paradigme du renforcement [sutton84] en adaptant la méthode *Expectation/Maximisation* [dempster77] pour obtenir un algorithme probabiliste [bengio94b]. Il est enfin possible de définir une architecture utilisant la méthode sigma-pi et des états bistables pour associer, par apprentissage, des événements de références temporelles différentes [dorizzi92]. Les données d'apprentissage ont également une influence non négligeable sur le succès ou l'échec des apprentissages comme l'a fait remarquer [nerrand94].

7.2.5.5/ Critique des méthodes d'apprentissage utilisées

Il devient aujourd'hui de plus en plus clair que toutes les méthodes d'apprentissage actuellement utilisées dans le domaine des réseaux connexionnistes récurrents ne sont pas adaptées à la tâche pour laquelle elles ont été définies et mises en place. La littérature comprend de plus en plus d'articles de recherche et de rapports techniques montrant que la particularité architecturale des réseaux connexionnistes à récurrence locale n'est pas correctement prise en compte.

Deux phénomènes distincts existent et prouvent qu'un grand problème reste encore à résoudre. Le premier problème concerne la perception du corpus des données par la procédure d'apprentissage, perception se faisant par l'intermédiaire du réseau dont l'architecture est modifiée. Le deuxième problème concerne les incapacités de la procédure de rétropropagation du gradient d'erreur à contrer les diminutions ou augmentations excessives des valeurs des gradients d'erreurs, *vanishing* ou *exploding gradients* dont nous avons déjà parlé au chapitre 6, paragraphe 6.3.8. Ce dernier problème est particulièrement sensible dans le modèle gamma où les valeurs de μ sont restreintes à l'intervalle]0,1] qui accentue d'autant plus le *vanishing* des gradients.

Le premier problème, relatif à la modification de la perception du corpus, est lié à la nature du mécanisme gamma qui implante un filtre et effectue une rétention des moments de Poisson du signal. Ce problème peut être généralisé à l'ensemble des réseaux connexionnistes à récurrence locale. L'apprentissage dans un réseau connexionniste statique permet, normalement, de faire le lien entre l'ensemble des données présentées à l'entrée et les sorties désirées. Ce lien est créé par une suite de modifications des poids connexionnistes diminuant l'erreur totale en sortie. Cette modification se fait, à un niveau local, par perception de l'erreur instantanée générée en sortie de l'unité. Cette unité synthétise à son niveau une nouvelle erreur en fonction de sa contribution aux erreurs des unités de la

couche aval qu'elle alimente, en tenant compte des connexions synaptiques.

La nouvelle architecture choisie, un réseau à l'architecture générale d'un perceptron dont les unités intègrent un mécanisme de filtre, modifie la perception du corpus par génération, pour chaque unité, d'une valeur de sortie qui tient compte tant des nouvelles entrées du réseau que des entrées passées. La partie de l'activation concernant les entrées passées doit permettre de simuler mathématiquement les notions d'excitation ou d'inhibition retardée mais peut grandement influer sur la sortie du réseau, en effectuant une sorte de lissage. Ce lissage peut créer une erreur en sortie qui n'aurait peut-être pas lieu d'être avec un perceptron, générant un processus de mise à jour des poids avec de fortes valeurs dans certains cas. Ce processus de mise à jour va entraîner une modification des poids connexionnistes du réseau tout autant que des valeurs des coefficients de feedback de chacune des unités. Le processus d'apprentissage se faisant par présentation itérée du corpus, les poids seront modifiés en partie en fonction de l'erreur générée par les filtres qui verront, eux, leurs comportements également modifiés au cours du processus d'apprentissage. La définition des poids connexionnistes et des coefficients de feedback est donc liée alors que l'erreur utilisée en sortie ne permet pas de prendre en compte le comportement des filtres. Autrement dit, l'association simple entre données d'entrée et données de sortie, valide pour un perceptron, ne permet pas de prendre correctement en compte les moments du signal, rendant l'apprentissage problématique. Nous avons ainsi pu remarquer un comportement oscillant des poids connexionnistes tout autant que des coefficients de régression lors des premiers apprentissages que nous avons effectués, le réseau étant alors incapable d'apprendre correctement la tâche qui lui était présentée.

Les deux procédures d'apprentissage existantes que nous avons vu respectivement au paragraphe 7.2.5.2 et au paragraphe 7.2.5.3 ont des buts foncièrement différents. Ainsi, le RTRL a été défini pour l'apprentissage de tâches se rapportant au domaine de l'identification des systèmes [williams89b]. Ce type de tâches consistent à observer régulièrement un système et à minimiser, à chaque fois que cela est possible, l'erreur existant entre la sortie du modèle et la sortie réelle. Cette procédure d'apprentissage pourrait donc être considérée comme étant à la limite des domaines de l'informatique et de l'automatique. À l'inverse, la BPTT s'intéresse aux tâches pour lesquelles l'observation de la totalité des entrées est nécessaire à la définition de la sortie du réseau. Ce type d'apprentissage est donc beaucoup plus proche des problèmes de reconnaissance et de classification de séquences que ne l'est le RTRL. Le problème des *vanishing gradients* est, par ailleurs, typique de la méthode BPTT puisque l'ajustement des poids oblige à déplier le réseau et donc à parcourir un nombre de pas de temps qui peut être suffisamment grand pour que le gradient d'erreur soit très probablement au moins une fois modifié par une valeur de poids synaptique très proche de 0.

Il est bien sûr possible de trouver une équivalence entre RTRL et BPTT. Cette équivalence s'obtient naturellement lorsque le nombre de pas de temps de dépliage nécessaires à l'apprentissage de la tâche par le réseau est ramené à 1 pour un apprentissage BPTT. Cette diminution est cependant insuffisante pour trouver une équivalence parfaite puisque le RTRL n'impose pas la remise à zéro des variable au premier pas de temps qu'impose la BPTT. Il est même possible de considérer ces deux procédures d'apprentissages comme très différentes. C'est ce que fait [beaufays94] qui démontre, grâce à la théorie des graphes, que ces deux méthodes sont, en fait, duales.

Mais le RTRL n'impose a contrario pas de rétropropager un gradient à chaque pas de temps. Il est tout à fait possible d'espacer les mises à jour des poids vis-à-vis des présentations de couples entrée-cible. Le RTRL a alors tendance à se rapprocher de la BPTT même si les gradients d'erreur sont toujours calculés à partir de valeurs approximées et déterminées récursivement, contrairement à ce qui est fait par la BPTT. Cet espacement des mises à jour à été étudié, et justifié, par [catfolis93]. La justification d'une telle modification du RTRL tient à la constatation que les poids du réseau sont normalement remis à jour à chaque pas de temps quelles ques soient les caractéristiques du système dynamique présenté à l'entrée. [catfolis93] estime nécessaire d'adapter le processus de mise à jour en fonction de la forme perceptible du processus. La figure 7.8 montre deux processus différents

observés avec un même taux d'échantillonnage. [catfolis93] pense que, si le processus de gauche pourra être correctement analysé avec le taux d'échantillonnage utilisé, le réseau chargé d'analyser le processus de droite verra ses capacités améliorés par l'utilisation d'une fréquence d'échantillonnage plus faible. Cette adaptation se fait, dans [catfolis93], par la mise en place d'un compteur de pas de temps qui définit la fréquence des mises à jour vis-à-vis de la fréquence de base d'échantillonnage du signal correspondant à la présentation des données en entrée du réseau. Ce compteur, τ, ralentit donc le rythme des mises à jour du réseau.

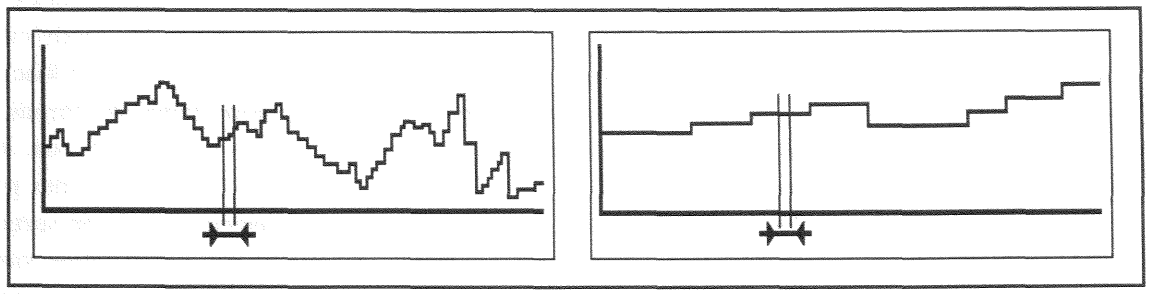


Figure 7.8 : Problème de l'échantillonnage à rythme constant de deux systèmes ne présentant pas le même comportement (d'après [catfolis93]).

La dernière idée que nous venons de voir semble assez intéressante. Elle doit, en fait, être considérée comme une redécouverte par l'auteur de la formulation des systèmes de prédiction faite par Takens dont nous avons déjà parlé au paragraphe 7.2.1.2, [catfolis93] ne semblant pas avoir eu connaissance de ces travaux. Mais, si le rapprochement entre le RTRL et la formule de Takens peut être fait, le processus de calcul du coefficient τ redéfinissant un taux d'échantillonnage sur les données d'entrée n'est toujours pas donné, pas plus par [takens81] que par [catfolis93] qui adapte ses paramètres arbitrairement. Il semble donc judicieux d'essayer de définir ce nouveau taux d'échantillonnage d'une autre manière.

7.2.5.6/ Définition d'un nouveau coefficient d'apprentissage

La définition d'un nouveau taux d'échantillonnage peut se faire de manière totalement heuristique en respectant la méthode présentée par [catfolis93]. Il est également envisageable de définir un nouveau taux d'échantillonnage par le biais de la procédure d'apprentissage qui soit directement en prise avec l'architecture du réseau. Comme nous l'avons déjà vu au paragraphe 7.2.2.3, certaines caractéristiques des filtres gamma peuvent être appréhendées par la notion d'échantillonnage, l'équation 7.6 donnant un temps d'échantillonnage tandis que l'équation 7.7 donne une période d'échantillonnage. Il est donc intéressant d'agir sur le coefficient μ pour définir un taux d'échantillonnage plus faible. Cependant, μ doit avant tout être défini en fonction de la tâche à apprendre et ne peut donc pas être fixé. Il peut cependant être contraint de manière à limiter des modifications trop importantes, fixant ainsi une approximation d'un nouveau taux d'échantillonnage. Cette contrainte pourra être imposée soit en définissant un espace restreint de liberté autour de la valeur initiale du coefficient, soit en définissant un terme supplémentaire dans les équations d'apprentissage, ce terme atténuant la modification du coefficient de feedback telle qu'elle est demandée par le gradient d'erreur.

Le deuxième intérêt que nous voyons au fait d'imposer une contrainte aux coefficients de feedback est relatif aux oscillations que nous avons observées lors des premiers apprentissages que nous avons effectué. Comme nous l'avons déjà signalé au paragraphe 7.2.5.5, ces oscillations sont provoquées par le déphasage provoqué par les moments du signal entre l'apprentissage des coefficients de feedback et l'apprentissage simultané des valeurs des poids synaptiques. Ce déphasage crée une différence entre la perception de la tâche par les poids synaptiques et la représentation du corpus par les coefficients de régression responsables de la production de traces de signal de durées et d'amplitudes variables. Ce problème est presque insoluble en l'état. La mise en

place d'un apprentissage des poids et des coefficients en alternat, par exemple, ne permet pas d'obtenir des résultats satisfaisants. Dans ce dernier cas, la procédure d'apprentissage se comporte de manière identique au cas où l'apprentissage se fait simultanément : la phase d'apprentissage des poids, à partir de coefficients de régression fixes, permet de réduire l'erreur du réseau observée mais le passage à la phase d'apprentissage des coefficients de feedback modifie ceux-ci en tenant compte de l'erreur, rendant les poids synaptiques inappropriés aux nouvelles conditions de mémorisation. La solution d'un apprentissage en deux temps n'est pas, non plus, une solution permettant d'obtenir de bons résultats. Ainsi, une première phase d'apprentissage ne modifiant que les poids, pour réduire au maximum l'erreur du réseau, qui précéderait une deuxième phase ajustant tout autant les poids que les coefficients de feedback a été testé sans grand succès.

Il existe une autre solution qui a également été mise en œuvre dans certaines autres recherches sur le modèle gamma. Cette solution consiste à introduire dans la procédure d'apprentissage un coefficient d'apprentissage supplémentaire. Ce nouveau coefficient doit en fait être considéré comme étant lié au coefficient d'apprentissage des poids connexionnistes et nous l'avons, de ce fait, baptisé atténuateur de couple. Ce terme a été choisi parce qu'il permet de rappeler la contribution positive qu'il apporte à la résolution du problème des oscillations dans le réseau par rapport aux coefficients d'apprentissage. Cet atténuateur doit être inférieur à 1 pour limiter les variations des coefficients de feedback vis-à-vis des poids. Cette contrainte ne donne cependant pas de valeur et tout atténuateur compris dans l'intervalle allant de 0 à 1 semble a priori candidat. Un atténuateur ayant une valeur très proche de 0 aura cependant tendance à pratiquement geler les coefficients de feedback, forçant les poids connexionnistes à s'adapter aux différentes capacités de mémorisation fournies par la procédure d'initialisation. Nous verrons cependant au cours de la présentation des problèmes étudiés qu'une très faible valeur de l'atténuateur peut permettre d'obtenir d'excellents résultats et que les coefficients de feedback ne sont pas totalement gelés après initialisation.

La notion d'atténuateur de couple en tant que coefficient supplémentaire d'apprentissage peut se retrouver dans deux articles traitant de l'apprentissage dans le modèle gamma ou dans un réseau qui lui est fortement apparenté. Un premier article [renals94b], très critique, permet de connaître le point de vue de l'équipe ayant mis au point une architecture connexionniste utilisant un mécanisme similaire à celui du filtre gamma et dont nous parlerons au paragraphe 7.2.6 suivant. Cet article note de manière très accentuée toute la difficulté de l'apprentissage avec un réseau gamma similaire à celui défini dans [principe93a], l'article qualifiant même cet apprentissage de non trivial ! La tâche étudiée a permis de faire une comparaison entre un perceptron à lignes de délais encastrés et un réseau gamma appliqués tous deux à une tâche de reconnaissance de la parole continue. Le réseau gamma a alors obtenu des résultats légèrement moins bons que ceux obtenus par le perceptron. De même, des tests supplémentaires effectués pour vérifier les capacités supposées d'adaptation au locuteur du modèle gamma n'ont pas donné de résultats permettant de confirmer cette hypothèse. Les auteurs concluent cependant que les résultats ne remettent pas en cause les capacités théoriques du modèle.

Le point le plus intéressant de cet article concerne la procédure d'apprentissage mise en œuvre par les auteurs pour s'assurer de la convergence dans le modèle gamma. [renals94b] souligne ainsi la nécessité de diminuer le facteur d'apprentissage η dans une proportion fixe lorsque ce facteur est utilisé pour l'apprentissage du coefficient de régression μ. L'équation 7.16 se réécrit donc avec un facteur supplémentaire, que nous baptisons ici η_μ, pour donner l'équation 7.22.

$$\Delta\mu = -\frac{\eta}{\eta_\mu}\frac{\partial E}{\partial \mu_i} \qquad \text{(Éq. 7.22)}$$

[renals94b] définit le coefficient η_μ comme étant égal à 10 ce qui signifie que le ou les coefficients de feedback varieront dans une proportion 10 fois plus faible que les poids synaptiques.

Ce coefficient de réduction, l'inverse de notre atténuateur de couple, a également été utilisé dans [sastry94]. Le réseau étudié, le réseau de neurones à mémoire (cf. chapitre 6, paragraphe 6.4.3.8), peut être facilement apparenté au modèle gamma comme nous le verrons au paragraphe 7.2.6. La procédure d'apprentissage définie pour ce réseau connexionniste particulier est similaire à la méthode RTRL mais prend, bien évidemment, la spécificité architecturale en compte. Cette procédure se voit également dotée de deux coefficients d'apprentissage, le premier, η, étant spécifique aux connexions synaptiques alors que le deuxième, η', est spécifique aux coefficients de feedback. Aucune relation n'existe a priori entre ces deux coefficients chez [sastry94] et le seul exemple de mise en œuvre du réseau permet de voir le choix des concepteurs pour des valeurs de η = 0,2 et de η' = 0,1 soit un atténuateur de couple de 0,5. Cette valeur est très élevée puisque [renals94b] avait opté pour une valeur de 0,1.

Le concept de l'atténuateur de couple n'est cependant pas omniprésent dans la littérature relative à l'apprentissage dans le modèle gamma. Ainsi, aucun article relatant les recherches effectuées dans le cercle des concepteurs de cette architecture ne parle d'un tel coefficient, toutes les procédures d'apprentissage exposées adaptant les connexions synaptiques et les coefficients de feedback avec le même coefficient d'apprentissage.

Un test effectué dans [lawrence96] comparant différentes architectures connexionnistes dont le modèle gamma ne fait pas non plus mention d'un quelconque coefficient supplémentaire. L'architecture gamma employée dans ce dernier article n'est cependant pas totalement similaire au modèle original puisque le mécanisme gamma est implanté en couche cachée et employé de concert avec des lignes de délais encastrés, approchant ainsi l'architecture définie par [kim92] et présentée au chapitre 6, paragraphe 6.4.3.5. Aucune mention du paradigme d'atténuateur n'est faite dans cet article, tendant à montrer que la procédure d'apprentissage a été parfaitement efficace. Le problème étudié est malheureusement trop simple pour permettre de tirer des conclusions générales.

7.2.6/ Réseaux apparentés

Ce paragraphe va nous permettre de présenter deux modèles qui sont architecturalement très proches du modèle gamma même si quelques nuances permettent de les distinguer tant au niveau de l'architecture que de la définition des paramètres internes.

Le plus ancien des deux modèles que nous tenons à présenter est le modèle défini dans [harrison89]. Ce modèle utilise deux échelons connexionnistes pour tenter de résoudre le problème de l'identification des phonèmes en parole continue. Il est important de remarquer, ou de rappeler, que ce modèle a été défini parallèlement à celui présenté dans [robinson89] et [robinson90a], ce dernier modèle ayant eu d'excellents résultats sur la même tâche par la suite [robinson94].

Le premier échelon permet d'identifier des unités définies comme infra-phonémiques par le concepteur, cette identification se faisant par un perceptron multicouche à deux couches cachées. Le deuxième échelon assure l'identification des phonèmes eux-même. Le mécanisme mis en œuvre à ce niveau fait appel à une récurrence locale qui respecte l'équation 7.23 et suit le schéma donné à la figure 7.9.

$$e(t) = (1-a) \cdot f(t) + a \cdot e(t-1) \qquad \text{(Éq. 7.23)}$$

Dans cette équation, *f(t)* représente l'entrée du neurone aux différents pas de temps considérés et *e(t)* représente la valeur obtenue en sortie. La valeur *e(t)* est ensuite transformée non linéairement après recombinaison facultative avec d'autres valeurs du même type [harrison89].

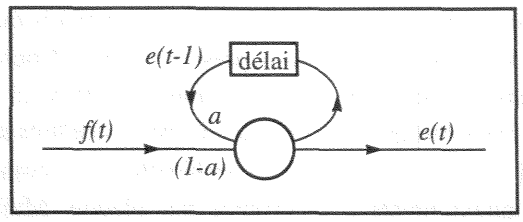


Figure 7.9 : Une unité de feedback (d'après [harrison89]).

La différence entre l'équation 7.1 et l'équation 7.23 est assez simple à comprendre mais ne constitue pas, en fait, le point le plus important à remarquer. La valeur du coefficient de feedback a de l'équation 7.23 n'est en effet pas unique pour toutes les unités du réseau. Elle n'est cependant pas définie par apprentissage puisque les différentes valeurs employées respectent les règles données par l'équation 7.24 pour le premier coefficient, a_1, et par l'équation 7.25 pour les coefficients a_i suivants.

$$a_1 = 0,25 \qquad \text{(Éq. 7.24)}$$

$$a_i = a_{i+1}^{\ 2} \qquad \text{(Éq. 7.25)}$$

Ces différents coefficients sont utilisés par des unités de feedback placées en parallèle sur une même couche et les unités associées à chacun des coefficients permettent donc de conserver des traces différentes du signal. Les coefficients sont cependant définis de manière définitive et correspondent à des mémoires ayant une profondeur toujours plus grande. Une mise en équivalence des coefficients μ avec les coefficients a_i obligerait ainsi à borner les coefficients μ sur un intervalle compris entre 0 et 0,75. De plus, le coefficient a_5 est déjà supérieur à 0,9, le coefficient a_{11} étant égal à 1 à 10^{-3} près. Ces dernières conditions prouvent que le signal en provenance des unités infraphonémiques n'est quasiment pas pris en compte par certaines unités de rang élevé, permettant d'avoir une très grande profondeur de mémoire. À l'autre extrémité du spectre, l'unité de rang 1 est équivalente à une unité gamma dont le coefficient μ serait égal à 0,75. Cette première unité est donc une unité de faible profondeur mais dont la résolution est assez mauvaise. Le modèle défini par [harrison89] est donc un modèle utilisant une mémoire à moyen et long terme.

L'autre modèle qu'il est impossible de ne pas citer est le modèle du réseau de neurones à mémoire [sastry94] dont nous avons déjà parlé au chapitre 6, paragraphe 6.4.3.8. La différence principale entre ce modèle et le modèle défini dans [vries90] porte sur l'utilisation du filtre gamma comme mécanisme annexe de mémorisation au niveau de la couche d'entrée mais également des couches cachées du réseau. Le mécanisme de filtre est cependant découplé par rapport aux neurones d'intégration synthétisant l'information en provenance de la couche inférieure. Ainsi, bien que les coefficients de régression soient ici définis par apprentissage grâce à un algorithme du même ordre que celui que nous présentons en annexe 1, les résultats obtenus sur des tâches d'ordre limité [sastry94] ne permettent pas de penser que le mécanisme de mémorisation est correctement employé.

7.3/ Développements apportés au modèle gamma

7.3.1/ Rappel de l'état de l'art

Comme nous venons de le voir au paragraphe 7.1.2, le modèle gamma est un modèle de mémoire très simple à implanter. La récurrence locale ne dépend que d'un seul paramètre, μ, et la stabilité du filtre est assurée sur un intervalle bien défini :]0,2[.

Le modèle gamma a, avant tout, été défini pour résoudre le problème de la taille de la plaque d'entrée des modèles intégrant une ou plusieurs lignes de délais encastrés. La mémoire de telles lignes est limitée dans le temps (cf. chapitre 5, paragraphe 5.4.2.2) et seul un mécanisme de récurrence locale permet de dépasser cette limitation sans entraîner de transformations architecturales aux niveaux mésoscopique et macroscopique. Ce choix peut être très facilement

critiqué. En effet, dans tout le pandémonium des réseaux récurrents que nous avons vu tout au long du chapitre 6, seuls quelques uns ont jusqu'à présent permis d'obtenir des résultats pleinement satisfaisants lors des phases d'apprentissage. Au rang de ceux ci se trouve le *Discrete Error Propagation Nework* [robinson89] qui a obtenu de très bons résultats sur des tâches de classification de voyelles [robinson94] avec un réseau dont l'architecture est comparable au modèle de Elman [elman90], un réseau connexionniste à récurrence par plaque. Mais, dans ce cas particulier, la procédure d'apprentissage a été étudiée de manière approfondie pour obtenir une méthode de descente de gradient récurrente qui, bien que n'étant pas fondamentalement différente d'autres [williams89], possède des paramètres parfaitement ajustés [robinson91]. Le mécanisme mis en œuvre dans de tels réseaux pour la représentation du temps n'est cependant pas explicite au sens où il n'est pas parfaitement identifiable.

Un mécanisme de récurrence locale permet de faire cette identification assez rapidement et, ce, même si le mécanisme mis en œuvre est complexe. Étant facilement identifiable, son ou ses paramètres de contrôle peuvent être adaptés en phase d'utilisation tout autant que lors de la phase d'apprentissage pour permettre au réseau de s'adapter à des cas particuliers de formes à classer ([pican95] pour de telles adaptations dans les réseaux connexionnistes et [mirghafori95] pour ces mêmes adaptations dans les HMM). Nous reviendrons brièvement sur ce point dans le chapitre 8.

Étant donné les caractéristiques intéressantes du modèle gamma, nous avons décidé de l'utiliser pour notre tâche de segmentation en lui apportant, cependant, quelques modifications architecturales que nous avons jugé intéressantes. Nous avons ainsi modifié les contraintes imposées à la mémoire de bas niveau qu'est la mémoire de la plaque d'entrée. Nous avons également défini une mémoire de plus haut niveau, plus abstraite, en modifiant l'architecture des neurones des couches cachées par adjonction d'unités gamma.

7.3.2/ Développement de la couche d'entrée

Le modèle gamma, tel qu'il a été défini originellement dans [vries90] ne comporte qu'un seul coefficient de régression pour l'ensemble de la plaque d'entrée. Les bandes fréquentielles définies lors de la phase de prétraitement sont donc toutes analysées avec ce seul et même coefficient. La profondeur D de la mémoire est ainsi identique pour toutes les bandes de fréquences. Cette contrainte nous semble superflue bien qu'elle puisse être défendue au titre d'une représentation correcte et uniforme des moments du signal analysé [kuo93a]. Ainsi, dans la figure 7.10, chaque unité gamma présente dans la plaque d'entrée doit utiliser le même coefficient de feedback pour répondre à l'architecture définie dans [vries90].

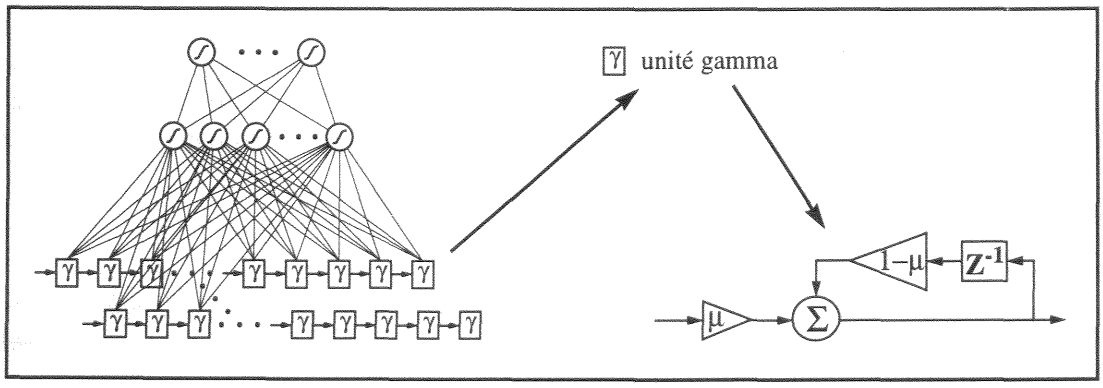


Figure 7.10 : Un réseau connexionniste gamma dont le mécanisme de mémorisation se trouve dans la plaque d'entrée (d'après [vries91a]).

La parole est cependant un phénomène variable en fonction de la fréquence. Les phénomènes de basses fréquences en parole laissent apparaître des variations globalement moins rapides que les phénomènes de hautes fréquences. La mémoire nécessaire à l'analyse des différentes bandes fréquentielles issues du prétraitement est donc différentes selon la bande considérée et il pourrait être

judicieux de définir un mécanisme de stockage tenant compte des différentes vitesses de variation et d'évolution des formes. Une des possibilités pour réaliser un tel mécanisme est la mise en place d'une plaque d'entrée composée de lignes de délais encastrés, à la manière d'un TDNN [waibel89], mais dont la longueur varierait selon l'échelle des fréquences, à la manière de l'architecture schématisée dans la figure 7.11. Certains militent pour de tels développements des techniques de reconnaissance automatique de la parole [bourlard95a].

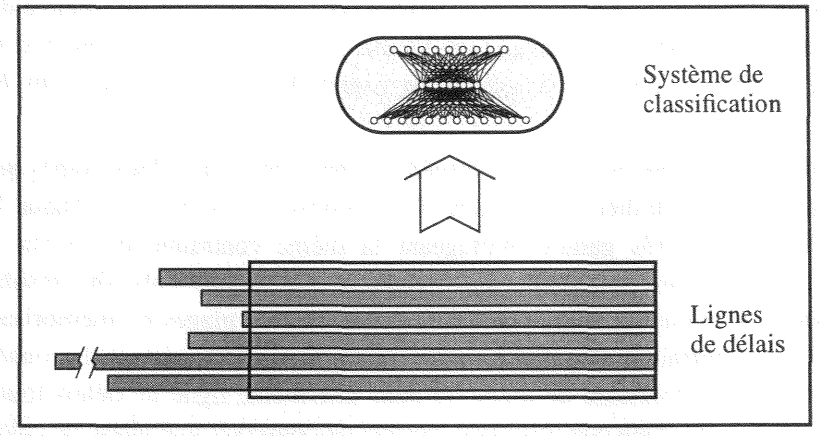


Figure 7.11 : Schématisation d'un réseau connexionniste dont la taille effective de la plaque d'entrée est variable.

Des capacités d'analyse variable des différentes bandes de fréquences fournies par l'étape de prétraitement peuvent être mises en œuvre très simplement grâce au modèle gamma. Il suffit pour cela de relâcher la contrainte d'uniformité du coefficient μ sur l'ensemble de la plaque d'entrée. Cette relaxation de la contrainte d'égalité du coefficient peut être faite en deux étapes successives.

Une première étape consiste à relâcher la contrainte d'égalité sur l'ensemble de la plaque d'entrée et de ne plus l'imposer que séparément, dans chaque ligne de délais gamma de la dite plaque. Chaque ligne de délais possède alors son propre coefficient de feedback partagé et est capable d'ajuster celui-ci plus finement par rapport aux contraintes d'apprentissage puisque les contraintes imposées par l'apprentissage ne sont désormais plus globales à l'ensemble de la plaque mais locales à chaque ligne. Les contraintes d'ajustement des coefficients ne résultent donc plus d'un compromis entre un besoin de mémoire des bandes de basses fréquences probablement important et un moindre besoin de mémoire des bandes des hautes fréquences. Il devient possible d'obtenir une plaque d'entrée dont les caractéristiques effectives sont équivalentes à celles de l'architecture présentée dans la figure 7.11 puisque la profondeur de chaque ligne de délais est fonction de son propre coefficient de feedback et que l'équation 7.11, qui permet de calculer la profondeur effective de la plaque d'entrée calculée à partir du nombre de délais d'une ligne et de la valeur de μ, reste applicable aux lignes de délais considérées séparément. Ce premier relâchement de contrainte permet d'obtenir un nombre de coefficients de feedback égal au nombre de lignes de délais dans la plaque d'entrée.

La deuxième étape de la relaxation de la contrainte d'égalité de μ dans la plaque d'entrée consiste à totalement relâcher la contrainte et à autoriser que les valeurs de μ soient différentes dans chaque unité gamma de la plaque. Cette relaxation extrême de la contrainte permet aux différentes unités de s'ajuster parfaitement en fonction du processus d'apprentissage d'une part et en fonction des valeurs initiales des coefficients d'autres part. La contrainte étant totalement relâchée et la valeur de μ étant limitée à l'intervalle $]0,1]$, il est possible de voir apparaître, au sein d'une ligne de délais, des points de rétention du signal, avec de faibles valeurs de μ, et de simples délais avec des valeurs de μ proches de 1. La possibilité de mixer des unités de profondeurs de mémoire différentes au sein d'une même ligne permet d'obtenir une image variable des phénomènes se déroulant dans une bande fréquentielle particulière. Ainsi, certaines unités de la ligne permettront d'obtenir une image exacte du signal si leurs coefficients sont proches de 1 et si ces unités sont les premières de la ligne. À

l'inverse, si une unité se trouve parmi les dernières de la ligne de délais et si son coefficient μ est proche de 0, cette unité permettra d'obtenir un signal correspondant à la mémorisation à long terme du signal traité, permettant ainsi de conserver une trace de l'activation globale lors des derniers pas de temps. Ce relâchement maximal de la contrainte d'égalité des coefficients de feedback permet d'obtenir un réseau où le nombre de degrés de liberté est très important. Ce grand nombre de degrés de liberté permet de modéliser finement les processus analysés mais peut également poser problème lors de la phase d'apprentissage puisqu'un nombre plus important de paramètres doit être déterminé sans plus aucune possibilité de connaître la direction globale du gradient d'erreur puisque la contrainte d'égalité, et donc le processus de calcul de la moyenne qui lui était associée, est supprimée.

Une troisième étape, qui est en fait intermédiaire par rapport aux deux étapes que nous venons de voir et qui n'a pas été étudiée lors de cette thèse, consiste à segmenter chaque ligne de délais en plusieurs groupes d'unités gamma partageant la même contrainte de valeur du coefficient de feedback. Il serait ainsi possible, dans certains cas dépendants du résultat du processus d'apprentissage, d'obtenir une ligne de délais segmentée en plages de mémorisation toujours plus importante. Cette contrainte d'égalité est intéressante puisqu'elle permet de conserver la possibilité d'avoir différentes profondeurs de mémoire dans une même ligne de délais tout en imposant une contrainte d'égalité qui, bien que restreinte, permet de conserver une phase de calcul de la moyenne. Le nombre des coefficients de régression est donc réduit par rapport au cas où la relaxation est totale et le processus d'apprentissage conserve l'avantage de gradients qui ne sont pas totalement locaux aux unités gamma. Il reste cependant à déterminer le nombre de groupes de délais dans la ligne et le nombre de délais dans chaque groupe, ces variables pouvant être considérées comme autant d'autres degrés de liberté dont il faudra déterminer la valeur en fonction de la tâche, tout comme pour le TDNN...

7.3.3/ Adaptation en couche cachée

Une autre adaptation des unités gamma peut être faite. Comme nous l'avons vu dans le chapitre précédent, de nombreuses architectures de neurones à récurrence locale ont été définies. La majeure partie d'entre elles n'imposent aucune contrainte sur les poids de régression, laissant ces derniers évoluer au gré de la phase de rétropropagation du gradient d'erreur. Ce manque de contrainte pourrait être à l'origine de problèmes de convergence puisqu'il est évident que si ces coefficients deviennent trop grands, la conservation des traces des activations passées peut conduire à des valeurs totalement aberrantes. À l'inverse, si ces coefficients deviennent trop faibles, la phase d'apprentissage est inefficace puisque les gradients d'erreur deviennent nuls en quelques pas de temps.

Certains modèles à récurrence locale imposent cependant ce type de contraintes aux coefficients de régression. Ainsi, les coefficients de feedback des réseaux autorégressifs (cf. chapitre 6, paragraphe 6.4.3.2) voient leurs valeurs possibles limitées à l'intervalle [0,1].

Ces limitations de coefficients nous semblent être d'un grand intérêt lors de la phase d'apprentissage puisqu'une éventuelle divergence des poids peut ainsi être évitée. Nous avons donc modifier l'architecture du neurone définie par McCulloch et Pitts [mcculloch43] en y adjoignant une unité gamma telle que définie dans [vries91]. Nous obtenons ainsi une unité neuronale dont la valeur de sortie est limitée, tout comme pour le neurone de McCulloch et Pitts, et où le traitement du temps se fait en dehors de la somme pondérée et de la non-linéarité, de manière autonome et contrainte.

L'architecture obtenue permet de se rapprocher de la définition formelle du neurone donnée dans [usher95] bien qu'apparaisse dans notre cas un coefficients de régression. Ce nouveau neurone est modélisé par l'équation 7.26. Cette équation, tout comme l'équation 6.21 du chapitre 6 relative aux réseaux autorégressifs [leighton91], est composée de deux parties. Une première partie permet de conserver une partie de l'activation courante du neurone et permet donc la mémorisation à un niveau

local. La deuxième partie correspond à la somme pondérée des activations de la couche inférieure telle qu'elle est également définie dans [mcculloch43].

$$y_{n,j,t} = (1-\mu)\, y_{n,j,t-1} + \mu\, f\left(\sum_i w_{ji} y_{n-1,i,t}\right) \qquad \text{(Éq. 7.26)}$$

Cette équation est schématisée dans la figure 7.12. Les éléments des lignes de délais de la plaque d'entrée sont des unités gamma telles qu'elles ont été définies dans la figure 7.10. Les unités de la couche cachée reprennent la définition de l'équation 7.26.

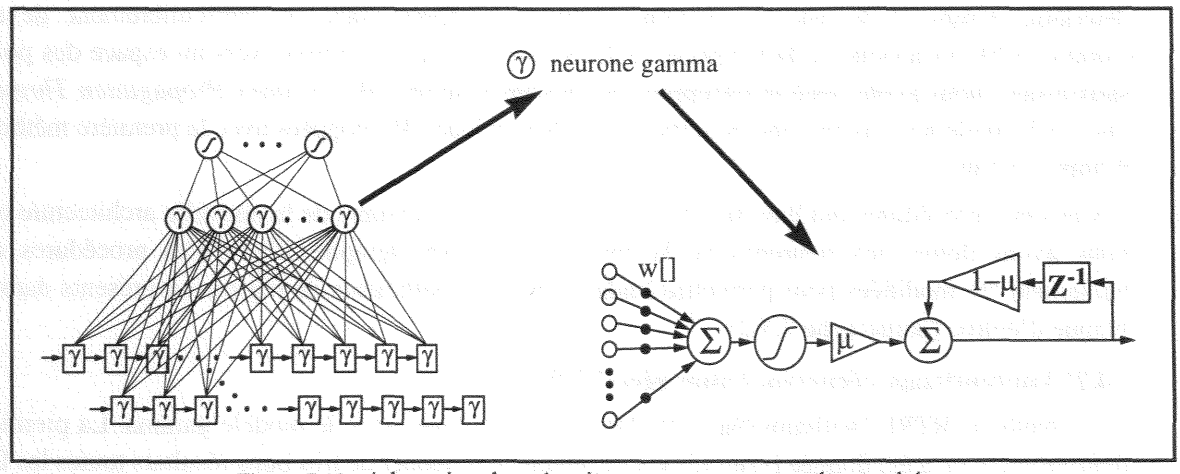


Figure 7.12 : Adaptation du mécanisme gamma aux couches cachées.

7.4/ Définition de la procédure d'apprentissage

7.4.1/ Problématique

Le processus d'apprentissage est responsable des performances du réseau lors des phases d'utilisation. Il est le point de passage obligé entre le réseau dans son état d'ébauche, c'est à dire une architecture définie dont les poids sont initialisés aléatoirement, et un réseau pleinement efficace et dont les poids ont convergé vers un état permettant de parfaitement analyser les signaux d'entrée et d'en donner une bonne classification.

Le processus d'apprentissage dans les réseaux connexionnistes dont les neurones respectent la définition de McCulloch et Pitts [mcculloch43] est désormais bien maîtrisé. Le processus de rétropropagation du gradient d'erreur est facilement compréhensible et différentes techniques, qui sont plus du ressort de l'heuristique que d'une théorie mathématique, ont permis d'améliorer la qualité et la vitesse de l'apprentissage. Mais les tâches apprises font, le plus généralement, partie du domaine de l'approximation de fonction. Il s'agit de trouver, pour des valeurs d'entrée fixées et pour la sortie associée à ces entrées, un ensemble de valeurs, les poids, qui permettent d'obtenir la sortie donnée correspondant aux entrées en fonction de l'architecture fixée. Ce processus d'approximation se fait de manière isolé, les couples à classer étant indépendants les uns des autres. La seule interdépendance existant entre les différents couples de valeurs se trouve dans la phase de rétropropagation de l'erreur, lorsque l'apprentissage se fait en mode *batch*, puisque l'erreur à rétropropager correspond alors à la somme des erreurs des différents couples de valeurs que le réseau doit apprendre à classer.

L'interdépendance entre deux formes que le réseau doit apprendre à classer successivement n'est donc pas l'objectif initial de la méthode de rétropropagation du gradient d'erreur et il faut effectuer des modifications architecturales et passer par des artifices algorithmiques pour mettre en place de telles capacités d'apprentissage de l'interdépendance.

Les modifications architecturales peuvent être de plusieurs types comme nous l'avons vu au chapitre 6. Le problème devant lequel nous nous sommes trouvés nous a poussé à effectuer un choix

de représentation de l'interdépendance que nous croyons être judicieux pour le domaine de la reconnaissance automatique de la parole en général et pour la représentation de durées moyennes en particulier. Mais, comme il a été vu au paragraphe 6.5, le processus d'apprentissage est encore loin de donner pleine satisfaction bien que de nombreuses méthodes aient été étudiées. Nous nous sommes donc efforcés de trouver une méthode d'apprentissage pour le modèle gamma qui soit efficace et qui nous donne pleine satisfaction. Deux types d'algorithmes ont ainsi été utilisés. Nous avons tout d'abord testé la méthode d'apprentissage récurrent en temps réel (RTRL, *Real Time Recurrent Learning*) à laquelle nous avons apporté quelques modifications heuristiques. Dans un deuxième temps, à la vue de certains résultats d'apprentissage et particulièrement devant l'imprévisible incapacité de la méthode RTRL à faire converger le réseau vers un espace des poids satisfaisant, nous avons testé la rétropropagation dans le temps (BPTT, *Back Propagation Through Time*) à laquelle nous avons ajouter certaines des heuristiques développées avec la première méthode d'apprentissage.

Ces deux procédures ont bien sûr été modifiées pour tenir compte de la nouvelle architecture que nous avons donné aux neurones de la couche cachée (paragraphe 7.3.3). Ces procédures ont également été modifiées pour permettre l'adaptation des coefficients de régression présents dans la plaque d'entrée (paragraphe 7.3.2).

7.4.2/ Apprentissage récurrent temps réel (RTRL)

La méthode RTRL [williams89a] a été largement employée avec le modèle gamma. La première définition qui en a été donnée [vries92] n'a cependant pas donner de très bons résultats toutes les fois où elle a été étudiée [renals94b]. Cette méthode se fonde en effet sur une approximation des gradients d'erreur qui semble être néfaste à l'apprentissage des valeurs correctes des coefficients de régression, comme nous l'avons déjà mentionné.

Nous avons également rencontrés quelques problèmes qui nous ont poussés à utiliser quelques heuristiques existantes et à en essayer d'autres. Parmi les heuristiques existantes que nous avons utilisées se trouve :

- le coefficient d'apprentissage qui permet de n'appliquer aux poids connexionnistes qu'une partie du gradient d'erreur calculé en sortie du réseau ou en couche cachée. Nous avons, la plupart du temps, utilisé un coefficient d'apprentissage de 0,9 qui est la valeur la plus largement répandue dans la communauté,
- le momentum qui permet de limiter les modifications en sens contraires des poids par prise en compte d'une partie du gradient d'erreur calculé au pas de temps précédent. Ce momentum permet de limiter les oscillations lors de l'exploration de l'espace des poids. Nous avons généralement utilisé une valeur standard de 0,3,
- le delta minimum qui permet de ne pas modifier les poids connexionnistes d'un réseau lorsque la valeur du gradient est trop faible vis-à-vis de la valeur minimale du delta déterminée empiriquement. Cette valeur minimale permet d'éviter l'*overtraining*, ou surapprentissage, qui se caractérise par un trop bonne adéquation des poids vis-à-vis des formes présentes dans le corpus d'apprentissage et une perte de généralisation caractérisée par un taux de reconnaissance des formes du corpus de validation allant en s'amoindrissant,
- la valeur d'élimination du méplat, *flat spot elimination*, petite valeur suffisament négligeable pour ne pas troubler le processus d'apprentissage mais suffisament importante pour empêcher le blocage de processus d'apprentissage sur le point nul de la dérivée de la fonction non linéaire, dérivée utilisée pour le calcul des gradients en couches cachées.

Parmi les heuristiques que nous avons définies et testées se trouvent :

- la définition du nombre de deltas à calculer avant toute modification des poids du réseau. Cette heuristique nous a été inspirée par [catfolis93] et sa redécouverte du principe énoncé par

[takens81]. Malheureusement, la détermination d'un nombre de pas de temps pour la mise en œuvre de cette heuristique s'est révélé être beaucoup moins évident que ne le laissait penser la présentation et plus particulièrement les schémas de la figure 7.8. La parole est un phénomène quelque peu plus complexe que les courbes présentées et l'échec relatif de cette heuristique n'est pas étonnant. Nous n'avons, en tout cas, pas observé d'améliorations notables lors d'essais avec différents pas,

- la définition d'un *boot* de convergence. Cette phase de convergence permet de minimiser l'erreur globale du réseau par une première phase d'apprentissage des seuls poids synaptiques du réseau. Cette première phase en précède une seconde où sont ajustés tant les poids synaptiques que les coefficients de régression. Cette heuristique a comme inconvénient majeur d'adapter les poids synaptiques à des coefficients de feedback qui ne sont peut-être pas les plus adaptés à la tâche puisqu'ils sont définis par initialisation aléatoire,
- la mise en place d'un apprentissage en alternat. Cette technique d'apprentissage permet d'effectuer un apprentissage en découplant totalement la phase d'ajustement des poids synaptiques et la phase d'ajustement des coefficients de régression. L'une et l'autre de ces phases s'effectuent en alternance pour tenter de minimiser les oscillations pouvant se produire lorsque les poids et les coefficients sont mis à jour simultanément. Cette heuristique n'est pas très efficace et l'accroissement du nombre des mises à jour effectuées dans chaque phase ne permet pas de stopper les phénomènes oscillatoires,
- la mise en place d'un atténuateur de couple. Cet atténuateur, qui n'est rien d'autre qu'un coefficient supplémentaire d'apprentissage, permet de modifier les coefficients de régression de manière beaucoup plus lente que les poids synaptiques. Nous avons déjà parlé de cette heuristique au paragraphe 7.2.5.6,
- l'initialisation des poids dans une plage restreinte. Un poids connexionniste et, plus encore, un coefficient de feedback sera d'autant plus difficile à modifié que sa valeur d'initialisation sera proche d'une des bornes de l'intervalle autorisé. Pour tenter de pallier ce problème, particulièrement gênant dans le cas des coefficients de régression, nous avons mis en place un ensemble de valeurs permettant de contraindre les valeurs issues d'une fonction de génération de nombres aléatoires suivant une loi uniforme. Ces différentes valeurs nous permettent de centrer l'initialisation autour du milieu de l'intervalle autorisé dans une plage dont la largeur peut varier en fonction des paramètres fournis. Cette heuristique nous a permis de réduire quelque peu le nombre des apprentissages non convergents qui peuvent cependant encore se produire avec une assez forte probabilité.

Toutes les heuristiques que nous venons d'exposer ne sont pas toutes utilisées bien que toutes aient été testées. Le delta minimum, la valeur d'élimination du méplat, le nombre de deltas à calculer, le boot de convergence ou l'apprentissage en alternat sont autant d'heuristiques que nous n'avons finalement plus utilisé devant le peu d'améliorations qu'elles avaient apportées.

À l'inverse, nous pensons que les heuristiques du coefficient d'apprentissage, du momentum, de l'atténuateur de couple et d'initialisation des poids dans un espace restreint sont de bonnes heuristiques puisqu'elles ont permis de gagner en probabilité ou en qualité de convergence lorsqu'elles étaient mises en œuvre tant avec l'apprentissage récurrent temps réel qu'avec la rétropropagation dans le temps.

7.4.3/ Rétropropagation dans le temps (BPTT)

La rétropropagation dans le temps, BPTT [werbos90], est une méthode d'apprentissage par rétropropagation du gradient d'erreur permettant de travailler sur les valeurs exactes des gradients. Cette capacité à travailler sur des gradients exacts oblige, en contrepartie, à stocker les états du réseau sur un nombre de pas de temps fonction de la tâche à apprendre.

Les heuristiques que nous avons utilisées avec cette méthode d'apprentissage sont les mêmes que

celles que nous avons présentées au paragraphe précédent. Certaines d'entre elles ne peuvent cependant pas être appliquées du fait de la nature même de la BPTT qui est foncièrement différente de la méthode RTRL. Ainsi, l'heuristique définissant le nombre de deltas à calculer avant la mise à jour effective des poids n'a ici aucun sens puisqu'elle est l'essence même de la BPTT.

La base algorithmique de la BPTT pose même un problème inverse à celui du RTRL puisque les segments de parole correspondant soit à une classe, soit à un phonème, ont une signification pendant toute la durée du segment alors que la BPTT est plus adaptée aux phénomènes possédant une signification finale c'est à dire une signification qui ne se réalise effectivement que lorsque le phénomène a été présenté dans son ensemble.

Cette caractéristique nous a en particulier permis de grandement améliorer les résultats de nos apprentissages sur des tâches de reconnaissance de séquences, bruitées ou non.

7.5/ Exposé des tâches étudiées

7.5.1/ Présentation

Notre objectif principal est de pouvoir modéliser une durée. Cette modélisation doit nous permettre d'obtenir un mécanisme d'inhibition retardée qui soit à même de définir une durée moyenne modulable en fonction des constatations phonétiques. Nous espérons ainsi obtenir de meilleurs résultats de segmentation que ceux obtenus avec la méthode de segmentation exposée dans le chapitre 4. L'intégration des connaissances phonétiques et des connaissances temporelles au sein d'un même réseau connexionniste devrait, par ailleurs, permettre d'obtenir des résultats d'une meilleure qualité par rapport à une architecture où les phases de classification phonétique et de segmentation temporelle seraient disjointes. L'intégration de ces deux types de connaissances au sein d'un même mécanisme devrait permettre à chacune des sources de connaissance de tempérer des décisions qui pourraient être trop brutales autrement. Ainsi, le découpage temporel, s'il constituait une étape à part entière, ne pourrait se faire que sur la base de la durée moyenne, sans aucune considération vis-à-vis de l'écart-type qui peut cependant être important (de l'ordre de 15% pour le corpus TIMIT selon [mirghafori95]).

Avant d'étudier les capacités du modèle gamma sur des tâches de segmentation de la parole, nous avons testé ses capacités de manière assez formelle avec des séquences abstraites. Le but de ces tests est de valider les capacités de mémorisation ainsi que les capacités de modélisation temporelle du modèle gamma dans ses différentes configurations. Ces différents tests ont également permis de tester les capacités d'apprentissage des différents algorithmes développés, des modifications que nous y avons apporté et des heuristiques que nous y avons adjoint.

Une première série de tests a donc permis de vérifier les capacités de mémorisation d'un réseau composé d'unités et de neurones gamma, la mémorisation devant être faite à partir de séquences simples, bruitées ou non. Dans ce type de tests, seul un élément de la séquence est significatif, les autres éléments de la séquence, ou le bruit, servant de distracteurs. Ce type de tests a été mené avec des séquences de longueur variable.

Une deuxième série de tests sur les séquences a permis de vérifier les capacités de modélisation temporelle du réseau. Il ne s'agit donc plus ici d'effectuer la simple mémorisation d'un des premiers éléments de la séquence sur un nombre de pas de temps variable. Le but de ces tests est de vérifier que le réseau est capable d'effectuer une tâche de classification correcte en tenant compte de l'ensemble des données fournies en entrée. Cette série de tests a été menée sur un sous-ensemble du code ASCII [cerf69]. Une première partie des tests a permis de vérifier que la classification des codes pouvait être faite alors que la deuxième partie des tests a permis de juger des capacités du réseau sur une tâche de transformation de la représentation séquentielle d'une série temporelle en représentation parallèle.

7.5.2/ Étude de tâches de mémorisation simples

Le modèle gamma fait partie des modèles connexionnistes récurrents bien que cette récurrence ne soit que locale au neurone. Les réseaux récurrents de tous types sont aujourd'hui étudiés de manière assez formelle car leurs possibilités ne sont pas encore très bien connues. La prise en compte d'une récurrence provoque en effet l'ouverture du domaine des réseaux de neurones vers la théorie des systèmes dynamiques et la théorie du chaos (cf. paragraphe 7.2.1). Toute la phase d'apprentissage ainsi que toute phase d'utilisation, même et surtout courte, deviennent dépendantes de notions telles que l'initialisation aléatoire et les trajectoires dans l'espace d'état.

Pour mieux comprendre les mécanismes mis en œuvre par les réseaux récurrents et pour arriver à surmonter les écueils posés par de tels problèmes, certains chercheurs se sont mis à étudier intensivement l'application des réseaux récurrents à des tâches d'apprentissage et de simulation de problèmes dynamiques simples au rang desquels se trouvent les automates d'états finis (cf. chapitre 6, paragraphe 6.3.6). Ces automates sont des outils très largement utilisés en informatique, la palette de leur utilisation s'étendant de la conception de circuits imprimés à la définition des langages de programmation où ils permettent de définir la syntaxe et d'associer la sémantique sous-jacente. La notion d'automate peut même se retrouver dans des théories comme la programmation dynamique et les chaînes de Markov, bien que la sémantique des états y soit plus complexe et que les transitions aient un rôle différent. Les automates d'état finis se caractérisent cependant par une définition simple. Un automate O est l'implantation d'un langage fini et régulier L. Cet automate O peut être défini par un ensemble de 5 variables : <A, E, I, F, T> où $A = \{a_1, a_2, ..., a_m\}$ est l'alphabet du langage implanté par l'automate ; $E = \{e_1, e_2, ..., e_m\}$ est l'ensemble des états de l'automate, cet ensemble étant par définition fini ; $I \in E$ est l'état initial de l'automate lors de l'analyse d'une chaîne faisant partie ou non du langage L ; $F \subseteq E$ est l'ensemble des états terminaux de l'automate, ou états d'acceptation, qui permettent de savoir si la chaîne analysée respecte les règles de L et $T : E \times A \rightarrow E$ représente l'ensemble des transitions possibles dans le graphe. Suite à cette définition d'un automate, on dit qu'une chaîne C est acceptée par l'automate O qui représente le langage L si un état d'acceptation $\in F$ est atteint après la fin de la lecture de la chaîne C.

Les automates peuvent grandement varier selon la définition de L mais leur structure peut le plus souvent être représentée sous forme de graphe, les langages les plus simples pouvant se contenter d'une structure arborescente. Pour vérifier qu'une chaîne est bien dans le langage, il faut parcourir le graphe orienté implantant l'automate, ce graphe comportant des états d'acceptation et des états de rejet, la lecture du caractère de fin de chaîne permettant de connaître l'état atteint dans le graphe. Dans certains cas, le graphe d'un automate comporte des transitions d'un état sur lui-même. Il se peut également qu'un langage régulier soit implanter à l'aide de plusieurs automates s'imbriquant les uns dans les autres pour représenter des concepts plus ou moins abstraits (cf. chapitre 5, paragraphe 5.2.2.2). C'est en particulier le cas lors de l'élaboration de compilateurs de langages informatiques de haut niveau où il peut être intéressant de repérer des blocs syntaxiques en vue de leur optimisation.

Les automates d'états finis, qui sont intéressants à étudier dans la domaine de la simulation de systèmes dynamiques par des réseaux de neurones, sont généralement des graphes peu complexes. Deux types d'automates sont tout particulièrement intéressants : les automates implantant des chaînes de grande longueur, nécessitant donc des traitements de longue haleine avant toute acceptation, et les automates implantant des récurrences sur un état ou sur un petit nombre d'états.

L'analyse de longues chaînes de "caractères" permet de se poser le problème de la rémanence de l'information passée dans le réseau. L'analyse d'une chaîne peut, par exemple, nécessiter de stocker de l'information à l'aide d'une pile externe au réseau (cf. chapitre 6, paragraphe 6.3.6 et 6.3.7). Un deuxième problème est l'encodage stable d'états au sein du réseau. L'implantation d'un système dynamique dans un réseau présentant un nombre de degrés de liberté trop important peut en effet

interdire un encodage stable de la tâche étudiée et créer des variations dans l'espace de phase, variations génératrices d'erreurs lors de l'étude de grammaires à graphe cyclique (chapitre 6, paragraphe 6.3.8).

Ces différents problèmes ont déclenchés des recherches dans plusieurs équipes, chacune ayant prôné une solution qui ne permet pas d'obtenir un modèle très général bien que la solution proposée soit, à chaque fois, acceptable.

Nous avons nous aussi choisi de tester notre modèle sur des tâches de modélisation d'automates. Le modèle gamma est cependant mal adapté à ce type de problèmes : la récurrence, qui est uniquement locale au neurone, ne permet pas de véritablement modéliser un automate à la manière des modèles présentés au chapitre 6, paragraphe 6.3. Nous ne pouvons pas, en effet, espérer dépasser de manière significative les capacités du modèle BPS, présenté chapitre 6, paragraphe 6.3.8, le modèle BPS ayant lui-même des capacités limitées de modélisation.

Les tâches que nous allons étudier sont donc plus en rapport avec la reconnaissance de séries temporelles linéaires qu'avec la reconnaissance de mots d'une grammaire à contexte libre. Il est hors de question, ici, de traiter des problèmes comme les grammaires de palindromes ou de langages bien parenthésés. Nous ne disposons en effet pas, avec l'architecture gamma, d'un réseau apte à modéliser un automate d'états fini. Les problèmes étudiés ici sont donc plus du domaine des mécanismes de réponse différée que du domaine de l'analyse grammaticale. Nous allons cependant parfois utiliser dans le suite de cette présentation la notation graphique utilisée pour la présentation des automates. Ce choix a été fait pour alléger un peu un exposé qui aurait, autrement, nécessité un développement plus long et fastidieux à lire. L'auteur espère d'ailleurs que la lecture s'est avérée assez agréable jusqu'à présent...

Un automate peut être représenté par une représentation en graphe faisant intervenir quatre types de symboles (cf. figure 7.13). Le symbole du double triangle orienté vers le haut définira ici l'état initial de l'automate, état à partir duquel commence toute analyse d'un mot. Les états de transition au sein du graphe sont représentés par de simples ronds. Ces ronds correspondent à un état intermédiaire lors de l'analyse pour lequel il n'est pas encore possible de dire si la chaîne en cours d'analyse est ou non correcte. Deux états particuliers marquent la fin du processus d'analyse effectué par l'automate. Un premier état, que nous avons représenté par deux cercles imbriqués, marque l'état final d'acceptation de la chaîne. Le passage par cet état signifie que la chaîne qui a été observée en entrée par l'automate correspond à un des mots du vocabulaire représenté. Un deuxième état particulier, que nous avons représenté par deux triangles imbriqués orientés vers le bas, marque lui l'état de rejet : la chaîne ou la sous-chaîne qui vient d'être analysée par l'automate ne correspond pas à un mot du vocabulaire représenté par l'automate. Les états d'acceptation et de rejet peuvent être multiples. Nous avons choisi, dans la figure 7.13, de n'utiliser qu'un seul état de rejet alors que l'utilisation de plusieurs états différents permet une analyse a posteriori de l'erreur.

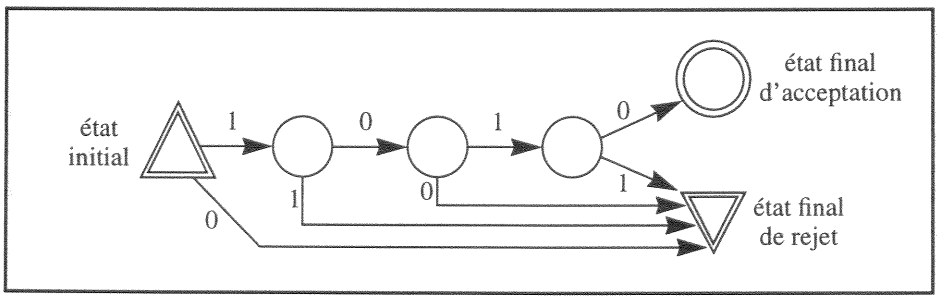


Figure 7.13 : Présentation d'un automate et des symboles associés.

Les études que nous avons réalisées sur des tâches de mémorisation simple ont pour but de nous permettre d'émettre un premier jugement sur le comportement d'un réseau gamma. Les tâches étudiées sont simples car nous voulions avoir une complexité minimale pour que le problème

lui-même ne soit pas un facteur de gêne lors de l'analyse des résultats.

Ces études peuvent être scindées en deux parties distinctes. Une première série d'études nous a permis d'observer le comportement d'un réseau sur une tâche de classification temporelle simple. La classification de deux séquences simples a tout d'abord permis de voir vers quelles valeurs les facteurs de gain du réseau convergeaient. Une analyse de ces valeurs en fonction de la connaissance de l'architecture gamma permet de voir le comportement émergeant du réseau tout au long de la phase d'apprentissage. Cette étude a également été l'occasion d'analyser les réactions du réseau à ces mêmes tâches mais bruitées par un bruit uniforme. Le but in fine de l'implantation et de l'étude de ce réseau étant la segmentation dans la parole dans le bruit, il était intéressant de faire cette étude au regard de celle qui avait été faite sur des séquences non bruitées. L'étude de la classification de séquences temporelles bruitées est en effet intéressante puisque les séquences que nous avons définies ne sont pas redondantes en information alors que le parole l'est. Ainsi, alors que les séquences ne pourront être discriminées que par l'observation d'une seule caractéristique bien particulière, la parole pourra être classée en fonction de plusieurs caractéristiques redondantes, la parole étant un signal redondant par nature [calliope89]. De bonnes capacités de traitement des séquences bruitées devraient donc laisser entrevoir de bonnes capacités de segmentation de la parole bruitée.

La deuxième série d'études porte sur les caractères ASCII dont nous avons sélectionné une sous partie composée des 26 lettres majuscules de l'alphabet. Ce choix se justifie pour une étude des capacités du réseau sur une tâche plus ardue que la simple reconnaissance de 2 séquences puisque les informations nécessaires à la classification sont ici plus nombreuses sans pour autant être redondantes. Le nombre de séquences augmente par ailleurs pour passer de 2 à 26. Deux types d'études ont été réalisées : une première concerne la classification des séquences ASCII (le réseau est alors constitué de 26 sorties) alors qu'une deuxième étude a porté sur le transcodage de ces mêmes séquences c'est à dire le passage d'une représentation séquentielle, ou sérielle, à une représentation parallèle (le réseau ayant, dans ce cas, 8 sorties).

7.5.3/ Étude des capacités de segmentation de la parole

Les deux types de tests sur les séquences temporelles ayant été effectués et l'architecture, ainsi que la phase d'apprentissage, étant validées dans une moindre mesure, nous avons effectué des tâches de segmentation et de classification du signal de parole.

Comme lors de notre étude de l'étape de segmentation dans le chapitre 4, nous avons effectué des tests de segmentation selon plusieurs classifications. Nous avons ainsi étudié les capacités de segmentation en grandes classes du réseau gamma avant d'étudier les capacités de segmentation et de reconnaissance de phonèmes comme, par exemple, les occlusives.

Ayant, encore une fois, rencontré des problèmes lors de la segmentation en grandes classes du fait de l'agglomération de plusieurs étiquettes de la même classe les unes à côté des autres, nous avons également effectué des tests avec une segmentation modifiant l'étiquetage manuel en diminuant la longueur des étiquettes par déplacement des bornes de début et de fin vers le milieu de celles-ci. Cette segmentation synthétique permet de créer artificiellement des segments dont le début et la fin sont visibles par le réseau. Ce type de manipulation ne pouvait, par ailleurs, pas être réalisé avec les réseaux statiques mis en œuvre dans le chapitre 4 puisqu'elle peut gêner l'apprentissage des caractéristiques phonétiques et n'aurait, dans ce premier cas, apporté aucune information temporelle puisque le réseau était alors incapable de les représenter.

Nous avons également effectué quelques tests de reconnaissances sur les occlusives qui sont des phénomènes acoustiques beaucoup plus difficiles à classer que les voyelles du fait de la faible durée de la partie du signal contenant les informations discriminantes. Cette dernière étude a été immédiatement réalisable grâce au choix que nous avons effectué lors de nos premiers tests sur la segmentation et la reconnaissance de la parole. Ce choix est relatif au pas de temps entre le calcul de

2 trames de coefficients issues de la phase de prétraitement. Nous avons en effet choisi de tester notre réseau avec des données calculées toutes les 4 millisecondes alors que le pas standard de calcul est de 10 *ms* et que la tendance actuelle de la RAP préconise l'emploi de pas de temps plus importants. Ce choix pour un pas de temps très faible a déjà été effectué dans d'autres recherches sur la reconnaissance des occlusives [fanty90]. Nous l'avons fait pour des raisons initialement différentes. En effet, notre nouvelle architecture devant être capable d'implanter une mémoire, nous avons jugé bon de la forcer à le faire en ne lui laissant pas le choix. La plaque d'entrée du réseau, utilisée lors des tests relatifs à la parole, est constituée de 12 canaux, correspondant aux coefficients Mel cepstres issus de la phase de prétraitement du signal, chacun des coefficients étant stockés dans une ligne de 6 délais gamma encastrés. Chaque trame de coefficients MFCC étant calculée à partir d'une fenêtre de 32 *ms* de signal temporel et les 6 trames utilisées étant calculées à des intervalles de 4 *ms*, le signal représenté dans la plaque d'entrée du réseau gamma représente donc un total de 52 *ms* lorsqu'aucun mécanisme de mémorisation n'est implanté. Cet intervalle de 52 *ms* est relativement court et [waibel89] fournit par exemple une fenêtre de 150 *ms* en entrée de son TDNN pour une tâche de reconnaissance des occlusives. Nous espérons donc, par notre choix pour une fenêtre très réduite, obliger le réseau à implanter de la mémoire par le biais du mécanisme de filtre passe-bas.

7.6/ Tâches de mémorisation simple

7.6.1/ Présentation des séries temporelles

7.6.1.1/ Introduction

Les différentes séquences mises en œuvre lors des expérimentations sur les séries temporelles simples sont présentées dans ce paragraphe. La première série a permis de faire une étude initiale sur les capacités du réseau à résoudre des problèmes de classification de séries temporelles. Qui plus est, cette première série a permis de faire une étude préalable des capacités de l'architecture gamma à reconnaître des séquences bruitées par un bruit uniforme. La série temporelle a ainsi été bruitée jusqu'à des rapports signal sur bruit de 5 décibels, ce qui est très faible au regard des capacités prévisibles du réseau. Nous reviendrons sur ce point ultérieurement.

Il est à noter que tous les éléments des différentes séries présentées utilisent les valeurs -1, 0 et +1. Ce choix d'une représentation sur un ensemble de trois valeurs nous semble judicieux puisqu'il permet de disposer, en plus des valeurs positives et négatives marquant les événements, d'une valeur nulle permettant de représenter un non-événement. Ce choix est également justifié par certaines études qui ont été réalisées par ailleurs sur les prétraitements de séries temporelles [omlin94].

Les séries temporelles qui suivent seront présentées de deux manières : nous présenterons dans le texte ces séries comme une suite de valeurs, à la manière des valeurs qui ont été fournies au réseau, et nous donnerons, de manière complémentaire, une représentation graphique de chacune de ces suites pour améliorer la lisibilité des événements les constituant.

7.6.1.2/ Première type de série temporelle

La première série temporelle est d'une grande simplicité. Elle est constituée de deux classes qui ne peuvent être distinguées que par le premier élément. Ainsi, la première classe est constituée par la suite {1, 0, 0, 1} tandis que la deuxième classe est constituée par la suite {-1, 0, 0, 1}. Cette série est également présentée graphiquement à la figure 7.14. La morphologie de cette série temporelle permet de juger de la qualité de la rétention de l'information. La classification dans une classe ou l'autre ne peut en effet s'effectuer que si la valeur présentée à t_0 est correctement conservée pendant t_1 et t_2 et de manière suffisamment bonne pour que la valeur présentée à t_3 n'empêche pas la classification par un masquage trop fort de la valeur d'activation résiduelle.

Si la notion de masquage ne semble pas trop importante a priori, elle a cependant une importance non négligeable dans le cas de séries temporelles bruitées. Dans le cas où la série temporelle n'est pas bruitée, la classe pourrait tout à fait être reconnue par simple classification de la différence finale

au sein des unités cachées. Mais cette différence aura cependant tendance à s'amenuiser au fur et à mesure que le rapport signal-sur-bruit diminuera dans les expériences sur des séries bruitées.

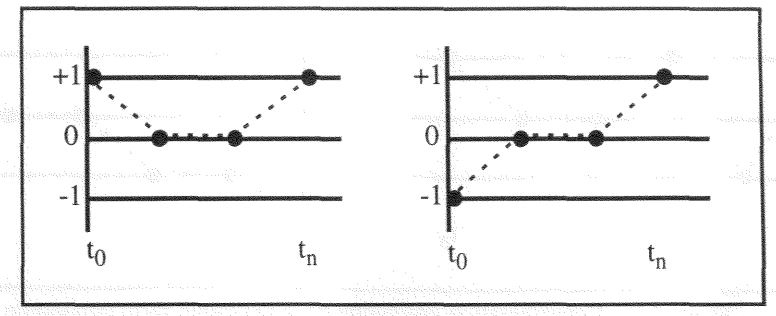


Figure 7.14 : Représentation graphique de la première série temporelle.

7.6.1.3/ Deuxième type de série temporelle

La deuxième série temporelle que nous avons étudiée est plus complexe que la première que nous venons de voir. L'élément discriminant entre les deux classes n'est plus placé en tête de la série mais en troisième position (t_2) dans une série de six éléments. Ainsi, la première classe est constituée par la suite {1, -1, 1, 0, 1, 1} tandis que la deuxième classe est constituée de {1, -1, -1, 0, 1, 1}. La représentation graphique de cette série est donnée à la figure 7.15. Les premiers éléments de cette liste n'interviennent que pour créer un distracteur dans la série mais leur rôle est en fait limité puisque la valeur du premier élément est presque totalement annihiler par le deuxième élément de la séquence. La fin de cette série temporelle varie par rapport à celle de la première puisque les deux dernières valeurs présentées sont égales à 1. Les deux derniers éléments de la séquence ont donc une grande importance dans la valeur finale des unités du réseau. La valeur présentée à t_2 nécessitera donc que les paramètres de mémorisation soient finement ajustés et traités efficacement par le réseau puisque l'information utile à la classification sera très peu présente dans les valeurs des unités gamma à la fin de la présentation de la séquence.

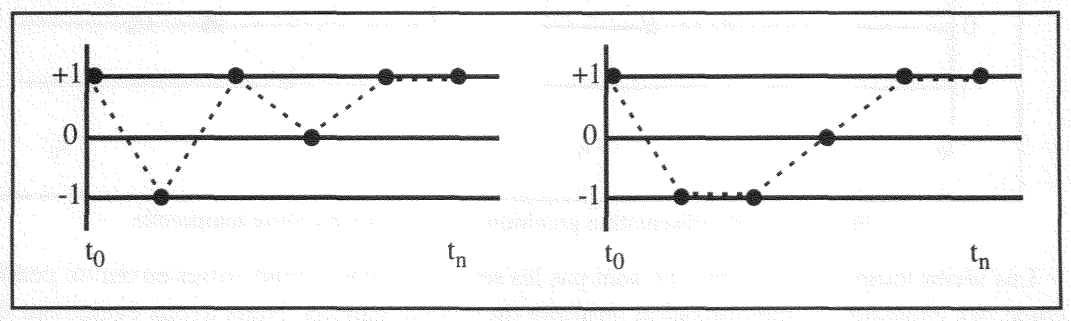


Figure 7.15 : Représentation graphique de la deuxième série temporelle.

7.6.1.4/ Troisième type de série temporelle

Le troisième type de séries temporelles que nous présentons maintenant est une modification de la deuxième que nous venons de présenter. Ces deux types de séries ne varient que par la valeur du cinquième et avant dernier élément de la liste qui n'est plus égal à 1 mais à 0. La mémorisation de l'élément discriminant de la classe devrait donc être facilité puisque seul l'élément t_5 perturbe la mémorisation. La première classe de cette série est donc constituée de la suite {1, -1, 1, 0, 0, 1} tandis que la deuxième classe est constituée de la suite {1, -1, -1, 0, 0, 1}. La figure 7.16 présente graphiquement ces deux séries temporelles.

Le passage de la valeur t_4 de +1 à 0 prend son importance dans le cas où le signal est bruité puisqu'un bruit de cinq décibels de RSSB peut aussi bien faire varier cet élément de manière à ce qu'il soit presqu'égal à +1 tout autant qu'il peut le faire varier de manière à ce qu'il s'approche de la valeur négative maximale. La mise en place d'une plage de deux valeurs nulles successives peut

donc grandement gêner l'apprentissage et la reconnaissance.

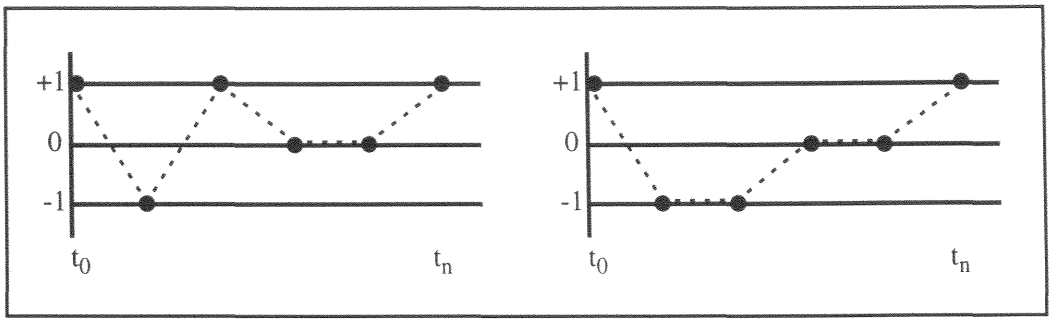


Figure 7.16 : Représentation graphique de la troisième série temporelle.

7.6.1.5/ Quatrième type de série temporelle

Le quatrième et dernier type de séries temporelles, enfin, permet de juger de l'efficacité de l'oubli d'informations. Les deux premiers éléments de la suite sont ici, pour les deux classes, égaux à 1. Le troisième élément varie et constitue l'élément discriminant des deux séries. L'accumulation des deux premières informations provoque une forte accumulation de potentiel dans les unités gamma. Le troisième élément, t_2, ne pourra être pris en compte correctement que si la mémorisation est assez faible pour permettre à t_2 de correctement surclasser les informations précédentes. La mémorisation doit cependant être assez bonne pour permettre à l'information du temps t_2 d'être conservé par le réseau jusqu'au temps t_5, dernier étape des séquences. Les deux séries temporelles sont donc constituées des suites {1, 1, 1, 0, 0, 1} et {1, 1, -1, 0, 0, 1} qui correspondent aux graphiques de la figure 7.17.

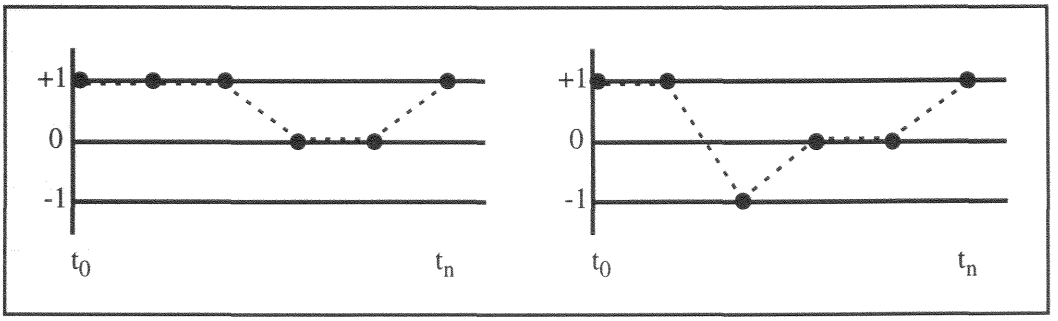


Figure 7.17 : Représentation graphique de la quatrième série temporelle.

Les séries temporelles simples ne sont pas les seules que nous avons mises en œuvre pendant cette étude des capacités de mémorisation abstraite du modèle gamma. Nous avons également étudié un type de séquences plus complexes puisque les informations nécessaires à une bonne classification sont dispersées dans toute la série et non pas uniquement présentes dans un seul élément comme pour les quatre séries que nous venons de voir. Ce problème de classification sera présenté au paragraphe 7.6.3 tandis que les résultats seront présentés au paragraphe 7.6.3.4.

7.6.2/ Résultats obtenus sur les séries temporelles

7.6.2.1/ Présentation

Nous allons exposer dans ce paragraphe la mise en forme adoptée pour la présentation des résultats et les conditions d'apprentissage qui ont permis de les obtenir.

Les tableaux des paragraphes suivants (du paragraphe 7.6.2.2 au paragraphe 7.6.2.5) sont tous présentés de manière identique pour faciliter la compréhension des résultats. Chaque tableau est constitué de 4 colonnes de 14 lignes. Les colonnes sont regroupées par paires, chaque paire correspondant à un type de contrainte de partage des poids dans la plaque d'entrée. La plaque d'entrée est, en l'occurrence, une unique ligne de délais du fait du problème posé. Cette ligne de

délais peut, tout d'abord, être organisée en canal ce qui signifie que toutes les unités de la ligne partageront un coefficient de régression de même valeur, le terme de canal caractérisant pour nous la mise en place d'une procédure de partage des poids. La ligne de délais peut également être organisée en ligne, ce qui signifie que chaque unité de la ligne possède un coefficient de régression dont la valeur lui est propre. La ligne de délais de la plaque d'entrée est, par ailleurs, constituée d'une ou de deux unités.

Dans le cas où la ligne ne comporte qu'une seule unité gamma, le cas où les poids sont indépendants et le cas où les poids sont partagés sont évidemment équivalents. Nous donnons cependant ces deux cas puisque l'apprentissage se fait à partir d'un ensemble de poids initialisés aléatoirement, cette initialisation pouvant conduire à des résultats différents. Les tables présentent, dans l'ordre des colonnes, le cas d'une ligne de délais canal avec une, puis deux unités gamma partageant le même coefficient de régression. Les colonnes suivantes présentent le cas d'une ligne de délais à coefficients de régression libres avec une, puis deux unités gamma.

La couche cachée est, elle, organisée selon deux architectures différentes et selon sept tailles différentes. L'architecture utilisée en couche cachée peut être une architecture de type perceptron multicouche, ce qui signifie qu'aucune unité gamma n'est implantée en sortie du neurone. Une autre possibilité consiste à utiliser un neurone gamma, neurone répondant à la définition de [mcculloch43] et auquel a été ajouté, en sortie, une unité gamma. Nous avons, dans ces paragraphes, uniquement utilisé le cas où les neurones gamma possèdent tous des coefficients de régression qui leur sont propres. Le problème posé dans le cas où la couche cachée est constituée de neurones gamma possédant tous le même coefficient de régression sera présenté plus avant (cf. paragraphe 7.6.2.6). Dans les deux cas d'architecture, nous avons utilisé une couche cachée de deux à huit neurones, que ceux-ci soient des neurones gamma ou non.

Les différentes possibilités architecturales mises en place pour la reconnaissance des séries temporelles sont résumées graphiquement dans la figure 7.18 suivante. Nous attirons l'attention du lecteur sur l'architecture minimale mise en œuvre lors des tests sur les séries temporelles. Le réseau minimal, possédant les plus faibles capacités de mémorisation, est constitué d'une seule unité gamma en couche d'entrée, de 2 neurones standards en couche cachée et de 2 neurones standards en couche de sortie.

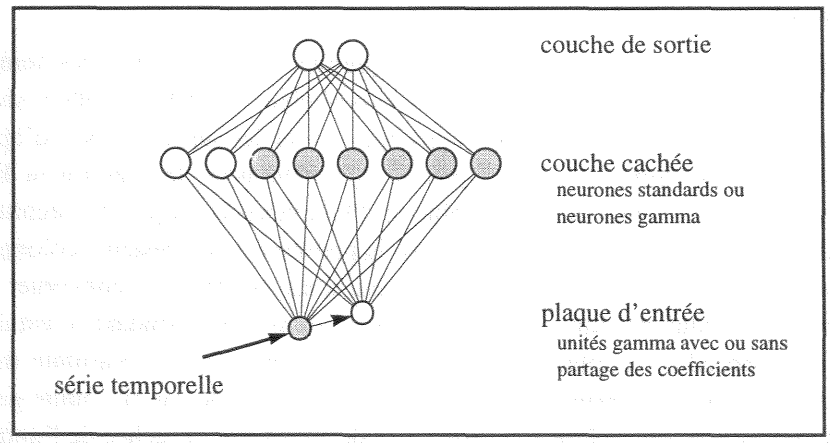


Figure 7.18 : Architecture mise en œuvre pour la reconnaissance de séries temporelles.

Les pourcentages donnés dans les différentes tables ne reposent pas tous sur un corpus de taille identique. Ainsi, lorsque le signal est propre et correspond aux séries temporelles présentées tout au long du paragraphe 7.6.1, seules 2 séries sont utilisées, chaque classe étant représentée de manière unique. Le corpus d'apprentissage est, dans ce cas, équivalent au corpus de test. Par contre, lorsque la série est bruitée par un bruit uniforme, le corpus d'apprentissage comprend 100 séries de chaque classe, le corpus d'apprentissage comprenant alors un total de 200 séries temporelles. Le corpus de

test utilisé dans le cas d'un corpus d'apprentissage bruité est, lui, constitué des 2 seules séries non bruitées, ce corpus de test étant ainsi équivalent au corpus de test utilisé dans le cas où les séries ne sont pas bruitées. Ce dernier choix a été fait de manière à nous permettre de juger de la qualité de l'apprentissage par rapport aux séries temporelles originales lorsque l'apprentissage est effectué en milieu bruité. Ce choix peut paraître quelque peu étrange mais l'exposé des résultats permettra au lecteur de mieux le comprendre.

Les choix effectués lors de la mise en place de la procédure d'apprentissage sont les suivant :

- utilisation de la rétropropagation dans le temps qui permet, par dépliage du réseau, d'obtenir les valeurs exactes des gradients d'erreur,
- le coefficient d'apprentissage a été fixé à 0,01 et l'ajustement des poids synaptiques se fait donc très lentement. Rappelons que la valeur habituelle de ce coefficient est de 0,9,
- le momentum a été fixé à 0,9 ce qui signifie que la phase de rétropropagation précédente pondère énormément la phase courante. La valeur standard de ce coefficient est de 0,3,
- la valeur du delta minimal a été fixée à 0. Cette heuristique n'est donc pas utilisée,
- la valeur d'élimination du méplat a été fixée à 0. Là encore, cette heuristique n'est pas utilisée,
- le *boot* de convergence, correspondant au nombre de présentations du corpus au réseau pour l'adaptation des seuls poids synaptiques, a été mis à 0 : l'heuristique n'est pas utilisée,
- le nombre de calculs de la valeur du delta avant mise à jour des poids est fixé à 1 : cette heuristique n'est donc pas utilisée mais n'avait pas lieu d'être avec un apprentissage effectué selon la méthode de la BPTT,
- utilisation d'un atténuateur de couple de 0,001. Cet atténuateur permettant d'amoindrir la mise à jour des coefficients de régression par rapport aux poids connexionnistes. Les poids synaptiques étant modifiés selon un coefficient d'apprentissage de 0,01, le coefficient d'apprentissage pour les coefficients de régression est donc de $0,01 \times 0,001 = 1.10^{-5}$.
- utilisation d'un espace très contraint pour les initialisations des poids synaptiques et des coefficients de feedback. Les poids connexionnistes sont initialisés avec une moyenne de 0 et un écart-type maximal de 0,1 tandis que les coefficients de feedback sont initialisés avec une moyenne de 0,5 et un écart-type maximal de 0,001.

7.6.2.2/ Premier type de séquences temporelles

Les résultats concernant la classification du premier type de séquences temporelles sont très rassurants... Dans le cas de séries temporelles non bruitées, dont les résultats sont donnés dans la table 7.1, seul un apprentissage n'a pas convergé, toutes les autres phases d'apprentissage ayant parfaitement réussi. Comme cela a été précisé précédemment, les cas où la ligne de délais ne comporte qu'une seule unité permet d'obtenir l'équivalence entre des architectures ayant des politiques de partage des poids différents. Ainsi, la première et la troisième colonne de résultats sont sémantiquement équivalente. Le fait que le réseau n'ait pas réussi à converger dans le cas où la plaque d'entrée possédait des (un...) coefficients de régression indépendants n'est donc pas à mettre à la charge de la procédure d'apprentissage dans son relâchement de contrainte mais plutôt dans sa qualité intrinsèque moyenne puisque le réseau a convergé dans le cas où l'unité gamma de la plaque d'entrée partageait avec elle-même son coefficient de régression ! L'échec de l'apprentissage est plus sûrement à mettre au compte d'une très mauvaise initialisation aléatoire des poids plutôt qu'au compte de l'excès de liberté de l'unique paramètre.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	100 / 100	100 / 100	100 / 100	100 / 100
	3	100 / 100	100 / 100	100 / 100	100 / 100
	4	100 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	100 / 100	100 / 100	100 / 100	100 / 100
	7	100 / 100	100 / 100	100 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100
Perceptron	2	100 / 100	100 / 100	50 / 50	100 / 100
	3	100 / 100	100 / 100	100 / 100	100 / 100
	4	100 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	100 / 100	100 / 100	100 / 100	100 / 100
	7	100 / 100	100 / 100	100 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100

Table 7.1 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le premier type de séquences temporelles non bruitées.

Le but ultime de cette partie de notre thèse étant d'obtenir une segmentation efficace de la parole en milieu bruité, il nous a semblé intéressant de tenter d'effectuer des apprentissages de reconnaissances de séquences temporelles en milieu bruité. Ces tentatives nous semblent être une bonne voie d'approche pour les phénomènes bruités même si le signal de parole est autrement plus complexe que les séquences analysées ici et même si le mode de bruitage que nous avons employé ne permet pas d'obtenir la complexité des bruits réels.

Les résultats obtenus sont cependant intéressants puisqu'ils permettent de constater la relativement bonne robustesse du réseau lors de l'apprentissage de séquences temporelles bruitées uniformément à 5 décibels. Quelques problèmes apparaissent cependant qui permettent de juger de la qualité de certains choix architecturaux lors du mariage d'un type d'architecture d'entrée et d'un type d'architecture de couche cachée.

Il est ainsi très aisé de constater, dans la table 7.2, que l'usage d'un perceptron en couche cachée ne permet pas d'obtenir des résultats d'aussi bonne qualité que ceux obtenus avec une couche cachée mettant en œuvre des neurones gamma. Cette constatation est une première justification de l'emploi de neurones à décroissance exponentielle de l'activité en couche cachée. Les résultats concernant les perceptrons permettent également de constater la présence de certains résultats assez peu compréhensibles. Il est ainsi aisé de comprendre que 50% des séquences de test sont bien reconnues lorsque seuls 50% des séquences d'apprentissage sont bien classées, il est par contre moins facile de comprendre que seule une des deux séquences de test est correctement reconnue alors que 98% des séquences d'apprentissage sont correctement classées. Cette différence s'explique principalement par le faible rapport signal-sur-bruit qui déforme les séries initiales de manière très importante et par notre choix de ne prendre dans le corpus de test des séries bruitées les deux seules séquences non bruitées, ce choix ne permettant pas d'effectuer l'évaluation en moyenne de l'apprentissage d'un phénomène déformé par un bruit aléatoire.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	91 / 100	93 / 100	91 / 100	100 / 100
	3	95 / 100	92 / 100	92 / 100	87 / 100
	4	93 / 100	100 / 100	96 / 100	100 / 100
	5	94 / 100	100 / 100	96 / 100	100 / 100
	6	95 / 100	100 / 100	95 / 100	100 / 100
	7	96 / 100	100 / 100	96 / 100	100 / 100
	8	97 / 100	100 / 100	97 / 100	100 / 100
Perceptron	2	93 / 100	94 / 100	97 / 100	93 / 100
	3	95 / 100	97 / 50	93 / 100	91 / 100
	4	50 / 50	99 / 100	50 / 50	99 / 100
	5	96 / 100	98 / 50	50 / 50	99 / 100
	6	96 / 100	98 / 50	96 / 100	95 / 100
	7	96 / 100	99 / 100	96 / 100	98 / 50
	8	50 / 50	99 / 100	97 / 100	99 / 100

Table 7.2 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le premier type de séquences temporelles bruitées à 5 décibels de rapport signal-sur-bruit par un bruit uniforme.

Cette première série étant la plus simple de toutes celles que nous avons étudié, nous allons maintenant aller plus avant dans la complexité des besoins en mémoire pour vérifier la supériorité du neurone gamma par rapport au neurone standard.

7.6.2.3/ Deuxième type de séquences temporelles

L'étude des résultats d'apprentissage et de reconnaissance du deuxième type de séquences temporelles permet de constater l'apparition de certains problèmes. Alors que les séries du premier type n'ont permis de constater que d'assez légères différences entre architectures à neurones standards et architectures à neurones gamma, le deuxième type de séries temporelles va nous permettre de constater le besoin de possibilités de mémorisation qui ne peuvent être entièrement fournies par une architecture comme le perceptron qui n'est capable que de discrimination.

Lors de l'étude des résultats d'apprentissage sur le premier type de séries temporelles, nous avons pu constater qu'un réseau ne possédant qu'une unité gamma en entrée et ne possédant par ailleurs que deux neurones standards en couche cachée pouvait parfaitement apprendre à discriminer les deux séquences temporelles. Les résultats fournis à la table 7.3 prouvent que le deuxième type de séquences temporelles est structurellement plus difficile à apprendre et que les besoins en unités aptes à la mémorisation se fait sentir de manière plus aiguë. Ainsi, les tentatives d'apprentissage des séries avec une seule unité gamma en couche d'entrée et des neurones standards en couche cachée ne permettent pas d'obtenir un réseau apte à résoudre la tâche. Seule la mise en place de deux unités gamma successives dans la plaque d'entrée, alors que la couche cachée reste constituée de neurones standards, permet au réseau de converger sans que cette convergence soit, pour autant, assurée.

Il est possible de conclure, à partir de cette étude sur des séries non bruitées, que le deuxième type de séries nécessite au moins deux unités gamma en couche d'entrée pour que les séries puissent être reconnues par un réseau ne possédant pas d'autres moyens de mémorisation. Cette extension de la ligne de délais à plus d'une unité équivaut, en quelque sorte, à une modification du type de celles qui prévalent à la définition des réseaux NARX (cf. chapitre 6, paragraphe 6.3.9) puisque la reconnaissance de séries ne devrait poser aucun problème avec une ligne de délais de quatre délais

simples (cf. figure 7.15). L'utilisation d'un seul délai dans la plaque d'entrée oblige à utiliser ici des neurones gamma en couche cachée pour obtenir un résultat parfaitement satisfaisant.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	100 / 100	100 / 100	100 / 100	100 / 100
	3	100 / 100	100 / 100	100 / 100	100 / 100
	4	100 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	100 / 100	100 / 100	100 / 100	100 / 100
	7	100 / 100	100 / 100	100 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100
Perceptron	2	50 / 50	100 / 100	50 / 50	100 / 100
	3	50 / 50	50 / 50	50 / 50	50 / 50
	4	50 / 50	100 / 100	50 / 50	50 / 50
	5	50 / 50	100 / 100	50 / 50	100 / 100
	6	50 / 50	100 / 100	50 / 50	100 / 100
	7	50 / 50	100 / 100	50 / 50	100 / 100
	8	50 / 50	100 / 100	50 / 50	100 / 100

Table 7.3 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le deuxième type de séquences temporelles non bruitées.

La reconnaissance des séries temporelles du deuxième type en milieu bruité avec un bruit uniforme à 5 décibels nous poussent à faire des constatations similaires à celles qui viennent d'être faites. La table 7.4 montre en effet des résultats équivalents à ceux obtenus avec des séries non bruitées (cf. table 7.3), la faiblesse des architectures n'employant qu'une unité gamma en couche d'entrée et utilisant des neurones standards en couche cachée étant, ici encore, parfaitement vérifiée.

Il est également possible de remarquer la faiblesse relative des réseaux constitués de neurones standards en couche cachée et de 2 unités gamma en couche d'entrée possédant des coefficients de régression libres. Cette architecture permettait de parfaitement reconnaître la série temporelle précédente, même à un rapport signal-sur-bruit de 5 décibels. Les résultats sont désormais beaucoup moins bons puisqu'il est même possible d'observer assez fréquemment de bons apprentissages corrélés avec une mauvaise reconnaissance des séries non bruitées. Il est cependant évident, à la vue des résultats, que le partage des poids dans la couche d'entrée a été bénéfique puisque les résultats obtenus dans ce cas sont de meilleure qualité que ceux obtenus avec l'architecture où la contrainte d'égalité avait été relâchée.

Malgré tout, les réseaux constitués d'unités gamma en couche d'entrée et de neurones gamma en couche cachée sont parfaitement capable d'apprendre à discriminer les séquences bruitées fournies en apprentissage, prouvant encore une fois l'intérêt pour des systèmes possédant une capacité intrinsèque de mémorisation à court et moyen terme.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	100 / 100	100 / 100	100 / 100	100 / 100
	3	100 / 100	100 / 100	100 / 100	100 / 100
	4	100 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	100 / 100	100 / 100	100 / 100	100 / 100
	7	100 / 100	100 / 100	100 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100
Perceptron	2	50 / 50	99 / 100	50 / 50	99 / 100
	3	48 / 50	99 / 100	48 / 50	50 / 50
	4	50 / 50	99 / 100	50 / 50	92 / 50
	5	53 / 50	47 / 50	50 / 50	86 / 50
	6	48 / 50	99 / 100	48 / 50	50 / 50
	7	50 / 50	99 / 100	47 / 50	47 / 50
	8	47 / 50	50 / 50	48 / 50	50 / 50

Table 7.4 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le deuxième type de séquences temporelles bruitées à 5 décibels de rapport signal-sur-bruit par un bruit uniforme.

Après l'étude des résultats obtenus sur les séries de deuxième type, nous allons maintenant étudier les résultats obtenus sur le troisième type de séries temporelles. Celles-ci semblent, de prime abord, assez proches des séries du deuxième type mais les résultats obtenus diffèrent cependant de manière assez importante.

7.6.2.4/ Troisième type de séquences temporelles

Le troisième type de séries temporelles n'est différent du deuxième type de séries que par un élément dans chaque série. Nous avons en effet défini le troisième type à partir du deuxième en modifiant l'avant dernier élément, que nous avons fait passer de +1 à 0. Les deux types de séries semblent donc relativement proches, l'élément discriminant étant toujours le troisième élément de chaque série qui comporte un total de 6 éléments successifs.

Les résultats, malgré l'apparente similitude des séries, sont assez différents. La première table, la table 7.5, qui porte sur la reconnaissance des séries temporelles non bruitées, est assez semblable à celle obtenue sur le deuxième type de séries temporelles (table 7.3), les résultats étant même de meilleure qualité dans le cas où le réseau est constitué d'une couche d'entrée de deux unités gamma à coefficients de régression libres et d'une couche cachée composée de neurones standards, à la manière d'un perceptron.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	100 / 100	100 / 100	100 / 100	100 / 100
	3	99 / 100	100 / 100	100 / 100	100 / 50
	4	99 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	99 / 100	100 / 100	99 / 100	100 / 100
	7	99 / 100	100 / 100	99 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100
Perceptron	2	50 / 50	98 / 50	50 / 50	86 / 50
	3	48 / 50	98 / 50	50 / 50	85 / 50
	4	50 / 50	98 / 50	49 / 50	87 / 50
	5	50 / 50	46 / 50	49 / 50	81 / 50
	6	49 / 50	96 / 50	50 / 50	55 / 50
	7	51 / 50	90 / 50	50 / 50	85 / 50
	8	50 / 50	53 / 50	50 / 50	55 / 50

Table 7.6 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le troisième type de séquences temporelles bruitées à 5 décibels de rapport signal-sur-bruit par un bruit uniforme.

Les résultats obtenus sur les séries bruitées du troisième type nous confortent encore plus dans notre certitude de la qualité des réseaux utilisant des mécanismes tels que ceux mis en œuvre par le filtre gamma. La qualité des résultats obtenus jusqu'ici ne doivent cependant pas faire penser que l'architecture gamma est la panacée à tous les problèmes qu'il est possible de rencontrer. Outre le fait que ce type de mécanisme soit, d'emblée, inadapté aux tâches que nous sommes en train d'étudier lorsque celles-ci sont constituées de nombreux éléments, quelques études supplémentaires peuvent amoindrir la confiance que nous avions développée jusqu'ici dans les capacités des neurones gamma, comme nous allons le voir maintenant.

7.6.2.5/ Quatrième type de séquences temporelles

Le quatrième type de séries temporelles a été élaboré à partir du troisième type. Les séries du deuxième, troisième et quatrième type sont donc toutes élaborées selon une base commune et varie, d'une expérience à l'autre, par la redéfinition d'un seul élément. La différence entre le deuxième et le quatrième type de séries temporelles ne se fait donc que sur deux éléments de chaque série. En comparant les graphiques de la figure 7.15 et de la figure 7.17, il est possible de constater que les deux séries ne varient que par leurs deuxième et cinquième éléments qui passent de -1 et +1 à, respectivement, +1 et 0.

Le quatrième type de séries est, comme le montre la figure 7.17, beaucoup plus présente dans la partie positive du spectre des valeurs possibles que ne l'était le deuxième type, la première série du quatrième type (partie gauche de la figure 7.17) étant d'ailleurs strictement positive ou nulle.

Les résultats obtenus avec ce type de séries temporelles dans les expérimentations non bruitées sont très satisfaisants. En effet, comme le montre la table 7.7, toutes les architectures que nous avons essayées sur ce problème ont parfaitement convergé. L'apprentissage a été parfait dans les cas où les capacités de mémorisation étaient maximales (partie haute du tableau) tout autant que dans les cas où la mémorisation était faible (partie basse du tableau). Cette remarque est en particulier vraie pour les architectures ne possédant qu'une unité gamma en couche d'entrée et ne possédant par ailleurs que deux neurones standards en couche cachée. Ce succès va, d'une certaine manière, à l'encontre des

constatations que nous avons faites à partir des études présentées dans les deux paragraphes précédents. La définition de la première classe de ce type de séries qui se caractérise par la présence d'éléments strictement positifs ou nuls par rapport aux séries de la deuxième classe, où un élément négatif provoque un déchargement brutal de l'activité accumulée dans les unités gamma et, par conséquent, dans les neurones gamma, pourraient expliquer la très bonne qualité de ces premiers résultats (cf. figure 7.17). Malheureusement, ces résultats ne se tiennent pas en présence de bruit comme nous allons le voir.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	100 / 100	100 / 100	100 / 100	100 / 100
	3	100 / 100	100 / 100	100 / 100	100 / 100
	4	100 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	100 / 100	100 / 100	100 / 100	100 / 100
	7	100 / 100	100 / 100	100 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100
Perceptron	2	100 / 100	100 / 100	100 / 100	100 / 100
	3	100 / 100	100 / 100	100 / 100	100 / 100
	4	100 / 100	100 / 100	100 / 100	100 / 100
	5	100 / 100	100 / 100	100 / 100	100 / 100
	6	100 / 100	100 / 100	100 / 100	100 / 100
	7	100 / 100	100 / 100	100 / 100	100 / 100
	8	100 / 100	100 / 100	100 / 100	100 / 100

Table 7.7 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le quatrième type de séquences temporelles non bruitées.

Comme pour tous les types de séries précédents, nous avons effectué des tests d'apprentissage sur des séries temporelles bruitées. Ces apprentissages se révèlent être de piètre qualité dans presque tous les cas comme le montre la table 7.8 ci-dessous.

Cette table de résultats montre en effet que, bien que les séries temporelles utilisées en apprentissage soient presque toutes reconnues, permettant d'obtenir des pourcentages de reconnaissance, à une exception près, de l'ordre de 97 à 99%, les résultats de reconnaissance obtenus sur les séquences de validation, qui ne sont pas bruitées, sont tous, à trois exceptions près, égaux à 50%. Ainsi, bien que les séquences bruitées puissent être correctement reconnues, les séquences de test non bruitée, qui ont servi à définir le corpus d'apprentissage, ne peuvent pas l'être. Le niveau de bruit a été assez fort pour permettre aux différents réseaux utilisés de partitionner l'espace des poids de manière à ce que le corpus d'apprentissage puisse être appris sans que le corpus de test puisse être correctement classé. Ce résultat est assez inattendu et les poids obtenus pendant ces expérimentations seraient intéressants à analyser par une méthode d'extraction de règles, si une telle méthode pouvait exister pour les réseaux d'unités gamma.

		Canal gamma		Unités gamma	
		1	2	1	2
Unités gamma	2	99 / 50	99 / 50	99 / 50	99 / 100
	3	99 / 50	99 / 50	99 / 50	99 / 50
	4	99 / 50	99 / 50	99 / 50	99 / 50
	5	99 / 50	99 / 50	99 / 50	99 / 100
	6	99 / 50	99 / 100	99 / 50	99 / 50
	7	99 / 50	99 / 50	99 / 50	99 / 50
	8	99 / 50	99 / 50	99 / 50	99 / 50
Perceptron	2	95 / 50	99 / 50	96 / 50	99 / 50
	3	97 / 50	99 / 50	97 / 50	99 / 50
	4	97 / 50	99 / 50	97 / 50	99 / 50
	5	97 / 50	48 / 50	98 / 50	99 / 50
	6	97 / 50	99 / 50	98 / 50	99 / 50
	7	98 / 50	99 / 50	50 / 50	99 / 50
	8	98 / 50	99 / 50	97 / 50	99 / 50

Table 7.8 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) sur le quatrième type de séquences temporelles bruitées à 5 décibels de rapport signal-sur-bruit par un bruit uniforme.

Les résultats assez particuliers que nous venons de voir à la table 7.8 peuvent être agrémentés d'autres résultats montrant l'intérêt d'une technique d'apprentissage par rapport à une autre.

7.6.2.6/ Problèmes observés à l'apprentissage

Nous avons décidé, pour cet exposé des différentes heuristiques d'apprentissage développées, d'utiliser le troisième type de séquences temporelles. Il est en effet possible de conclure, après l'étude des 4 différents types de séquences que nous venons de présenter, que le troisième type de séquences est celui pour lequel le besoin en mémoire est le plus évident puisque l'utilisation d'une couche cachée implantant des neurones standards ne permet pas d'obtenir de bon résultats lorsque la couche d'entrée n'est constituée que d'une seule unité gamma.

Ce paragraphe va nous permettre de présenter les résultats obtenus lorsque nous faisons varier les conditions d'apprentissage. Les conditions de base sont les mêmes que celles que nous avons employées pour présenter les résultats précédents et nous ne ferons varier qu'une seule condition à la fois, pour faciliter la compréhension de l'incidence de chacune des heuristiques. Les résultats présentés ont été obtenus avec un réseau possédant une unité gamma en couche d'entrée, deux neurones gamma en couche cachée et deux neurones standard en couche de sortie.

La première heuristique que nous présentons concerne la condition de partage des coefficients de régression en couche cachée. Nous avons, tout au long des paragraphes précédents, utilisé la condition la moins contraignante pour le réseau puisque nous avons laissé les coefficients se définir isolément. Cette condition est donc celle qui génère le plus grand nombre de degrés de liberté. Il est possible de contraindre cette liberté en obligeant les neurones gamma de la couche cachée à partager un seul et même coefficient de feedback.

Nous demandons au lecteur de bien vouloir croire en nos affirmations vis-à-vis des résultats que nous présentons dans ce paragraphe puisque le manque de couleurs des figures qui suivent porte grandement atteinte à la compréhension des résultats...

La figure 7.19 présente les résultats de convergence de deux réseaux dont l'un était contraint à n'avoir qu'une valeur de coefficient dans la couche cachée. Les résultats d'apprentissage sont

quasiment équivalents bien que le réseau possédant trois coefficients de régression, c'est à dire le réseau qui n'était pas contraint, ait été le premier à reconnaître 100% du corpus puis le premier à se stabiliser à 100%.

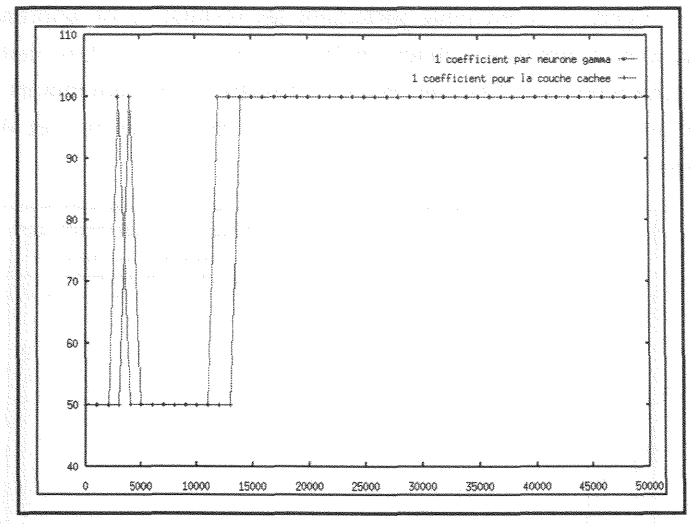


Figure 7.19 : Rapidité de convergence de l'apprentissage selon la condition de partage des coefficients des neurones gamma de la couche cachée.

La figure 7.20 présente l'évolution des deux ou trois coefficients de régression du réseau et peuvent être comparés aux deux courbes d'apprentissage de la figure 7.19. Comme le montre les deux graphiques, les évolutions les plus fortes ont eu lieu dans le réseau le moins contraint. La valeur du coefficient de régression de la couche cachée du réseau contraint n'a, par ailleurs, pratiquement pas variée.

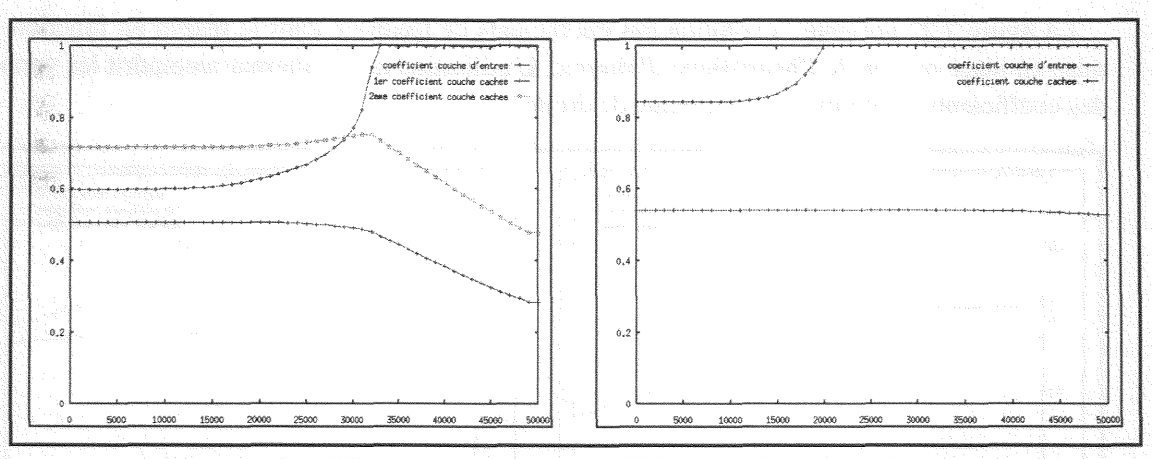


Figure 7.20 : Les différentes définitions de coefficients de régression selon que les neurones de la couche cachée partagent (droite) ou non (gauche) le même coefficient.

Les résultats présentés sont meilleurs que d'autres qui nous ont poussés à abandonner la contrainte d'égalité des coefficients de feedback en couche cachée. Plus précisément, lorsque cette contrainte est mise en œuvre avec un réseau dans une tâche de segmentation de la parole, le coefficient de la couche cachée se positionne presque toujours aux alentours immédiats de 1, empêchant alors la couche cachée d'implanter de la mémoire. Ce comportement est dû au fait que, lorsque la contrainte n'est pas mise en œuvre, un grand nombre de neurones gamma de la couche cachée positionnent leurs coefficients à 1, laissant un petit nombre de neurones implanter de la mémoire. Nous reviendrons sur cette constatation dans le paragraphe 7.7.2 traitant la segmentation de la parole.

Une autre heuristique possible est la mise en place d'un apprentissage par alternat. Cette heuristique permet d'alterner (...) les phases de modification des poids synaptiques et des coefficients

de régression, permettant ainsi à chacun des ensembles de connexions de s'ajuster par rapport à des conditions d'intégration de l'information ou de mémorisation fixes. Il est possible de définir un nombre variable de présentations de corpus pour chaque tour d'alternat, nombre qui peut être au moins égal à 1. La figure 7.21 présente justement les résultats d'un apprentissage sans alternat et d'un apprentissage avec alternance à chaque présentation de corpus. Les résultats permettent de tirer des conclusions similaires à celles que nous avons faites précédemment : l'apprentissage sans alternat permet d'atteindre plus rapidement 100% de reconnaissance et de se fixer, de manière définitive, plus rapidement aux 100%.

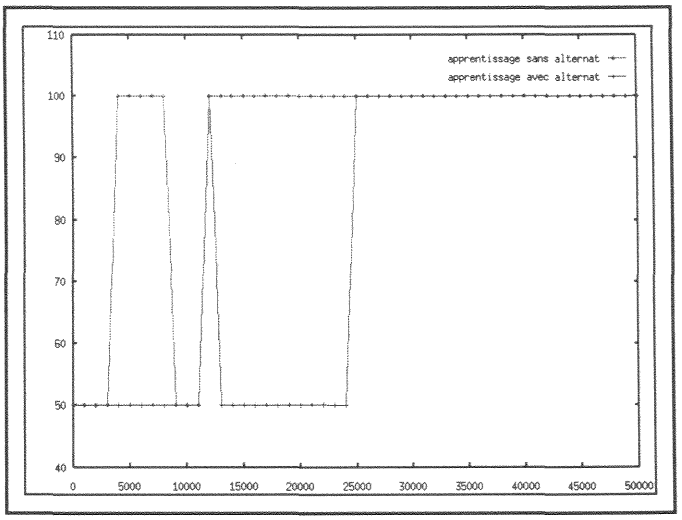


Figure 7.21 : Convergence de l'apprentissage pour des réseaux dont les poids et les coefficients sont modifiés, ou non, en alternat.

La figure 7.22 présente l'évolution des coefficients de feedback dans le réseau en fonction de la mise en place ou non de l'heuristique d'alternat. L'utilisation de cet alternat amoindrit les variations des coefficients de régression (graphique de droite).

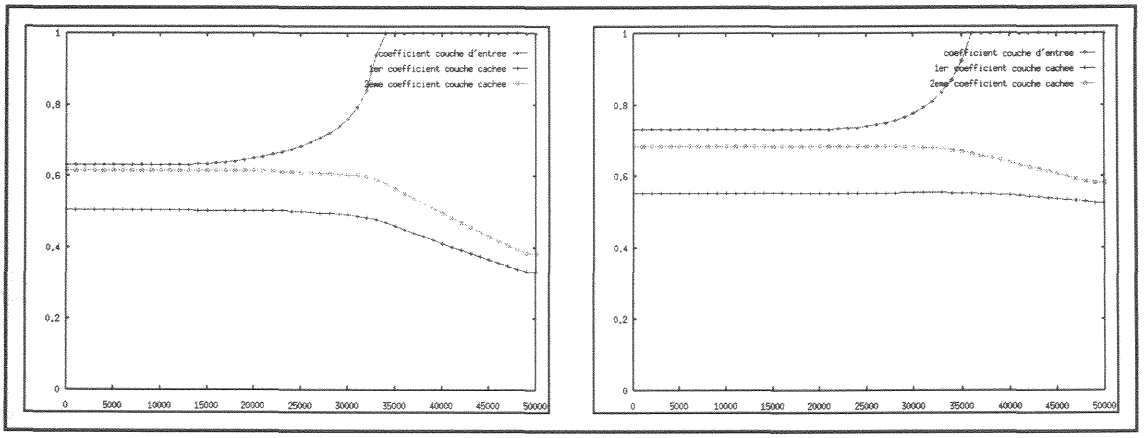


Figure 7.22 : Les définitions des coefficients de régression lors d'apprentissages sans (gauche) et avec (droite) alternat.

Une autre heuristique, reposant sur le même concept de dissociation des modifications des poids synaptiques et des coefficients de régression, peut également être envisagée par la mise en place d'une phase de *boot* de convergence. Cette phase se caractérise par la seule modification des poids synaptiques. Ceux-ci s'adaptent donc aux conditions de mémorisation définies par la phase d'initialisation aléatoire. Une fois cette première phase terminée, les poids et les coefficients de régression sont modifiés de concert. Comme le montre la figure 7.23, la mise en place de la phase de *boot* permet au réseau considéré d'atteindre le premier 100% de reconnaissance bien que

l'apprentissage soit instable et perde ensuite en qualité. Le réseau n'implantant pas l'heuristique de *boot* est le premier à atteindre 100% de manière définitive.

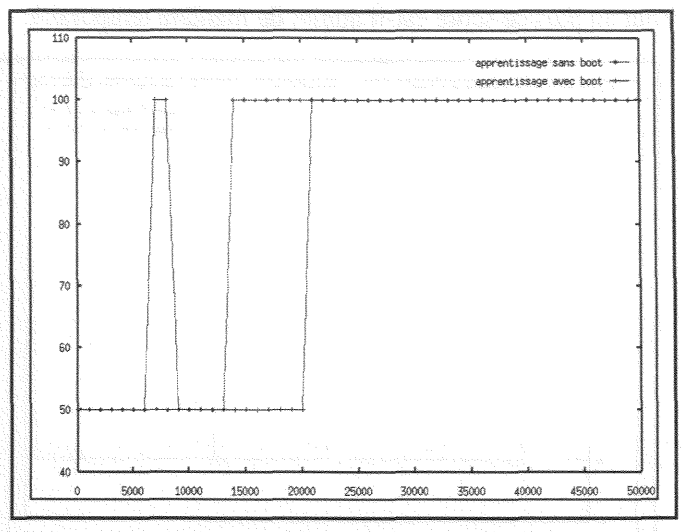


Figure 7.23 : Convergence de l'apprentissage pour des réseaux dont les poids et les coefficients de régression sont modifiés après un boot ou pas.

La figure 7.24 présente l'évolution des coefficients de régression lors des deux apprentissages effectués avec ou sans *boot*. Le graphique de droite présente l'évolution des coefficients de régression lors d'un apprentissage lorsque l'heuristique est utilisée. Ces coefficients sont fixes pendant 10.000 pas de temps. Nous avons en effet choisi d'effectué une phase de *boot* sur 5.000 présentations du corpus d'apprentissage soit 10.000 présentations d'une séquence. Les modifications des coefficients sont, dans l'ensemble, beaucoup plus forte sans *boot* qu'avec. Cela s'explique par le fait que la phase de *boot* a réduit le partie la plus importante de l'erreur, les gradients rétropropagés après la première phase étant assez faibles.

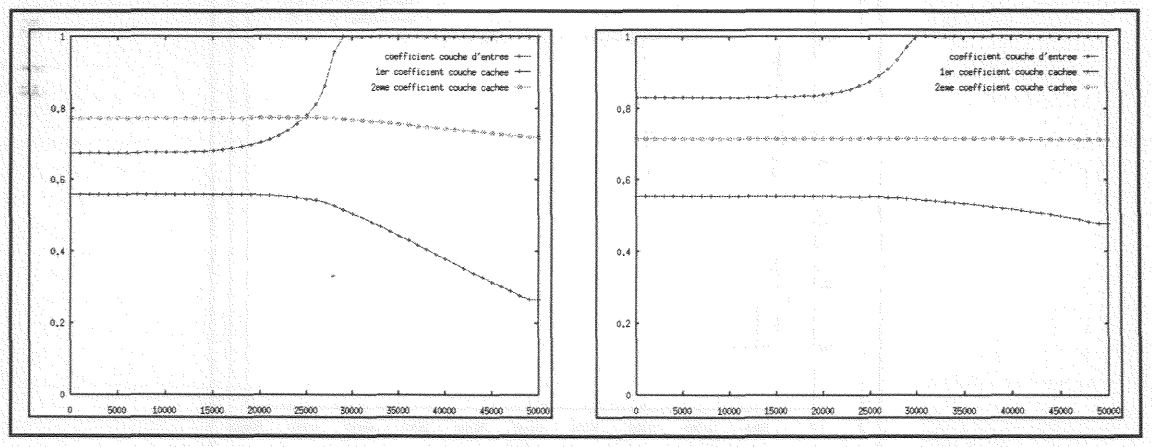


Figure 7.24 : Les définitions des coefficients de régression lors d'apprentissages sans (gauche) et avec (droite) boot de convergence.

La dernière heuristique que nous allons présenter est la plus importante. Il s'agit de l'heuristique relative à l'atténuateur de couple auquel nous avons déjà consacré tout le paragraphe 7.2.5.6. Les résultats que nous allons présenter maintenant sont suffisament explicites pour justifier pleinement cette heuristique et montrer tout l'intérêt qu'il peut y avoir à la mettre en œuvre.

Les résultats d'apprentissages que nous avons présentés jusqu'à présent ont été obtenus avec un atténuateur de couple de 0,001. Nous avons donc décidé de présenter des résultats d'apprentissage pour des atténuateurs allant de 1 à 0,001 par pas de puissance de 10 pour que le lecteur puisse juger par lui même des modifications progressives que la réduction apporte. La figure 7.25 présente les

résultats d'apprentissage obtenus avec deux réseaux pour lesquels les atténuateurs de couple ont été fixés à 1 et à 0,1. Comme le montre cette figure, aucun des réseaux ne réussit à converger de manière stable vers 100%. Tout au plus ce score est-il atteint de manière temporaire.

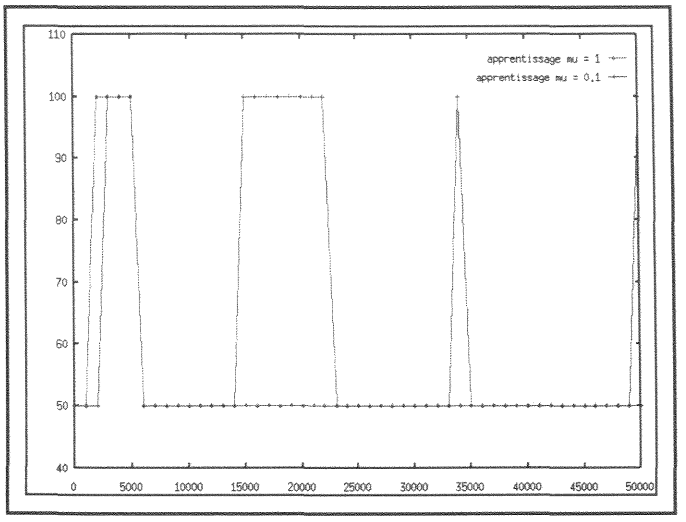


Figure 7.25 : Convergence de l'apprentissage pour des réseaux gamma dont le l'atténuateur de couple est à 1 et à 0,1.

La deuxième figure présentant des résultats d'apprentissage est beaucoup plus intéressante. Elle montre les convergences de réseaux pour des atténuateurs de couple de 0,01 et 0,001. Les deux réseaux présentent un premier pic à 100% puis une convergence définitive et assez rapide à 100%. Le réseau atteignant le premier 100% et restant stable à 100% le premier est celui pour lequel l'atténuateur de couple a été fixé à 0,001.

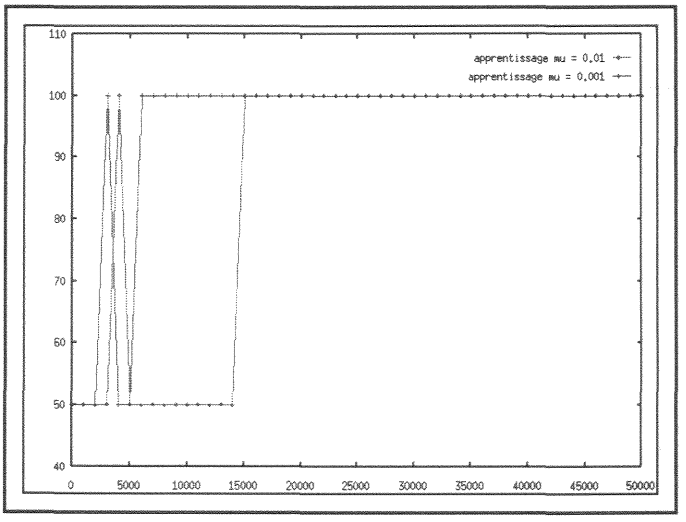


Figure 7.26 : Convergence de l'apprentissage pour des réseaux gamma dont le l'atténuateur de couple est à 0,01 et à 0,001.

L'avantage de la mise en place d'un atténuateur de couple peut être facilement compris en observant les deux figures qui suivent. La première, la figure 7.27, présente les évolutions des coefficients de régression pour des atténuateurs de couple de 1 et de 0,1. Cette première figure permet de constater l'évolution brutale et presqu'anarchique suivie par les coefficients. Cette évolution brutale est principalement due, comme nous l'avons déjà dit précédemment, au déphasage qui existe entre la définition des poids synaptiques et la définition des coefficients de régression. La modification de ces deux ensembles de poids, en même temps et au même rythme, provoque une inadéquation entre les conditions de mémorisation des traces du signal et l'exploitation de ces traces

par l'intermédiaire des poids synaptiques, cette inadéquation étant responsable d'une erreur assez forte pour permettre au processus de s'auto-générer.

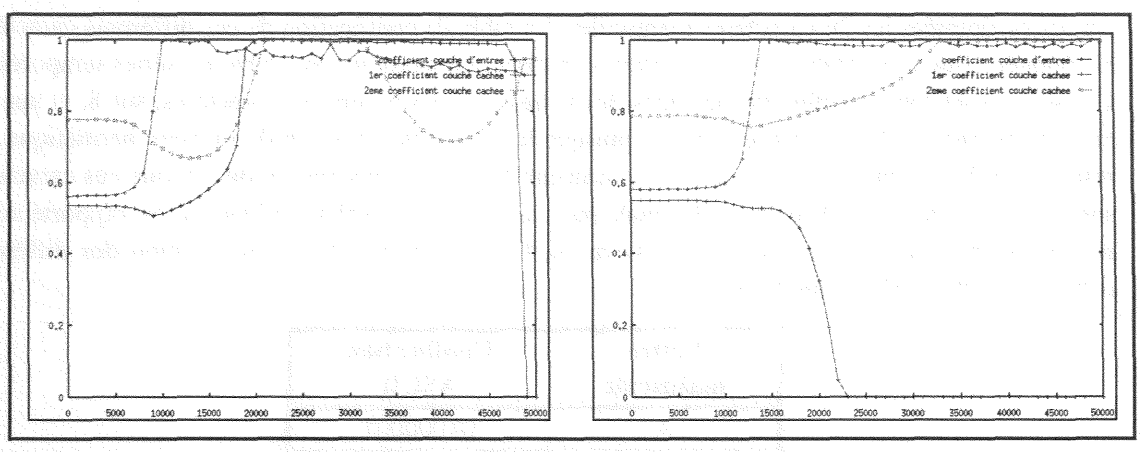


Figure 7.27 : Les définitions des coefficients de régression lors d'apprentissages avec un atténuateur de couple à 1 (gauche) et à 0,1 (droite).

La figure 7.28 présente l'évolution des coefficients synaptiques dans les cas où l'atténuateur de couple est fixé à 0,01 et 0,001. Ces courbes d'évolution présentent des changement beaucoup plus lents, permettant aux poids synaptiques de s'adapter beaucoup plus facilement aux conditions de mémorisation. Ces conditions de mémorisation changent très peu mais ne sont pas pour autant gelées par la faiblesse relative de l'atténuateur et évolue au gré de l'erreur globale du réseau. Le lecteur pourra constater dans le graphique de droite de la figure 7.28 la faible variation des coefficients de feedback du réseau.

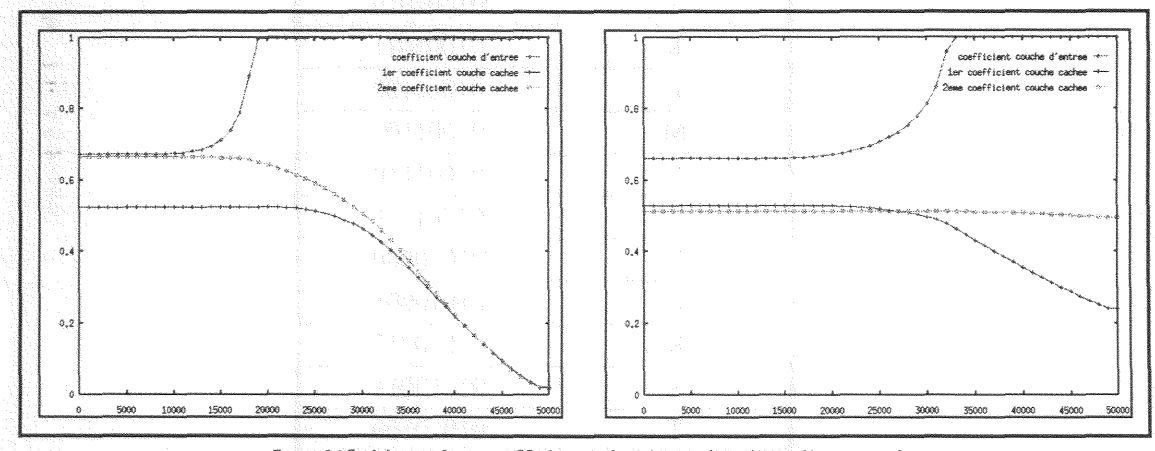


Figure 7.28 : Les définitions des coefficients de régression lors d'apprentissages avec un atténuateur de couple à 0,01 (gauche) et à 0,001 (droite).

Nous ne présenterons pas ici de résultats relatifs à l'heuristique du nombre de deltas calculés avant chaque rétropropagation du gradient selon le principe développé par [catfolis93]. Cette heuristique nécessite en effet que l'apprentissage soit effectué avec la méthode RTRL qui donne d'assez mauvais résultats en apprentissage de séquences, la BPTT étant bien mieux adaptée au problème. De plus, la faible longueur des séquences étudiées ne permet pas de mettre cette heuristique en œuvre de manière équitable vis-à-vis de sa concurrente directe.

7.6.3/ Modélisation du code ASCII

7.6.3.1/ Présentation du problème

Les expériences sur le code ASCII (*American Standard Code for Information Interchange*, [cerf69]) ont pour but de vérifier que le réseau est capable d'apprendre correctement des problèmes tels que nous allons les présenter et ainsi de vérifier la puissance des différentes possibilités de mise

en œuvre de l'algorithme d'apprentissage.

Seule une sous-partie du code ASCII complet a été utilisée et nous avons choisi de prendre comme corpus de données les 26 caractères majuscules. La table de codification de ces différents caractères est donnée ci-dessous (table 7.9). Ce corpus constitue, lui aussi, un ensemble de séries temporelles. Ces séries sont toutes différentes les unes des autres dans les 5 derniers caractères, sur 8, et aucune règle ne permet de les distinguer à priori puisque le code a été construit de manière hermétique. Le réseau peut bien tenter de les modéliser en utilisant la simple constatation du fait que ces caractères sont répartis entre 65 et 90 en base décimale mais le cumul de la valeur, même s'il est rapporté sur 0, ne peut se faire que de manière progressive, au fur et à mesure de la présentation des différents constituants de la série à analyser.

Lettre majuscule	Codification ASCII
A	01000001
B	01000010
C	01000011
D	01000100
E	01000101
F	01000110
G	01000111
H	01001000
I	01001001
J	01001010
K	01001011
L	01001100
M	01001101
N	01001110
O	01001111
P	01010000
Q	01010001
R	01010010
S	01010011
T	01010100
U	01010101
V	01010110
W	01010111
X	01011000
Y	01011001
Z	01011010

Table 7.9 : Les codes ASCII des caractères majuscules.

Toutes ces séries peuvent être représentées par l'intermédiaire d'un graphe qui modélise l'automate à mettre en œuvre. Ce graphe d'automate est présenté à la figure 7.29. Tous les états de rejet et tous les états d'acceptation y ont été représentés (cf. paragraphe 7.5.2 pour une présentation).

La figure 7.29 présente un sur-ensemble des problèmes que nous allons étudier au sujet de la modélisation du code ASCII. Nous n'avons ainsi pas défini une tâche sur l'acceptation ou le rejet de la séquence temporelle comme étant un code de caractère majuscule valide ou non. Notre étude s'est

uniquement portée sur les capacités de reconnaissance et de modélisation de caractères ASCII valides et seules des chaînes correctes ont été présentées au réseau. Les états de rejet de la figure 7.29 ne nous ont donc pas intéressé et seuls les états d'acceptation de la figure 7.29 ont été modélisés.

Il faut noter que la tâche de reconnaissance ou de transcodage du code ASCII est plus ou moins difficile en fonction de l'ordre de présentation de la séquence. Ainsi, si les bits de poids fort sont présentés avant les bits de poids faible de la série, il n'y a aucun besoin de développer des capacités de mémorisation forte puisque les trois bits de poids fort sont non significatifs. À l'inverse, lorsque les bits de poids faible sont présentés avant les bits de poids fort, les capacités de mémorisation doivent être développées de façon beaucoup plus importante de manière à ce que le passage des bits de poids fort dans le réseau en fin de présentation ne vienne pas trop perturber les informations déjà stockées sur les bits de poids faible. La présentation de gauche à droite des séries, bits de poids fort en premier, est donc a priori beaucoup plus simple à réaliser que la présentation de droite à gauche, cas où les bits de poids fort sont présentés en dernier et où les bits de poids faible sont, par conséquent, beaucoup plus difficile à retenir. Nous ne nous sommes intéressés qu'aux cas les plus simples, ces cas nous ayant déjà posés beaucoup de problèmes...

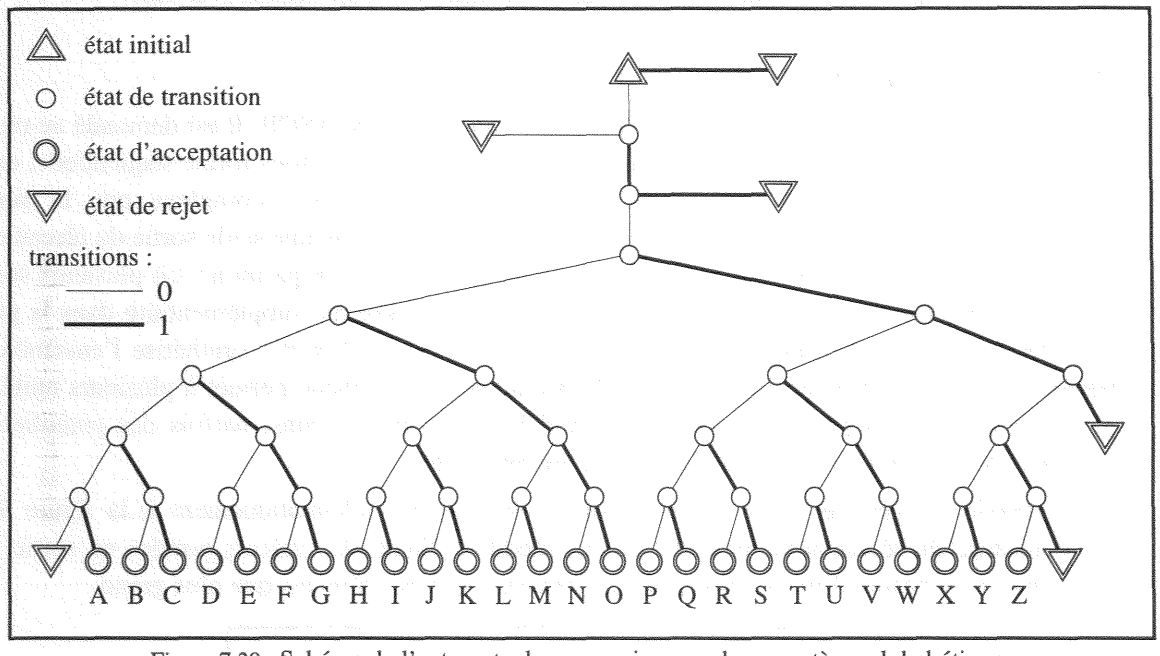


Figure 7.29 : Schéma de l'automate de reconnaissance des caractères alphabétiques majuscules de code ASCII.

7.6.3.2/ Classification du code ASCII

Les premiers tests portent sur une tâche de simple classification des 26 éléments du corpus. Dans ce cas, l'expérience est équivalente à celles menées précédemment puisque la tâche du réseau gamma est de trouver la bonne sortie parmi toutes celles qui sont possibles. La tâche est cependant plus difficile à résoudre que celles présentées au paragraphe 7.6.1 puisque, contrairement à toutes les séries présentées alors, l'information à retenir par le réseau est désormais présente dans cinq des huit éléments de la série alors que seul un élément de la séquence devait être retenue pour effectuer correctement la classification dans les cas précédents.

Nous présentons une schématisation de cette tâche dans le graphique de la figure 7.30. L'entrée du réseau, constituée d'une seule unité gamma se voit fournir successivement l'ensemble des éléments de la séquence. Les unités gamma de la couche cachée assure, elles, une mémorisation à plus long terme des données qui ont été fournies en entrée et, enfin, la couche de sortie assure la reconnaissance de la séquence par activation du neurone, d'architecture standard, attaché au caractère ASCII correspondant.

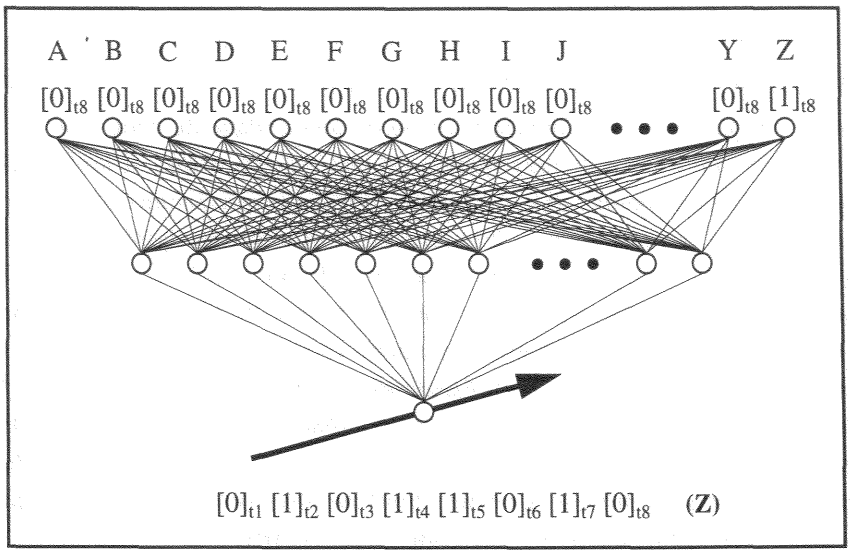


Figure 7.30 : Reconnaissance de séquences. Schéma d'une application pour une classification de type 1 parmi n.

7.6.3.3/ Transcodage du code ASCII

La deuxième série de tests portent sur la transformation du code ASCII. Il est demandé au réseau d'effectuer un transcodage et de transformer le code ASCII, présenté sous forme séquentielle, en un code ASCII représenté de manière parallèle. Cette tâche est plus complexe que la simple classification puisque la tâche n'est plus ici de nature "1 parmi n", où une seule sortie du réseau est à 1 pendant que toutes les autres sont à 0, mais une tâche de nature "m parmi n", où plusieurs sorties du réseau sont simultanément à 1. Ce choix apporte une complexité supplémentaire dans la phase d'apprentissage. En effet, alors que dans le premier cas une seule sortie synthétise l'ensemble du potentiel de modification positif des poids, le deuxième type de tâche permet à plusieurs unités de sortie de générer des potentiels positifs de modification, générant ainsi parfois des potentiels de modification contradictoires lors de la phase d'apprentissage.

Ce deuxième type de tâche, le transcodage, est présenté schématiquement à la figure 7.31. L'architecture du réseau n'est pas, elle-même, modifiée. Seules les valeurs portées en sortie sont différentes et le besoin d'un algorithme d'apprentissage efficace n'en est que plus grand.

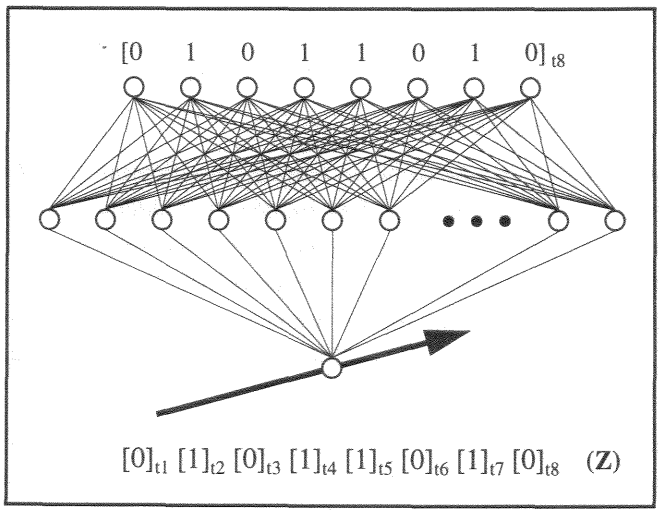


Figure 7.31 : Reconnaissance de séquences. Schéma d'une application pour une transformation du séquentiel au parallèle.

7.6.3.4/ Résultats obtenus en modélisation du code ASCII

Les tests effectués sur la classification ou le transcodage du code ASCII sont assez simples à présenter. Nous n'avons pas effectué d'apprentissage en milieu bruité comme nous en avons effectué pour les tests de reconnaissance de séquences temporelles. Nous avons eu, en effet, le plus grand mal à faire converger notre réseau, que la tâche soit un transcodage ou une identification.

Cette difficulté résulte d'un fait simple à comprendre : contrairement au cas des séquences temporelles, l'information à retenir se trouve à chacun des cinq derniers pas de temps de la séquence. Il faut donc stocker, ou mémoriser, chacun de ces éléments de manière efficace pour être capable, au dernier pas de temps, de donner une réponse correcte. Il ne s'agit donc plus de trouver une frontière délimitant deux classes dans l'espace des activations possibles mais de trouver toutes les frontières dans un espace d'activations qui ne doit pas comporter de vides. Le terme de vide est ici à mettre en rapport avec la constitution des 26 lettres du code ASCII qui sont définies linéairement par incrémentation à partir de la base représentant la première lettre.

Chacune des unités du réseau doit donc participer au processus de mémorisation mais nous ne saurions dire si les unités des couches cachées se spécialisent sur la mémorisation d'un élément de la séquence ou, plutôt, si chacune des unités participe à une mémorisation diffuse des éléments de la séquence. En outre, il semble que la tâche de transcodage pose encore plus de problèmes que la tâche de classification. Nous pensons que cette difficulté résulte de la définition des sorties qui génèrent des erreurs différentes de celles générées par la tâche de classification. En effet, plusieurs sorties peuvent désormais être à 1 et le processus de recombinaison des erreurs semble alors amoindrir l'efficacité de la rétropropagation. Ceci doit être, à notre avis, mis en relation avec le fait que le gradient d'erreur n'est pas calculé à partir d'une erreur quadratique mais à partir de la simple soustraction de la réponse par rapport à la cible (cf. annexe 1, paragraphe A1.2.1.2). Cette difficulté particulière est venue, naturellement, s'ajouter à notre problème plus général de convergence.

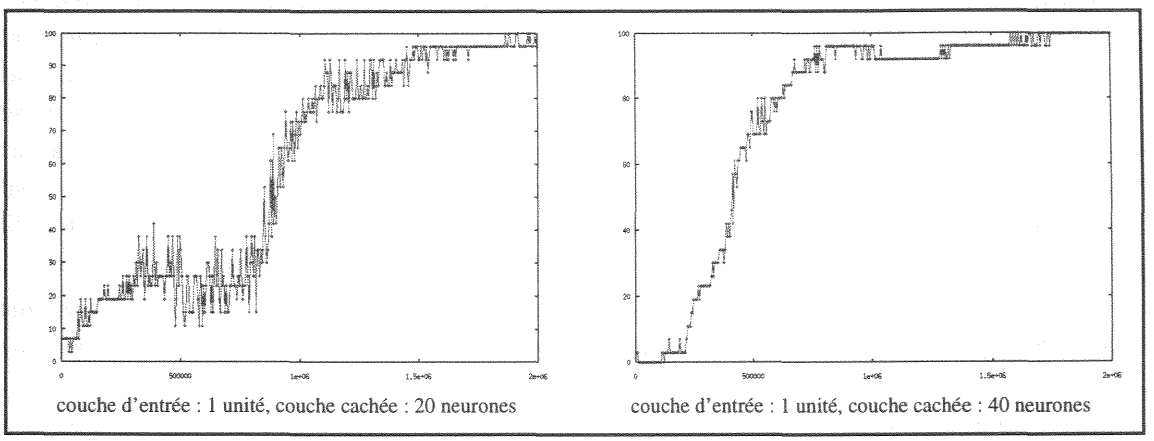
Ce problème de convergence peut se comprendre à partir de résultats que nous ne présenterons pas... Nos premiers tests, qui ont été forts nombreux, ont été réalisés en employant un apprentissage connexionniste de type RTRL (cf. paragraphe 7.2.5.2 et paragraphe 7.4.2). La mise au point a été particulièrement longue et rares ont été les apprentissages parfaitement convergents pour la seule tâche d'identification, la tâche de transcodage semblant, alors, hors d'atteinte. L'apprentissage RTRL était alors le seul que nous avions jugé bon d'implanter puisque celui-ci nous semble d'un emploi naturel en parole. Nous avons alors décidé d'implanter un algorithme d'apprentissage selon la méthode BPTT avec laquelle de bons résultats avaient été obtenus par ailleurs [lefebvre93] (cf. paragraphe 7.2.5.3 et paragraphe 7.4.3). Les capacités d'apprentissage de notre réseau se sont alors grandement améliorées même si celui-ci reste très dépendant des conditions initiales obtenues par initialisation aléatoire des poids connexionnistes et des coefficients de régression. Ainsi, nombre d'apprentissages ne convergent pas malgré nos efforts pour trouver une méthode d'initialisation permettant de maximiser la probabilité a priori d'obtenir une bonne convergence qui semble être fonction de l'espace couvert par la procédure d'initialisation. Nos tentatives pour définir les poids dans un espace très restreint ne nous aura permis que de très légèrement améliorer cette probabilité a priori mais ne nous a pas permis de la maximiser. Cette dépendance de la qualité de l'apprentissage dans les réseaux récurrents vis-à-vis des conditions initiales a d'ailleurs été observée par ailleurs [massone95]. Rappelons que la qualité des apprentissages réalisés avec la BPTT n'est pas surprenante puisque cette méthode d'apprentissage est particulièrement bien adaptée à ce type de problèmes, le réseau étant recopié pendant toute la phase de présentation de la séquence avant que la rétropropagation ajuste les poids avec les valeurs exactes des gradients et détermine la valeur effective de chaque poids par calcul de la moyenne de toutes ses copies.

Les résultats obtenus sont assez satisfaisants, en cas de convergence. Il nous a ainsi été possible de faire parfaitement converger un réseau aussi bien pour la tâche d'identification d'un élément du code

que pour la tâche de transcodage. Les graphiques présentés ci-après ne concernent que la seule BPTT avec des réseaux d'architectures variées. Nous avons ainsi employé des réseaux à une ou deux couches cachées, possédant plus ou moins d'unités tant dans la ou les couches cachées qu'en couche d'entrée où nous avons étudié les capacités d'apprentissage avec une ou deux unités gamma.

La figure 7.32 montre ainsi la qualité des apprentissages effectués dans le cadre de la tâche d'identification avec des réseaux à une couche cachée et une seule unité en couche d'entrée. La couche cachée du réseau du graphique de gauche comportait 20 neurones gamma tandis que le réseau du graphique de droite en comportait 40. Les résultats sont présentés sur les premiers 2.000.000 pas de temps. Le coefficient μ a été borné par 1 et 0,1 le coefficient d'apprentissage a été fixé à 0,3, le momentum à 0,9 et l'atténuateur de couple à 5.10^{-5}.

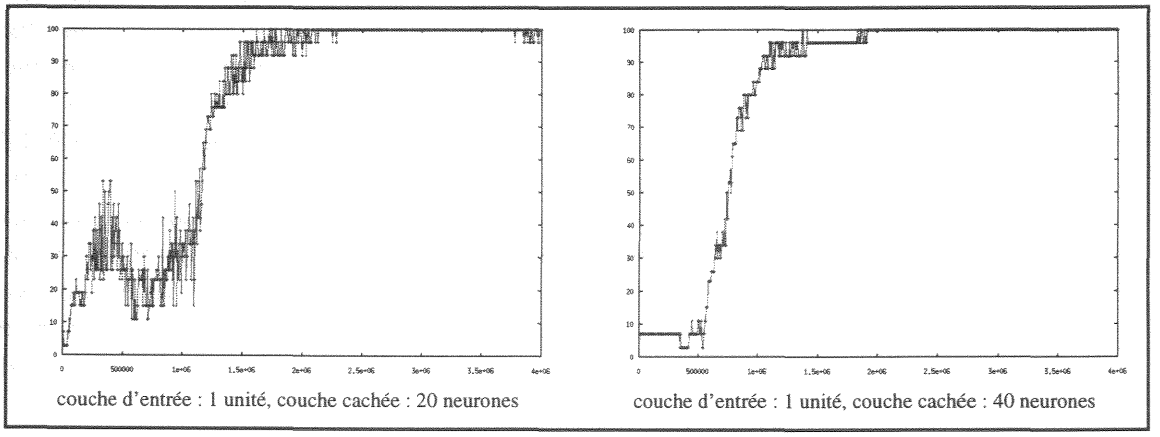
Les résultats montrent, de manière rassurante, qu'un grand nombre d'unités gamma en couche cachée permet d'améliorer le résultat par rapport à un réseau en comportant moins. L'apprentissage est cependant très lent, probablement à cause de la très faible valeur de l'atténuateur de couple qui ne permet que d'infimes modifications des coefficients de feedback dans le réseau à chaque rétropropagation. Ce très faible atténuateur est cependant une des raisons du succès des apprentissages présentés.



couche d'entrée : 1 unité, couche cachée : 20 neurones — couche d'entrée : 1 unité, couche cachée : 40 neurones

Figure 7.32 : Résultats d'apprentissage de deux réseaux gamma à une couche cachée pour la tâche de classification du code ASCII, 2.000.000 d'itérations.

La figure 7.33 correspond aux mêmes apprentissages que ceux présentés à la figure 7.32 mais sur 4 et non plus 2.000.000 d'itérations. Le comportement de la courbe d'apprentissage est très proche des courbes de la figure 7.32 même si le réseau à 20 neurones en couche cachée semble avoir eu une initialisation de meilleure qualité que celle du réseau correspondant dans la figure 7.32.



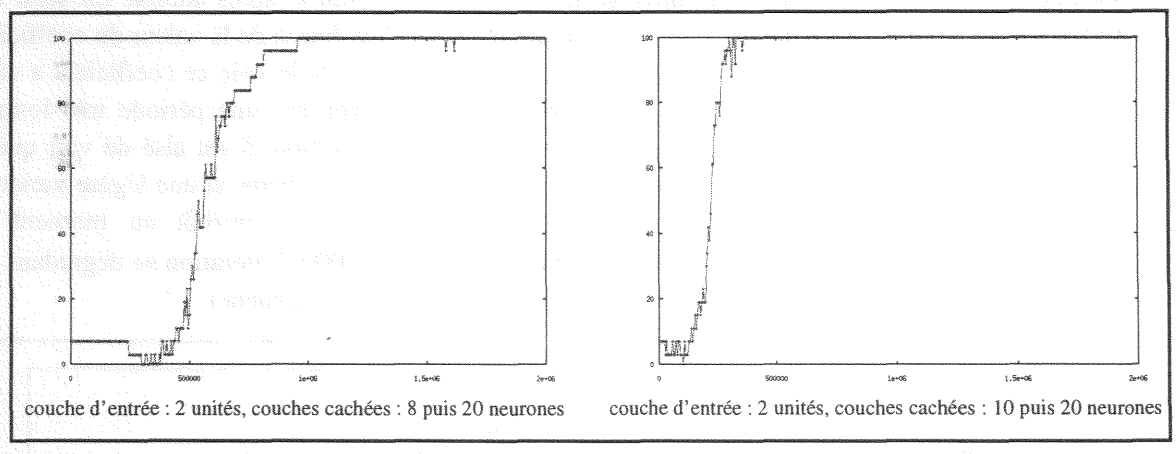
couche d'entrée : 1 unité, couche cachée : 20 neurones — couche d'entrée : 1 unité, couche cachée : 40 neurones

Figure 7.33 : Résultats d'apprentissage de deux réseaux gamma à une couche cachée pour la tâche de classification du code ASCII, 4.000.000 d'itérations.

Un fait intéressant à remarquer dans le graphique de gauche est la diminution de la qualité de l'apprentissage puisque le réseau, après avoir atteint 100%, décline quelque peu vers la 4.000.000ème itération. Cette diminution n'est pas à rapprocher du concept d'*overtraining* qui veut qu'un perceptron, lors du perfectionnement de son apprentissage, voit ses qualités de classification diminuer sur le corpus de test après passage par un maximum. Ce comportement est simplement dû à une perte réelle de qualité par prise en compte de l'erreur résiduelle obtenue sur le corpus d'apprentissage lors de chaque présentation du corpus après convergence à 100%. Cette perte doit très probablement être mise en rapport avec l'espace interne de représentation qui est évidemment plus faible pour un réseau à 20 neurones en couche cachée que pour un réseaux à 40 neurones dans cette même couche, la courbe d'apprentissage ne présentant pas de perte de qualité dans ce dernier cas (cf. figure 7.33, graphique de droite).

La figure 7.34 permet de présenter deux apprentissages particuliers qui peuvent être rapprochés de ceux présentés à la figure 7.32. Ce graphique présente les résultats d'apprentissages effectués avec des réseaux possédant 2 unités gamma en couche d'entrée et 2 couches cachées de 8 et 20 neurones gamma d'une part et de 10 et 20 neurones gamma d'autre part. L'adjonction de toutes ces unités et neurones nous a permis d'obtenir, dans l'ensemble, des résultats d'aussi bonne qualité avec un temps d'apprentissage plus court.

Curieusement, l'adjonction de 2 neurones supplémentaires en première couche cachée permet d'accélérer l'apprentissage d'un facteur 2 entre le graphique de gauche et le graphique de droite. Cette fois encore, l'initialisation des poids doit être tenue pour responsable de ce comportement plus que ne doit l'être la modification architecturale. Il est intéressant de noter la petite baisse des performances vers le 1.600.000ème itération dans le graphique de droite, perte de qualité qui est par la suite reprise mais qui peut être rapprochée de la baisse de performances du graphique de gauche de la figure 7.33.



couche d'entrée : 2 unités, couches cachées : 8 puis 20 neurones

couche d'entrée : 2 unités, couches cachées : 10 puis 20 neurones

Figure 7.34 : Résultats d'apprentissage de deux réseaux gamma à deux couches cachées pour la tâche de classification du code ASCII, 2.000.000 d'itérations.

L'avant dernier graphique de ce paragraphe, la figure 7.35, présente la courbe d'apprentissage obtenue lors de l'apprentissage de la tâche de transcodage du code ASCII avec un réseau composé d'une unité gamma en couche d'entrée et de 40 neurones gamma en couche cachée. Les paramètres de la procédure d'apprentissage varient par rapport à ceux utilisés jusqu'à présent : l'atténuateur de couple n'est, en effet, plus égal à 5.10^{-5} mais à 5.10^{-7}. Cet atténuateur de très faible valeur entraîne un gel presque total des coefficients de feedback qui ne varient alors quasiment plus. La variation maximale observée pour les coefficients de régression en couche cachée lors de cet apprentissage est de 0,18, la variation minimale étant de 0 avec une précision de 10^{-4}, la moyenne des variations étant de 0,02.

Le résultat présenté à la figure 7.35 est, en tout état de cause, de très bonne qualité puisqu'il a

permis de parfaitement apprendre la tâche du transcodage en moins de 600.000 itérations. Mais ce bon résultat ne doit pas faire oublier les nombreux autres apprentissages aux résultats moyens, voire médiocres, que nous avons obtenu.

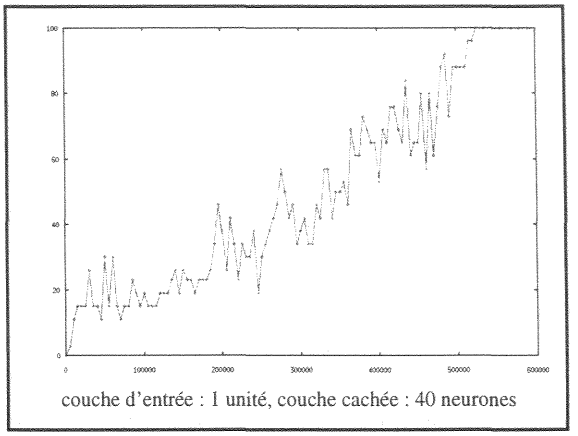


Figure 7.35 : Résultats d'apprentissage d'un réseau gamma à une couche cachée pour la tâche de transcodage du code ASCII, 600.000 itérations.

Pour approfondir les remarques qui viennent d'être faites sur la variation des coefficients de feedback, nous allons présenter le seul coefficient qui puissent être présenté isolément : celui de la plaque d'entrée qui ne comporte qu'une seule unité. Nous reprenons pour cela la courbe d'apprentissage que nous avons présentée à la figure 7.35 mais sur toute la durée de l'apprentissage qui comprend un peu moins de 2.000.000 d'itérations (très exactement 1.925.014 en raison d'un arrêt de la machine). Cette courbe d'apprentissage, présentée en partie gauche de la figure 7.36, montre que le réseau a été capable d'apprendre la tâche du transcodage du code ASCII en moins de 600.000 itérations et que cette apprentissage n'a pas perdu en qualité après atteinte du seuil des 100%. La partie droite de la figure 7.36 permet d'appréhender l'évolution de la valeur du coefficient de rétropropagation dans la plaque d'entrée. Comme il est possible de le voir, ce coefficient a varié de manière assez importante même si cette variation est observée sur une période très longue, relativisant ainsi l'importance des changements à l'échelle de l'itération. Il est aisé de voir que le coefficient de feedback finit par s'immobiliser aux alentours de 0,275 même si une légère variation peut, encore, être observée par la suite. Cette immobilisation se produit au moment où l'apprentissage converge, la lente variation observable après la 600.000^{ème} itération ne dégradant pas les résultats de l'apprentissage comme cela peut arriver (cf. figure 7.33 gauche).

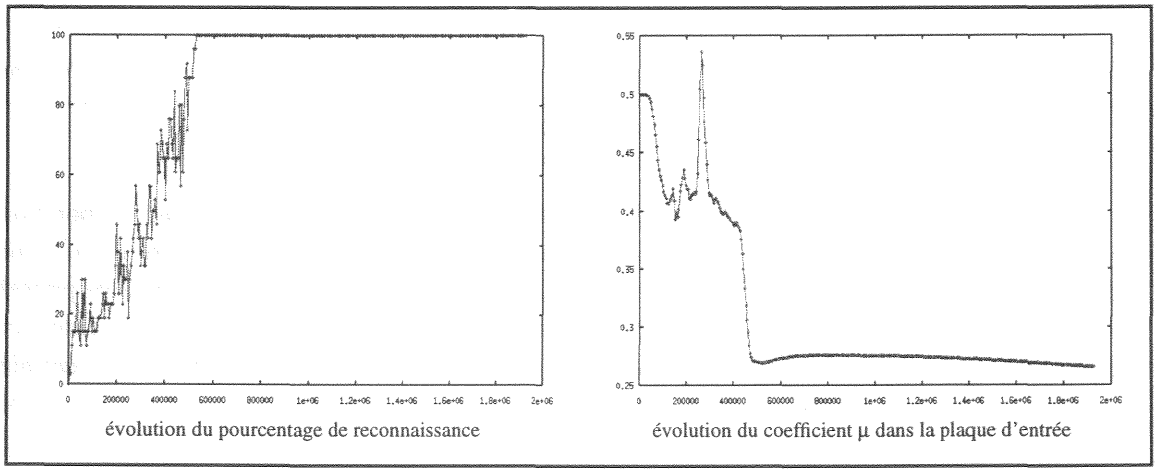


Figure 7.36 : Résultats d'apprentissage d'un réseau gamma à une couche cachée pour la tâche de transcodage du code ASCII sur 2.000.000 d'itérations avec présentation du graphe des variations du coefficient de régression.

Dans le dernier cas que nous venons de présenter, la profondeur de la plaque d'entrée à la 2.000.000^{ème} itération est égale à :

$$D = 1/0{,}2656 = 3{,}765$$

Ceci signifie qu'il aurait fallu 4 délais encastrés pour réaliser la même tâche avec un TDNN, sachant que les unités de la couche cachée implantent également un mécanisme de mémorisation. Rappelons qu'il faut un total de 5 délais pour résoudre le problème : la seule couche d'entrée ne permet donc pas de le faire totalement ici, prouvant l'intérêt du choix architectural fait pour la couche cachée.

7.7/ Application du modèle gamma à la parole

7.7.1/ Présentation du problème

Nous avons utilisé et étendu le modèle gamma avec l'espoir que sa capacité théorique à améliorer l'architecture du TDNN et à modéliser le temps de manière fine nous permettrait d'obtenir de meilleurs résultats dans la phase de segmentation que ceux que nous obtenions avec notre perceptron multicouche dont la courbe de sortie était lissée. Les résultats qui suivent ne sont malheureusement pas à la hauteur de nos espoirs puisque les résultats sont assez moyens et très uniformes. Nous pensons cependant que ce modèle pourrait voir ses capacités grandement améliorées par l'emploi d'une procédure d'apprentissage qui lui soit véritablement adaptée. La seule rétropropagation du gradient d'erreur nous semble en effet inadaptée aux réseaux faisant intervenir la dynamique du signal observé en entrée. La définition d'une heuristique dissociant le rythme d'apprentissage des différents paramètres du réseau va d'ailleurs dans le sens d'une confirmation de cette inadaptation.

Les résultats que nous avons réussi à obtenir jusqu'ici sont néanmoins intéressants car ils permettent d'observer des comportements d'adaptation qui sont impossibles à obtenir avec des réseaux statiques comme les perceptrons ou les TDNN. Comme nous le verrons dans le paragraphe suivant présentant nos résultats en segmentation simple, le comportement moyen d'une ligne de délais gamma placée en couche d'entrée est très intéressant sous certaines conditions de partage des poids.

7.7.2/ Segmentation simple

La présence du filtre gamma dans un réseau effectuant une tâche de segmentation n'a plus uniquement pour but de mémoriser des traces du signal d'entrée comme lors de nos tâches de reconnaissance de séries temporelles. Cette mémorisation doit désormais également servir à définir un mécanisme d'inhibition retardée qui permet d'empêcher une tenue trop longue des différentes sorties du réseau. L'information mémorisée ne doit donc plus uniquement servir à reconstruire une sortie en fin de présentation d'une séquence mais doit servir à la modélisation d'une durée moyenne de tenue de sortie et faire que celle-ci soit de plus en plus difficile à tenir par le réseau à mesure que croît la durée d'une réponse. Ce mécanisme doit donc pondérer les décisions prises lors de l'intrégration des connaissances phonétiques.

Nous avons réalisé différents types d'apprentissage avec le modèle gamma en essayant plusieurs types d'architectures et de contraintes sur les coefficients de régression. Nous avons également testé ce modèle selon différentes conditions en faisant varier la taille du corpus d'apprentissage. L'architecture du réseau restait la même quelques soient les différentes définitions des architectures neuronales et des conditions de partage des coefficients. Le réseau est ainsi constitué d'une plaque d'entrée de 12 lignes de 6 délais gamma permettant de conserver les valeurs des coefficients MFCC obtenus lors de la phase de prétraitement du signal de parole. Cette première couche était surmontée d'une unique couche cachée de 9 neurones. La condition de partage des coefficients de feedback dans la plaque d'entrée pouvait être une condition de contrainte maximale, toutes les unités de la plaque partageant le même coefficient, ou une condition de contrainte intermédiaire, toutes les unités d'une même ligne partageant le même coefficient, ou enfin une condition de contrainte minimale,

toutes les unités pouvant alors définir un coefficient qui leur était propre. De même, les neurones de la couche cachée pouvaient être des neurones standards, des neurones gamma partageant tous le même coefficient de régression ou des neurones gamma possédant chacun un coefficient spécifique.

L'algorithme d'apprentissage utilisé n'est plus la rétropropagation dans le temps, BPTT, comme lors de notre étude de la reconnaissance de séries temporelles mais l'apprentissage récurrent temps réel, RTRL. Nous avons, au début de ce chapitre, critiqué la méthode RTRL car elle utilise une approximation du gradient d'erreur passé pour calculer le gradient à rétropropager à un instant t. Mais cette technique d'apprentissage est la seule réellement envisageable dans le domaine de la reconnaissance automatique de la parole puisque la sortie n'est pas valide uniquement après un certain nombre de présentations de données d'entrée mais l'est constamment, la justesse de la cible par rapport à l'entrée étant variable du fait du problème de l'ancrage du symbole associé à la sortie [harnad90]. Le mode d'apprentissage utilisé ici est donc similaire à celui qui serait employé dans une tâche d'identification de système.

Les trois tableaux de résultats que nous présentons ci-après partagent les mêmes caractéristiques d'apprentissage. Le coefficient d'apprentissage a été fixé à 0,01 tandis que le momentum était fixé à 0,9. La liste des coefficients rendus inopérant est la suivante : le delta minimum est fixé à 0, le nombre de mise à jour du delta à faire avant toute rétropropagation a été fixé à 1, la valeur d'élimination du méplat de la dérivée de la fonction non linéaire a été fixée à 0, la nombre de présentation du corpus lors de la phase de boot de convergence a été fixé à 0 et l'apprentissage des poids synaptiques et des coefficients de régression en alternat n'a pas été mis en œuvre.

Enfin, l'atténuateur de couple a été fixé à 0,1. Cette valeur semble très importante vis-à-vis de celles que nous avons employées lors de nos expériences en reconnaissance ou transcodage de séquences (cf. paragraphe 7.6) mais semble ici beaucoup plus adaptée comme nous le montrerons plus loin avec nos tests sur des corpus multilocuteurs. Cette valeur est égale à celle déjà utilisée par [renals94b] dans sa tâche de reconnaissance de la parole continue.

Les tests que nous avons effectués avec un unique locuteur en apprentissage et un unique locuteur en validation ont, bien sûr, permis d'obtenir les meilleurs résultats. Ces résultats montrent cependant un certaine uniformité quelques soient les conditions d'apprentissage employées. Nous avons retrouvé cette uniformité des résultats lors de tous nos tests ce qui tend à prouver que le mécanisme gamma est finalement assez mal exploité lors de tâches relatives à la parole et que la procédure d'apprentissage n'est donc pas encore satisfaisante.

Les tests réalisés en mode monolocuteur sont assez intéressants puisqu'ils permettent de constater que le mécanisme gamma peut obtenir de bons résultats dans des cas cependant limités en complexité. Nous avons utilisé le corpus TIMIT pour l'ensemble de nos études sur la segmentation de la parole. Pour notre étude en monolocuteur, nous avons sélectionné le locuteur masculin *dr1/mcpm0* pour l'apprentissage et le locuteur masculin *dr1/mdab0* pour le test. Pour chacun de ces locuteurs, nous avons sélectionné les phrases du corpus *si*, *diverse sentences*, qui ont la particularité d'être toutes différentes les unes des autres ce qui assure une indépendance tant phonétique que temporelle des données entre locuteurs et pour un même locuteur, chacun ne disposant cependant que de trois phrases de ce type.

Les apprentissages que nous avons réalisés, quelque soit l'architecture employée, ont parfaitement convergé puisque le taux de classification a été, à chaque fois, compris dans un intervalle allant de 96 à 99%. Ce taux est particulièrement élevé et s'explique par la petitesse du corpus. Les résultats obtenus en test sont beaucoup plus variables comme le montre la table 7.10 ci-dessous.

		Couche cachée		
		perceptron	**μ partagé**	**μ libre**
Plaque d'entrée	**μ par plaque**	97 / 67	96 / 70	97 / 73
	μ par ligne	na	na	99 / 72
	μ par unité	98 / 65	99 / 72	97 / 80

Table 7.10 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de segmentation des voyelles suivant l'architecture de réseau utilisée. Apprentissage et validation monolocuteur.

À titre d'exemple et pour permettre au lecteur de comprendre pourquoi nous pensons devoir abandonner la condition de partage des coefficients de régression en couche cachée, nous présentons ci-dessous trois jeux de coefficients de régression obtenus au cours de divers apprentissages, coefficients que nous avons triés en ordre décroissant des valeurs par souci de lisibilité. Les deux premières lignes montrent ainsi deux ensembles de coefficients de régression de la couche cachée dont seules 2 des 9 unités implantent de la mémoire, les 7 autres neurones n'en implantant presque pas. Le troisième jeu de coefficients présente, quand à lui, plus de neurones gamma dont les coefficients permettent d'implanter une mémoire conséquente.

1 : {0,9761 ; 0,9649 ; 0,9591 ; 0,9445 ; 0,9235 ; 0,9147 ; 0,9057 ; 0,8915 ; 0,6790}
2 : {0,9981 ; 0,9957 ; 0,9677 ; 0,9638 ; 0,9492 ; 0,9420 ; 0,9398 ; 0,8805 ; 0,7245}
3 : {0,9999 ; 0,9993 ; 0,9933 ; 0,8991 ; 0,8814 ; 0,7695 ; 0,6668 ; 0,4982 ; 0,4083}

De telles tendances peuvent se retrouver dans le cas où un même coefficient est partagé par l'ensemble des neurones de la couche cachée. Le problème, dans ce cas, est que cet unique coefficient aura une forte probabilité d'être égal à 1 en fin d'apprentissage, comme était forte la probabilité qu'un coefficient non contraint par la condition de partage soit égal à 1. Ainsi, sur 19 tests que nous avons réalisés avec cette condition de partage des coefficients, seuls 3 n'ont pas positionné l'unique coefficient de régression de la couche cachée à 1.

La qualité de la segmentation obtenue avec un réseau gamma sur les phrases de l'unique locuteur d'apprentissage pourra être constatée grâce à la figure 7.37. Cette courbe est presque parfaite mais ce type de résultats n'est malheureusement pas généralisable à des corpus de taille plus importante.

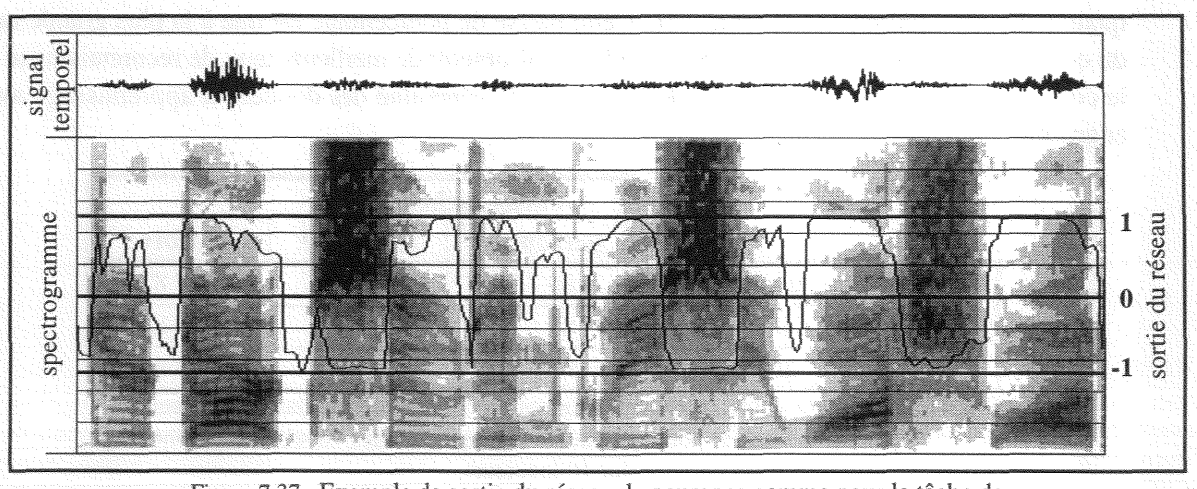


Figure 7.37 : Exemple de sortie du réseau de neurones gamma pour la tâche de segmentation des voyelles en mode monolocuteur.

Un point intéressant à remarquer, dans cet apprentissage d'envergure limitée comme lors d'autres apprentissages plus conséquents, est la disposition des capacités de mémorisation acquise lorsque tous les coefficients de la plaque d'entrée sont laissés libres et donc lorsqu'aucune contrainte n'est imposée. Nous avons dans ce cas pu observé un comportement moyen des coefficients de la ligne qui

est obtenu indépendamment de la phase d'initialisation aléatoire. La ligne de délais a alors tendance à implanter une mémoire de profondeur de plus en plus grande à mesure que le rang du délai augmente. Ainsi, le premier délai a une résolution maximale, la valeur du coefficient étant très proche de 1. Les délais de rang supérieur utilisent, eux, des coefficients de régression de valeur toujours plus faible, implantant par là même une mémoire de toujours plus grande profondeur. Le relâchement de toute contrainte de partage des coefficients dans la plaque d'entrée permet donc d'avoir un réseau possédant plusieurs capacités de mémorisation au sein d'une même structure, à la manière de l'architecture présentée par [harrison89] (cf. paragraphe 7.2.6). La figure 7.38 montre la disposition moyenne des coefficients de régression dans une ligne de la plaque d'entrée. Cette figure fait ressortir un comportement qui peut être observé dans une très grande majorité de cas.

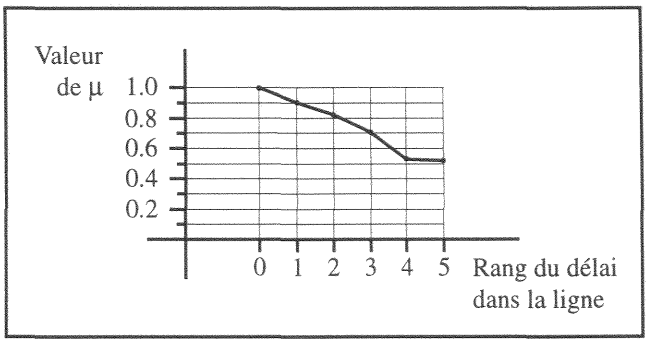


Figure 7.38 : Valeur moyenne de μ dans la ligne de délais en fonction du rang du délai dans cette ligne lors d'une tâche de segmentation monolocuteur.

Nous avons également réalisé des apprentissages multilocuteurs pour vérifier les capacités de notre modèle connexionniste sur des tâches de plus grande envergure. Nous avons réalisé une première série de tests avec 10 locuteurs en apprentissage en conservant le même et unique locuteur en test. Les locuteurs choisis pour constituer le corpus d'apprentissage sont les 10 premiers locuteurs masculins du corpus TIMIT de parole continue. Pour chacun d'eux, nous avons sélectionné la première phrase *si - diverse sentences* disponible. Le corpus de test est, comme précédemment, constitué des trois phrases *si* du locuteur de test *dr1/mdab0*.

Les résultats obtenus en apprentissage (cf. table 7.11) sont, bien évidemment, de moins bonne qualité que ceux présentés à la table 7.10. Cette baisse du pourcentage est due à la plus grande taille du corpus d'apprentissage qui permet, par ailleurs, d'obtenir de meilleurs taux de reconnaissance sur le corpus de test que ceux présentés à la table 7.10, la généralité des données d'apprentissage ayant augmenté.

		Couche cachée		
		perceptron	**μ partagé**	**μ libre**
Plaque d'entrée	**μ par plaque**	89 / 83	87 / 72	86 / 82
	μ par ligne	87 / 77	86 / 83	86 / 85
	μ par unité	88 / 87	89 / 71	89 / 86

Table 7.11 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de segmentation des voyelles. Apprentissage sur 10 locuteurs.

Les résultats que nous présentons pour ces derniers tests sont tout de même assez décevants puisque les résultats obtenus avec des neurones standards en couche cachée sont presqu'aussi bons que ceux obtenus avec un nombre maximal de degrés de liberté, architecture qui donnait les meilleurs résultats dans notre précédente étude sur la reconnaissance de séquences.

Les derniers tests réalisés en multilocuteurs ont portés sur 50 locuteurs. Ces locuteurs ont été sélectionnés dans le corpus d'apprentissage TIMIT, dans les sous-corpus *dr1* et *dr2*. Comme

précédemment, seule la première phrase de type *si - diverse sentences* a été sélectionnée pour chacun de ces locuteurs. De même, seules les 3 phrases *si* du locuteur de test *dr1/mdab0* ont été sélectionnées. La taille du corpus d'apprentissage permet de valider les résultats de manière, cette fois, tout à fait sûre puisque, pour ces derniers tests, l'incertitude de l'intervalle de confiance à 95% varie, pour les résultats d'apprentissage, de ± 0,379% à ± 0,387% tandis que cette même incertitude varie de ± 1,550% à ± 1,658% pour les résultats obtenus sur les données de validation.

		Couche cachée		
		perceptron	**μ partagé**	**μ libre**
Plaque d'entrée	**μ par plaque**	82 / 83	82 / 81	83 / 84
	μ par ligne	83 / 84	83 / 83	82 / 83
	μ par unité	83 / 81	83 / 82	82 / 82

Table 7.12 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de segmentation des voyelles. Apprentissage sur 50 locuteurs.

Les résultats obtenus (cf. table 7.12) sont, encore une fois, de moins bonne qualité en apprentissage et de meilleure qualité en test et les pourcentages obtenus sont, cette fois, équivalents puisque l'égalité est presque parfaite pour chacun des couples de pourcentages en tenant compte de la valeur de l'incertitude. De même, et peut-être plus encore qu'avant, il est facile d'observer une très grande uniformité dans les résultats, les variations architecturales ne permettant pas d'obtenir des changements significatifs dans les pourcentages obtenus. Ces résultats prouvent, une fois encore, l'apparente inutilisation du mécanisme gamma pour les tâches de segmentation de la parole.

Tous les tests qui précèdent ont été réalisés avec une valeur d'atténuateur de couple fixée à 0,1. Une baisse de cette valeur ne permet pas pour autant d'améliorer les résultats au contraire de ce que nous avons pu faire remarquer au paragraphe 7.6.2.6 au sujet de l'apprentissage des séquences. Nous avons ainsi effectué des tests avec un atténuateur de couple fixé à 0,01 avec un corpus d'apprentissage constitué des 24 locuteurs du sous-corpus *train/dr1* de TIMIT pour chacun desquels nous avons sélectionné la première phrase *si - diverse sentences*. Le corpus de test est, comme à l'habitude, constitué des trois phrases *si* du seul locuteur de test *dr1/mdab0*.

		Couche cachée		
		perceptron	**μ partagé**	**μ libre**
Plaque d'entrée	**μ par plaque**	81 / 79	73 / 70	80 / 77
	μ par ligne	81 / 79	75 / 73	77 / 75
	μ par unité	80 / 79	79 / 76	79 / 77

Table 7.13 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de segmentation des voyelles. Apprentissage sur 24 locuteurs avec un atténuateur de couple à 0,01.

Les résultats obtenus (cf. table 7.13) sont, de manière générale, de moins bonne qualité que ceux que nous avons obtenus avec un atténuateur de couple de valeur plus élevée tant pour le corpus d'apprentissage que pour le corpus de validation. L'incertitude varie, ici, de ± 0.538% à ± 0.609% pour les données d'apprentissage et de ± 1.717% à ± 1.931% pour les données de validation, cette incertitude étant, comme précédemment, relative à un intervalle de confiance à 95%. L'étude des variations des coefficients de régression montre que ceux-ci varient de manière assez peu importante du fait de la faible valeur de l'atténuateur. Cette faible variation oblige donc la procédure de rétropropagation à s'accommoder des caractéristiques initiales de mémorisation plus qu'elle ne le faisait lorsque l'atténuateur était fixé à 0,1. La bonne qualité des apprentissages effectués avec des atténuateurs de forte valeur est une des caractéristiques des tests réalisés sur des tâches de reconnaissance automatique de la parole.

7.7.3/ Pseudo segmentation

Comme lors de notre étude sur la segmentation du signal de parole par perceptron multicouche, nous avons pu remarquer que la définition de grandes classes phonétiques permettait à des phonèmes différents d'être placés côte à côte, la sortie désirée du réseau restant dans un état identique pendant une période de temps couvrant un ensemble de phonèmes d'une même grande classe. Ce problème devient très gênant puisqu'il peut désormais provoquer des distorsions tant dans le domaine phonétique, lorsque des phonèmes acoustiquement proches sont classés différemment, que dans le domaine temporel, la trop longue tenue d'une cible venant modifier la perception temporelle des phénomènes élémentaires.

Pour pallier ce problème, nous avons décidé de modifier l'étiquetage manuel pour faire ressortir clairement les débuts et les fins des segments, en intercalant entre deux phonèmes contigus appartenant à la même classe une plage de réponse opposée. Cette plage doit normalement permettre au réseau de mieux appréhender la durée des différents phénomènes élémentaires qui sont, de cette manière, présentés isolément.

Nous avons ainsi décidé de modifier les débuts et les fins des segments manuels en modifiant de 1 à 3 trames dans le cas où deux phonèmes d'une même grande classe se suivent. Cette modification se fait simplement en inversant la valeur de la sortie désirée pour les trames considérées.

Les résultats que nous présentons ont été obtenus à partir d'un corpus d'apprentissage constitué des 24 locuteurs masculins du corpus *timit/train/dr1* pour lesquels la première phrase *si - diverse sentences* a été sélectionné, dans l'ordre des répertoires et des locuteurs. Le corpus de test est, lui, constitué des premières phrases *si* des 50 premiers locuteurs masculins de *timit/test*, sous-corpus *dr1* et *dr2*. Les résultats présentés à la table 7.14 sont de deux types : la partie gauche présente les résultats obtenus au niveau vectoriel, c'est à dire au niveau des trames, sur les corpus d'apprentissage et de test tandis que la partie droite de la table présente les résultats obtenus au niveau segmental sur le corpus de test.

Ces résultats ne sont pas très probants puisque, pour les résultats segmentaux, le gain vis-à-vis du pourcentage de fusion se fait au détriment du pourcentage des segments corrects qui ne sont, à aucun moment, améliorés. Ainsi, la diminution de 7% du taux de fusion ne se traduit pas par une augmentation du taux des segments corrects mais par une diminution de 4% de ce taux et, de manière bien plus grave, par une augmentation de 11% du taux des segments élidés. Les taux de classification vectorielle baissent, eux, de manière très importante au fur et à mesure que s'accroît le nombre de trames modifiées.

nombre de trames inversées	trames		segments				
	app.	test	correct	insertion	fusion	élision	division
0	88%	86%	63%	3%	34%	3%	1%
1	83%	83%	63%	3%	33%	4%	1%
2	73%	75%	62%	2%	29%	8%	1%
3	62%	67%	59%	1%	27%	14%	1%

Table 7.14 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) en pseudo segmentation en fonction du nombre de trames modifiées.

Notre étude des sorties nous a permis de constater la trop forte brutalité des modifications que nous imposions aux différents segments, expliquant en grande partie le très fort taux d'élision que nous obtenons en modifiant 3 trames au début et à la fin de chaque segment. En effet, il se peut, dans ce cas, que certains segments disparaissent presqu'entièrement dans la séquence des sorties cibles présentée lors de l'apprentissage du fait de leurs très courtes durées initiales, rendant leur apprentissage impossible. Aussi, si nous réussissons, par nos modifications, à supprimer certains

problèmes relatifs à des segments trop longs, comme le prouve la diminution du taux des fusions, nous faisons apparaître des problèmes de suppression pour les segments plus courts, comme le montre le taux d'élision. Ceci prouve la trop grande cécité de notre méthode qui aurait peut-être été améliorée avec la mise en place d'une heuristique de durée minimale. Nous avons cependant décidé d'affiner nos classes plutôt que d'améliorer cette technique.

7.7.4/ Reconnaissance de phonèmes d'une classe

La segmentation effectuée par un réseau gamma peut être améliorée tout comme elle pouvait l'être dans le cas de la segmentation effectuée par perceptron par simple partition des classes utilisées. La segmentation ne se fait plus alors en grandes classes mais en classes de taille réduite ou, lorsque le processus est poussé à l'extrême, en autant de classes qu'il y a de phonèmes dans le vocabulaire. Cette accroissement du nombre de classes, et donc du nombre de sorties du réseau, permet d'isoler les phénomènes acoustiques tout comme nous avons essayé de la faire par modification des cibles (cf. paragraphe 7.7.3). La modification est cependant moins dangereuse puisque la sortie désirée du réseau respecte toujours la définition phonétique qui est donnée par l'étiquetage manuel. Notre pseudo segmentation précédente portait, elle, atteinte à cette définition.

Dans les tests qui suivent, nous avons tout d'abord essayé de segmenter le signal en 3 classes phonétiques que sont, premièrement, les voyelles, les semi-voyelles et les nasales, deuxièmement, les affriquées et, troisièmement, les occlusives. Tous les phonèmes sont donc regroupés en fonction de similitudes facilement identifiables. Les résultats que nous avons obtenus, et que nous présentons à la table 7.13, ont été obtenus sur de gros corpus et permettent de considérer ces pourcentages comme sûrs puisque l'incertitude maximale observée pour un intervalle de confiance à 95%, pour l'apprentissage comme pour le test, est de ± 0,455%.

Ces premiers résultats montrent l'intérêt de la mise en place d'un grand nombre de coefficients de régression, et donc de degrés de liberté, dans le réseau puisque les meilleurs résultats ont été obtenus avec un réseau dont les 12×6 unités de la plaque d'entrée et les 9 neurones gamma de la couche cachée possédaient chacun un coefficient propre. Ces résultats viennent ainsi contredire la très forte uniformité que nous avons pu observer jusqu'à présent. Mais les résultats de la table 7.13 n'améliorent pas pour autant ceux que nous avons déjà obtenus auparavant, comme ceux de la table 7.12 ou ceux de la table 7.14, prouvant ainsi que l'affinage de nos deux grandes classes en trois plus petites aura été inutile.

		Couche cachée (μ libre)
Plaque d'entrée	**μ par plaque**	71 / 72
	μ par ligne	79 / 73
	μ par unité	84 / 83

Table 7.15 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de segmentation en 3 classes phonétiques.

Nous avons également divisé les 3 classes précédentes pour en créer 4, la première étant constituée des voyelles, la deuxième des semi-voyelles et des nasales, la troisième des affriquées, la dernière étant constituée des occlusives. Ce découpage supplémentaire aurait dû, lui aussi, améliorer les résultats que nous avions obtenus précédemment. Là encore, et plus encore que dans le cas précédent avec trois classes, les résultats ne sont pas très probants bien que l'incertitude maximale d'un intervalle de confiance à 95% soit, pour l'apprentissage et le test, de ± 0,475%. Les résultats présentés à la table 7.13 sont même de très mauvaise qualité dans les cas où le nombre de degrés de liberté n'était pas maximal. Les coefficients ont donc été très mal déterminés, de même que les connexions.

		Couche cachée (μ libre)
Plaque d'entrée	**μ par plaque**	35 / 36
	μ par ligne	34 / 28
	μ par unité	73 / 71

Table 7.16 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de segmentation en 4 classes phonétiques.

Ces derniers résultats sont probablement dus à deux phénomènes conjugués : une trop petite taille de l'espace des paramètres et, surtout, une mauvaise initialisation des poids synaptiques et des coefficients de régression, cause de problèmes de convergence non négligeables [massone95]. La procédure d'apprentissage peut donc, encore une fois, être mise en cause.

Il doit être signalé que nous avons également essayé de mettre en place une heuristique de lissage de la cible de manière à ce que le changement de classe ne soit pas brutal mais plutôt progressif pour minimiser les erreurs en sortie du réseau pendant les phases de transition entre étiquettes. Ce lissage, effectué sur un nombre variable d'au plus 9 trames, doit permettre d'atténuer d'éventuels problèmes dus à l'étiquetage manuel et provoqués par les erreurs commises par l'expert en phonétique [phillips87]. Cette heuristique n'a cependant pas permis d'améliorer les résultats de manière significative.

7.7.5/ Reconnaissance des occlusives

Nous nous sommes, de manière annexe, également intéressé à la reconnaissance des occlusives pour vérifier les capacités de résolution de l'algorithme d'apprentissage. Comme nous l'avons déjà fait remarqué précédemment, le pas de temps que nous avons choisi d'utiliser pour espacer le calcul de deux trames consécutives de valeurs de prétraitement sur le signal est très court, 4 millisecondes, alors que le pas standard utilisé dans la communauté de RAP est de 10 *ms*, ce pas de temps étant même jugé trop faible par certains (cf. chapitre 4, paragraphe 4.1.2). Ce choix pour un pas de temps très court a été, à l'origine, fait pour obliger le réseau gamma à implanter de la mémoire, et donc favoriser la profondeur, puisque la tâche de segmentation étudie normalement des phénomènes tels que les voyelles qui sont des phonèmes de longue durée relativement à ce faible pas de temps et à l'étroitesse de la plaque d'entrée.

Un pas de temps très court doit cependant être utilisé dans certains cas tels que, par exemple, lors de la reconnaissance des occlusives. Ces phénomènes acoustiques se caractérisent par deux étapes successives et totalement différentes [lonchamp90]. La première partie d'une occlusive est constituée d'une barre d'explosion, résultat du relâchement des lèvres, tandis que la deuxième partie d'une occlusive est, normalement, constituée par un souffle. Ce souffle apparaît généralement lorsque l'occlusive est prononcée isolément et n'est pas suivie par une voyelle qui peut prendre la place de ce souffle lorsqu'elle est présente. L'information discriminante d'une occlusive se trouve principalement concentrée dans la barre d'explosion [djezzar95] et une analyse fine de cette barre permet d'obtenir d'excellents résultats. Le problème majeur rencontré par un système de RAP est que le pas de temps normalement utilisé est beaucoup trop important et ne permet pas d'obtenir des informations sur le signal en qualité et en nombre suffisants. Certains ont donc essayé, avec succès, de réduire ce pas de temps jusqu'à des intervalles de 3 millisecondes pour capturer suffisamment d'informations sur la barre d'explosion [fanty90].

Notre système possède, de par le pas de temps choisi initialement, de bonnes capacités intrinsèques pour discriminer correctement les occlusives. Le réseau mis en œuvre devra cependant choisir, par la procédure d'apprentissage, de ne pas trop implanter de mémoire et, donc, de maximiser la résolution sur le signal au détriment de la profondeur. Nous avons donc décidé de tester notre architecture sur ce type de tâche pour vérifier nos hypothèses et vérifier, également, les

capacités de la procédure d'apprentissage qui devra choisir des valeurs adéquates pour les coefficients de régression.

Les résultats que nous présentons à la table 7.17 ne sont pas de très bonne qualité puisque les pourcentages de reconnaissance obtenus sont assez faibles, particulièrement sur le corpus de validation. Ces résultats ne peuvent donc pas prétendre concurrencer ceux présentés dans [djezzar95] pourtant obtenus avec des réseaux statiques de type perceptrons multicouches.

		Nombre de neurones gamma en couche cachée		
		9	**11**	**13**
Nombre de filtres dans une ligne de la couche d'entrée	**2**	73 / 59	74 / 59	79 / 59
	3	76 / 60	77 / 63	81 / 61
	6	80 / 59	na	na

Table 7.17 : Résultats obtenus (pourcentage sur le corpus d'apprentissage / pourcentage sur le corpus de validation) pour la tâche de reconnaissance des occlusives en fonction du nombre de neurones gamma en couche cachée et du nombre d'unités en plaque d'entrée.

La répartition de la moyenne des coefficients de régression est également décevante puisque, si la répartition des coefficients est effectivement différente de celle obtenue lors de nos tâches de classification des voyelles (cf. paragraphe 7.7.2), la valeur moyenne des coefficients est assez faible quelque soit le rang du délai dans la ligne. Nous pensions, en commençant ces expérimentations, obtenir des coefficients de valeurs beaucoup plus élevées. Or la répartition très généralement obtenue fait apparaître des valeurs de coefficients de régression très uniformes et très faibles puisque la moyenne des valeurs présentées à la paragraphe 7.39 est approximativement de 0,55, prouvant que les unités de la ligne de délais ont opté pour une solution de profondeur plutôt que de résolution.

La répartition présentée à la figure 7.39 peut varier, les valeurs des coefficients correspondant aux délais de milieu de ligne pouvant être réparties sous la forme d'une cuvette avec des valeurs restant cependant dans un intervalle allant de 0,4 à 0,6, le dernier délai étant toujours caractérisé par une valeur très faible.

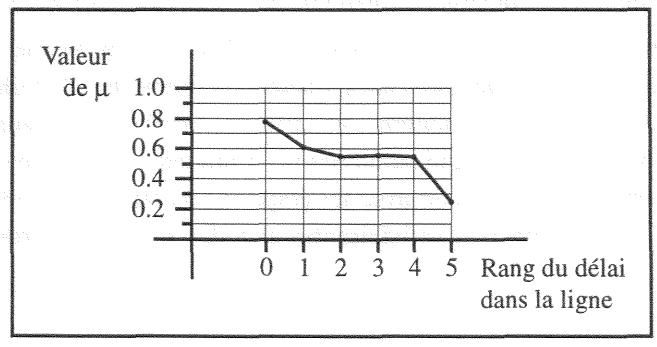


Figure 7.39 : Valeur moyenne de μ dans la ligne de délais en fonction du rang du délai dans cette ligne lors d'une tâche de reconnaissance des occlusives.

Le comportement que nous observons dans cette tâche de reconnaissance des occlusives où le besoin de résolution est largement reconnu par les études qui ont déjà été faites par ailleurs semble, une fois de plus, montrer la faiblesse de la procédure d'apprentissage qui est incapable de déterminer les poids de manière optimale par rapport à la tâche étudiée. Nous constatons ainsi que le mécanisme d'inhibition retardée n'a pas ou a été très mal implanté lors de nos tâches de segmentation de la parole tandis que le choix de valeurs de coefficients de régression permettant d'avoir une bonne résolution lors de nos tâches de reconnaissance des occlusives n'a pas été fait.

7.8/ Problèmes posés par l'algorithme d'apprentissage

Les résultats que nous avons obtenus lors de la mise en œuvre du modèle gamma sont en demi-teinte puisque nos résultats en reconnaissance de séquences temporelles ont permis d'obtenir de bons résultats, y compris avec des séquences bruitées, tandis que les résultats que nous avons obtenus en reconnaissance automatique de la parole sont de qualité assez moyenne sur des corpus qui, en outre, ne sont pas bruités.

Le mécanisme que nous avons utilisé, le filtre gamma, a ainsi été capable de mémoriser de l'information par décroissance exponentielle de l'activité interne et a ainsi pu classer des séquences temporelles après la présentation du dernier élément de celles-ci. Il n'a cependant pas, à notre avis, été capable d'implanter une inhibition retardée permettant de modéliser la durée de phénomènes, modélisation qui nous aurait permis de mettre en œuvre une étape de segmentation prenant en compte, en une seule fois, des notions phonétiques et temporelles. Cette union de deux types de connaissances aurait dû permettre à chacune de pondérer des décisions uniquement fondées sur l'autre. La grande uniformité des résultats prouve que le nombre variable de degrés de liberté accordé au réseau n'a pas permis d'obtenir une amélioration des résultats observés dans les cas où ce nombre était limité.

Les résultats ont été obtenus avec deux procédures d'apprentissage différentes qui semblent, à chaque fois, très bien adaptées aux problèmes étudiés. Nous avons ainsi utilisé un apprentissage récurrent temps réel pour nos problèmes de segmentation de la parole, car ils sont proches de problèmes d'identification de systèmes, alors que nous avons utilisé une rétropropagation dans le temps pour nos tâches de reconnaissance de séquences car la sortie désirée n'est connue que lorsque le dernier élément de la séquence est présenté.

Un échange des procédures d'apprentissage ne permet pas d'obtenir de meilleurs résultats. La rétropropagation dans le temps a ainsi été développé après constatation des piètres résultats obtenus par l'apprentissage récurrent temps réel en classification de séquences. À l'inverse, la rétropropagation dans le temps est difficilement applicable au domaine de la parole puisque les phénomènes observés possèdent des réponses associées, les symboles, qui sont valables pendant toute la durée de l'intervalle couvrant le phénomène et non pas uniquement à la fin de cet intervalle.

Au terme de cette étude, il semble que nous avons abordé un problème autre que celui qui était le nôtre au début de cette thèse et qui nous aura, finalement, empêché d'atteindre des résultats probants avec une architecture qui nous semble, en théorie, parfaitement adaptée à la segmentation automatique de la parole bruitée et qui aurait permis d'atteindre, en cas de succès, de bons résultats si elle avait été mise en œuvre de concert avec les étapes de reconnaissance des voyelles et de mots qui étaient effectuées de manière très satisfaisantes par des réseaux neuromimétiques statiques (cf. chapitre 4).

CHAPITRE 8 : DÉVELOPPEMENTS ULTÉRIEURS

"And yet... do you communicate often with machines by voice? More to the point [...] do your parents or neighbours? If the answer is no, then perhaps there's a mismatch between our expectations of speech recognition and today's reality."

Melvyn Hunt, directeur de Dragon System UK
Real speech recognition technology doesn't need hype

"Rien n'est plus pénible à surmonter que les difficultés que l'on croyait surmontées."

Alexis de Tocqueville

Résumé

Ce chapitre de conclusions nous permettra de donner un point de vue sur la thèse qui a été développée et sur ce que nous aurions voulu y développer si notre exploration des mécanismes et des lois d'apprentissage dans les systèmes dynamiques non linéaires avait abouti. Ainsi présenterons nous quelques modèles que nous avions envisagé d'étudier et que nous avons, finalement, laissé au domaine de l'inconnu et de l'inexploré.

8.1/ Conclusions de la thèse

8.1.1/ Réseaux connexionnistes statiques

Nos expériences de reconnaissance de petits vocabulaires en milieu bruité à l'aide de réseaux de neurones statiques du type perceptron multicouche se révèlent, dans l'ensemble, positives. Les résultats obtenus tant en segmentation en grandes classes par perceptrons multicouches qu'en reconnaissance de voyelles et de mots par perceptrons ou *Selectively Trained Neural Network* permettent de considérer ce type d'architecture comme une solution envisageable jusqu'à des rapports signal-sur-bruit assez faibles c'est à dire de l'ordre de six décibels et, ce, avec des bruits stationnaires tout autant que non stationnaires.

L'architecture développée ne peut cependant pas être appliquée avec succès à des rapports signal-sur-bruit inférieurs à six décibels sans faire apparaître de graves problèmes, tout particulièrement dans l'étape initiale de segmentation par perceptron multicouche. Les durées des segments deviennent alors très peu plausibles du fait de leur grande taille, le phénomène étant provoqué par l'agrégation de plusieurs segments les uns aux autres, le bruit masquant les frontières au regard du réseau en charge de la segmentation. Cette constatation est la raison qui nous pousse à croire que d'autres recherches seront nécessaires pour améliorer plus encore les systèmes de reconnaissance de petits vocabulaires tels que les lettres ou les chiffres épelés en milieu bruité. Des

recherches connexes sur les capacités de perception humaine prouvent en effet que ces tâches peuvent être résolues de manière excellente par l'homme jusqu'à des rapports signal-sur-bruit de cinq décibels négatifs c'est à dire lorsque le bruit, un spectre de parole à long terme dans ce cas, est plus important que le signal de parole porteur de l'information. Les capacités de reconnaissance humaine s'amenuisent cependant très rapidement une fois le cap de cinq décibels négatifs dépassé vers des rapports signal-sur-bruit toujours plus négatifs.

L'observation des résultats obtenus avec les perceptrons dans la phase de segmentation à de très faibles rapports signal-sur-bruit nous a poussé à vouloir augmenter les capacités de cette étape de segmentation par adjonction d'une capacité de modélisation temporelle pour que le réseau en charge de la segmentation ait les capacités d'apprendre la durée la plus plausible des événements à segmenter tout autant que la forme des événements dans un plan phonétique statique.

L'extension architecturale et fonctionnelle que nous voulions mettre en œuvre pouvait se faire de plusieurs manières. Il aurait ainsi été possible de mettre en place deux systèmes successifs dont un aurait permis une classification basée sur des connaissances phonétiques tandis qu'un autre aurait implanté une méthode permettant d'approcher les durées les plus plausibles des phonèmes par scission des noyaux trop longs en d'autres plus petits par respect strict d'un calcul statistique de la durée et sans aucune connaissance phonétique. Mais nous avons préféré essayer d'intégrer ces deux sources de connaissance, phonétique et temporelle, au sein d'un même système qui devait donc pouvoir garder les acquis du système existant tout en possédant des capacités de modélisation temporelle. Ce choix nous a conduit a étudier les systèmes connexionnistes dynamiques, aptes à répondre à notre besoin.

8.1.2/ Réseaux connexionnistes dynamiques

Pour résoudre le problème qui nous était posé dans le domaine temporel, nous avons décidé d'employer un réseau connexionniste à récurrence locale. Ce type d'architecture est encore assez peu répandu, le développement des premières architectures de ce type ayant commencé il y a fort peu d'années. Nous avons jugé ce type d'architectures dynamiques plus intéressant que d'autres architectures comme celles des réseaux fortement récurrents ou celles des réseaux à récurrence par plaque qui nous semble moins apte à modéliser des notions temporelles.

Qui plus est, l'architecture gamma et les extensions que nous avons apportées à ce modèle nous semblent être d'excellentes solutions pour la modélisation répartie de phénomènes de durée. La modélisation répartie de la durée impose bien entendu de conserver le caractère boite noire du réseau connexionniste choisi, permettant aux détracteurs du connexionnisme de critiquer plus encore cette architecture que celle des perceptrons multicouches. Nous voyons cependant quelques avantages à utiliser cette solution sur lesquels nous nous étendrons dans le paragraphe 8.2.1.

Cependant, le choix que nous avons effectué pour un modèle de réseau à récurrence locale ne s'est pas révélé aussi parfait que nous l'espérions. Ce type d'architecture nécessite en effet de voir ses paramètres définis par l'intermédiaire d'une procédure d'apprentissage efficace apte à traiter le caractère temporel du signal à analyser. Or cette procédure n'existe pas pour l'instant. Tout au plus est-il possible de mettre en œuvre des procédures d'apprentissage approximant imparfaitement le corpus définissant le problème à résoudre. L'inexistence d'une procédure d'apprentissage issue du domaine d'étude des systèmes dynamiques non linéaires et notre incapacité à en définir une qui nous donne toute satisfaction ne nous a pas permis d'aller plus avant dans le développement de nos idées que nous allons exposer maintenant comme autant de suites possibles au travail entrepris lors de cette thèse.

8.2/ Développements de l'axe connexionniste

Nous entrevoyons quatre axes de recherche qu'il serait bon de suivre à l'issue des travaux que nous avons entrepris. Tous ces axes n'ont cependant pas la même importance, tant au niveau de la

portée théorique que du travail à entreprendre.

8.2.1/ Adaptation des coefficients de régression

Un premier développement possible de l'architecture que nous avons utilisée est la mise en place d'un système d'estimation orthogonale des poids tel que présenté dans [pican95] pour déterminer les coefficients de régression d'un réseau composé d'unités gamma. L'estimation orthogonale des poids est d'ailleurs intéressante à deux titres différents.

Il est tout d'abord possible de se demander si l'utilisation d'une telle procédure, capable de définir les valeurs des connexions synaptiques par emploi d'un réseau annexe, ou d'arrière plan par rapport au réseau principal, ne permettrait pas de tout simplement éliminer le problème de l'apprentissage qui existe du fait de la modification de l'architecture des neurones eux-même. Cette procédure emploie en effet un réseau statique comme base de définition des connexions du réseau principal et permettrait ainsi de supprimer la phase du calcul récursif et approximé des valeurs des gradients d'erreur spécifiques aux coefficients de régression.

Mais nous ne jugeons pas ce premier point comme étant le plus intéressant, quoique... L'estimation orthogonale des coefficients de régression nous semble en effet une très bonne méthode pour adapter ces coefficients par rapport au signal étudié. La dimension temporelle du modèle gamma ne dépend en effet que de ces coefficients de régression qui sont totalement isolés du reste des poids synaptiques du réseau. Il est donc envisageable de les adapter en cours d'utilisation en fonction de la modification de la vitesse du flux d'information. Le réseau gamma pourrait donc être fonctionnellement modifié selon le rythme d'élocution ou les différentes déformations de la voix telles que l'effet Lombard ou les modifications dues au stress, physique ou psychique, telles que la parole spontanée ou la parole criée.

8.2.2/ Définition d'une méthode d'apprentissage efficace

Le problème majeur que pose le modèle gamma par rapport aux perceptrons que nous avons utilisés dans la première partie de notre thèse reste la phase d'apprentissage qui a énormément de mal à positionner correctement tous les paramètres du réseau, dévaluant ainsi la qualité intrinsèque de l'architecture. Que sert en effet d'avoir une architecture connexionniste possédant des qualités reconnues théoriquement comme supérieures si aucune procédure d'apprentissage ne permet d'ajuster correctement les poids synaptiques d'un réseau respectant cette architecture ?

Nous avons vu, au cours de notre exposé, quels pouvaient être les liens entre les méthodes d'apprentissage connexionnistes existantes et les différentes études réalisées dans le domaine des systèmes dynamiques non linéaires. Ces études permettent de prendre, entre autres, conscience du phénomène de l'influence de l'échantillonnage du signal observé sur la phase de modélisation de celui-ci par toute méthode apte à la modélisation d'un système du même type que celui qui engendre le signal. Ainsi avons nous parlé au cours de notre thèse de la méthode développée par Takens pour la modélisation d'un système dynamique non linéaire et des notions que cette méthode met en place. Au rang de ces notions se trouve le pas de temps τ redéfinissant l'échantillonnage du signal observé à partir du taux d'échantillonnage original. Cette notion nous semble essentielle pour deux raisons. La première vient de la simple constatation de l'existence d'un courant de la recherche en reconnaissance automatique de la parole qui préconise d'utiliser des intervalles de taille supérieure à ceux couramment utilisés jusqu'à présent. La deuxième raison concerne le lien, que nous suspectons, entre ce coefficient de rééchantillonnage et le facteur du coefficient d'apprentissage que nous avons baptisé atténuateur de couple et qui permet de définir deux rythmes d'apprentissage dans notre modèle, un premier rythme pour les poids synaptiques habituels et un deuxième pour les coefficients de régression, dont la modification en cours d'apprentissage est effectuée plus lentement.

Nous avons mis en place cet atténuateur de couple à l'instar de quelques autres recherches, peu nombreuses, mettant en œuvre des architectures similaires à la nôtre. Notre volonté d'utiliser un taux

d'échantillonnage faible lors du calcul de nos vecteurs de prétraitement, 4 millisecondes plutôt que 10, a peut-être été à l'origine d'une utilisation de notre part d'atténuateurs de valeur plus faible que ceux auxquels nous pouvions nous référer même si tous les problèmes étudiés avec ce concept ne portaient pas uniquement sur la reconnaissance automatique de la parole. Notre tentative réussie de résoudre des tâches de reconnaissance et de classification de séquences abstraites a d'ailleurs été l'occasion pour nous d'utiliser des atténuateurs de couple avec des valeurs plus importantes que celles que nous avions employées en segmentation de la parole.

Cependant, la difficulté évidente de l'apprentissage dans les modèles récurrents, où il est parfois question d'apprentissage par divination, nous pousse à croire que même si un lien existe entre le taux d'échantillonnage τ de Takens et notre atténuateur de couple, il n'est pas le seul problème à prendre en compte. Nous pensons en effet que cet atténuateur de couple peut être vu comme une projection dans le domaine temporel du coefficient d'apprentissage présent dans toute procédure d'apprentissage connexionniste. Le coefficient d'apprentissage permet en effet de modifier les poids synaptiques de manière lente pour canaliser, en quelque sorte, l'exploration de l'espace des poids jusqu'à la découverte d'un minimum acceptable au regard de la tâche. L'abandon de ce coefficient provoque une modification trop importante et trop rapide des poids lors de la présentation des exemples du corpus, modification qui empêche tout apprentissage. C'est d'ailleurs dans la même optique d'atténuation des modifications de poids qu'a été mis en œuvre pour les perceptrons le principe de momentum qui permet de prendre en compte la dernière modification de poids effectuée pour limiter la modification courante lorsque celle-ci tente de modifier les poids de manière contraire au dernier ajustement.

Notre observation des premiers apprentissages avec un réseau gamma nous a donc conduit à mettre en place un coefficient supplémentaire d'apprentissage pour interdire les oscillations trop rapides des valeurs des coefficients de régression qui semblaient empêcher la convergence lors des phases d'apprentissage des paramètres du réseau. La modification des coefficients de régression modifiant la perception du corpus, les poids synaptiques ne pouvaient pas découvrir les invariants de ce corpus et, modifiant leurs valeurs, ils modifiaient la valeur de l'erreur de sortie, responsable de l'ajustement des coefficients de régression. La boucle était donc bouclée dans le processus responsable des oscillations tant des poids connexionnistes que des coefficients de régression. Nous n'avons cependant pas juger nécessaire d'implanter un mécanisme supplémentaire qui aurait été similaire à celui du momentum pour les coefficients de régression.

Il est cependant nécessaire de se demander si l'adjonction de coefficients supplémentaires dans l'apprentissage ne devrait pas être abandonné au profit d'une méthode plus lourde. Ainsi, ne faudrait-il pas essayer de déterminer les caractéristiques du corpus étudié par détermination d'une fréquence fondamentale moyenne dans chacun des canaux de données issus du prétraitement ? Cette méthode pourrait peut-être permettre de trouver une valeur approximée de t et donc de m dans la couche d'entrée. Mais cette méthode ne permettrait pas, nous semble-t-il, de tirer quelque conclusion que ce soit pour les unités à mémoire de la ou des couches cachées, la rendant ainsi en partie vaine.

Existe-t-il une justification de notre choix pour des atténuateurs de couple de très faible valeur permettant aux poids connexionnistes de s'ajuster correctement vis-à-vis des coefficients de régression ? Cette faible valeur limite-t-elle trop les modifications possibles des coefficients de régression pour que l'apprentissage puisse les modifier de manière correcte et suffisante sans être obligé de s'accommoder de paramètres de régression initialement en mauvaise adéquation avec le problème à résoudre ? Cet atténuateur ne serait-il pas encore trop fort du fait de la possibilité, encore trop importante à notre goût, de voir une phase d'apprentissage ne pas converger vers un état acceptable des poids vis-à-vis de la tâche à résoudre ? Faut-il enfin accepter de voir une phase d'apprentissage ne pas converger sans remettre en cause la procédure d'apprentissage elle-même, malgré toutes les heuristiques mises en œuvre ?

La dernière question à se poser consiste à savoir si la notion d'atténuateur de couple est un concept valide ou s'il s'agit d'une mauvaise heuristique permettant de résoudre le problème tout en cachant la vraie question qui permettra de résoudre, après découverte de la réponse, les problèmes d'apprentissage dans les réseaux connexionnistes d'architectures similaires à celle que nous avons définie. Et la question est, nous semble-t-il, grandement ouverte.

8.2.3/ Développement de modèles autorégressifs

La mise en place du modèle gamma tel qu'il est défini dans cette thèse ne constituait pour nous, à l'origine, qu'une étape vers des modèles plus complexes. La mauvaise qualité de la phase d'apprentissage dans les modèles dynamiques a été pour nous un frein qui nous a interdit d'aller plus avant puisque ce point paraît, à juste titre, essentiel.

Il est cependant possible d'imaginer des modèles d'architecture encore plus complexes que le modèle gamma. Notre choix du modèle gamma se justifie aux regards d'autres possibles par sa relative simplicité et par les connaissances théoriques disponibles à son égard. D'autres modèles possèdent cependant des caractéristiques qui semblent encore plus intéressantes.

Ainsi, le filtre de Laguerre nous semble-t-il au moins aussi intéressant que le filtre gamma [silva92] puisque des comportements similaires à ceux qu'il engendre ont déjà été observé dans le cerveau [brinker92]. Le filtre de Laguerre est d'une définition un peu plus complexe que le filtre gamma et nécessite donc une modification encore plus importante de la procédure d'apprentissage. Mais sa définition permet d'implanter une mémoire plus fine que celle du filtre gamma. Nous donnons l'architecture d'un tel filtre à la figure 8.1. De tels filtres pourraient bien évidemment être associés pour constituer une ligne de délais ou être positionnés en sortie de neurones standard, toutes ces possibilités ayant déjà été réalisées avec le filtre gamma.

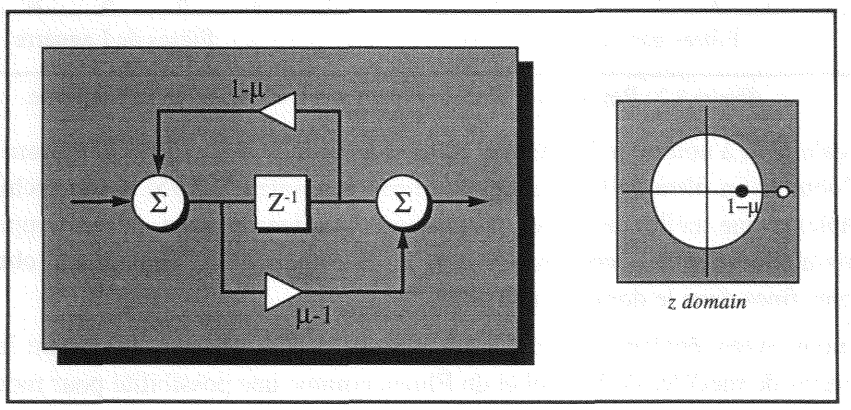


Figure 8.1 : Schéma d'un filtre de Laguerre.

Les caractéristiques du filtre de Laguerre sont plus intéressantes que celles du filtre gamma. La profondeur et la résolution sont équivalentes pour une ligne de K délais encastrés composée de filtres gamma ou de filtres de Laguerre partageant la même valeur de coefficient de régression, la résolution R étant toujours égale à μ alors que la profondeur D est encore égale à K/μ. La différence entre ces deux filtres se trouve en fait au niveau des réponses qui peuvent être obtenues. Ainsi, si les mécanismes d'excitation et d'inhibition retardée peuvent être obtenus par l'intermédiaire des filtres gamma avec l'aide des coefficients synaptiques positifs ou négatifs exploitant la sortie de ces filtres, le filtre de Laguerre est lui capable d'implanter seul une excitation suivie d'une inhibition, à la manière de ce qui peut être observé dans les connexions synaptiques véritables, sous l'appellation de *spike* (cf. chapitre 2, figure 2.10).

Une présentation des réponses comparées de ces deux filtres est présentée à la figure 8.2. La partie gauche de cette figure présente les réponses des cinq premiers filtres d'une ligne de délais encastrés de filtres gamma sur 14 pas de temps pour un coefficient μ de 0,7. Ce premier graphique est à rapproché du graphique c de la figure 6.2 du chapitre 6. La partie droite de la figure 8.2 présente, de

manière similaire, la réponse des cinq premiers filtres de Laguerre d'une ligne de délais encastrés sur 30 pas de temps.

Comme cela peut facilement se remarquer, la réponse d'un filtre de Laguerre se caractérise par une première montée de la réponse du filtre, suivie par une décroissance très forte, faisant passer la réponse du filtre du positif au négatif avant une dernière remontée de la réponse dans la partie positive du spectre, réponse qui est finalement amortie. Ce filtre permet donc d'obtenir un comportement similaire à un *spike* même si le graphe du *spike* présente une inversion de signe. La réponse obtenue est, de toute façon, bien plus complexe que celle qui peut être obtenue avec des filtres gamma.

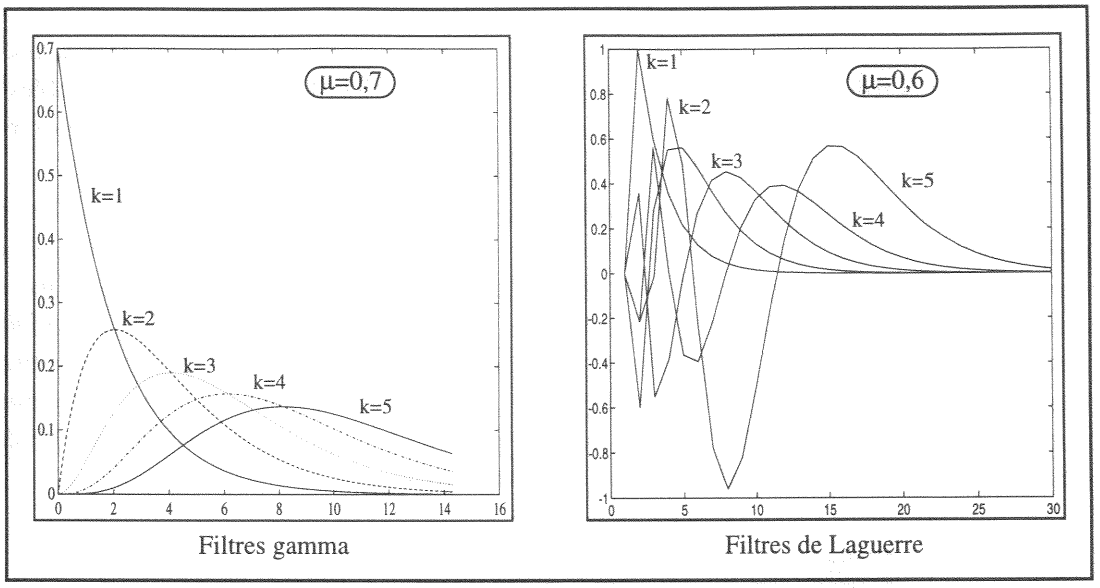


Figure 8.2 : Réponses type des filtres gamma et des filtres de Laguerre.

Il est également à noter que la création d'un filtre à partir de deux filtres gamma placés tête-bêche permet d'obtenir un filtre de la famille des filtres de Kautz [silva95]. Cette architecture est encore plus complexe que celle du filtre de Laguerre, obligeant ainsi à développer une procédure d'apprentissage encore plus complexe. Ce type de filtre permet cependant d'obtenir des réponses toujours plus fines dans le domaine temporel.

Enfin, nous avons également pensé à implanter un TDNN récurrent mêlant les caractéristiques architecturales de modèles de Waibel et de Elman comme une possibilité pour modéliser le bruit. Ce type de modélisation n'a, à notre connaissance, jamais été entreprise et nous pensons qu'il y aurait fort à gagner à mettre en œuvre un réseau apte à une telle modélisation. Notre sujet de thèse, portant sur la reconnaissance de petits vocabulaires en milieu bruité, ne nous a poussé qu'à trouver une architecture apte à la modélisation de la durée des phonèmes. Mais, au regard de certains résultats obtenus avec notre premier système, nous pensons qu'il serait bon de tester à grande échelle les qualités d'un système dont la tâche aurait été apprise dans certaines ambiances bruitées. Nous avons en effet pu constater que la qualité de l'apprentissage pouvait varier en fonctions des bruits d'apprentissage et que ces bruits permettaient d'obtenir des systèmes dont les capacités de généralisation sont plus ou moins bonnes vis-à-vis de bruits inconnus.

Ces constatations ouvrent peut-être la voie à une notion de qualité de bruit et de quantification de sa difficulté tout autant que de sa généricité. Ainsi, s'il semble évident que le bruit synthétique basé sur un spectre à long terme de la parole ou, pire encore, un bruit réel de parole impose de très fortes contraintes à un système de reconnaissance automatique de la parole, il est a priori beaucoup moins évident de juger de la gêne que pourra provoquer un bruit mécanique.

Il nous semble également intéressant d'étudier les capacités d'apprentissage de bruits dans

l'optique de généricité en tenant compte des études actuelles sur l'amélioration du signal de parole bruité à l'aide de mécanismes de soustraction dans les domaines fréquentiels, quéfrentiels (cf. chapitre 1, paragraphe 1.7.2.1) ou autres. Ce principe de soustraction d'un spectre, au mieux à "court" terme, qui se voit généraliser avec des techniques telles que la *Parallel Model Combination* nous semble tout à fait utilisable avec une base connexionniste. Notre exploration des modèles dynamiques nous a en effet permis de constater l'existence d'un domaine de modélisation de la musique, finalement assez connexe avec le domaine de la modélisation des séquences, que nous trouvons très intéressant pour la modélisation du bruit et son éventuelle soustraction. Un bruit non stationnaire ne peut, en effet, pas être modélisé aujourd'hui par manque d'une méthodologie générale, encore une fois en rapport avec les systèmes dynamiques non linéaires. Différents modèles connexionnistes nous semblent cependant aptes à devenir candidats pour une investigation de l'apprentissage et la généralisation de bruits, stationnaires et non stationnaires.

8.2.4/ Développement de modèles de bruits

La modélisation du bruit par réseau de neurones nous semble être du domaine d'un possible assez accessible. Plusieurs recherches permettent en effet d'envisager que de tels développement sont à portée de main, ou peut-être de bras. Les études faites sur la modélisation connexionniste de la musique nous semble ainsi très intéressantes vu à travers la lorgnette des recherches effectuées en reconnaissance automatique de la parole bruitée. Mais les modèles connexionnistes utilisés jusqu'à présent restent dans un respect quasi dogmatique des réseaux à récurrence par plaques. D'autres modèles connexionnistes, et d'autres modèles de reconnaissance des formes qui nous semblent en relation assez forte avec certains modèles connexionnistes, pourraient être appliqués pour résoudre ce type de tâches, pouvant nécessiter une modélisation d'automates sur plusieurs niveaux. Le modèle de la cascade de McClelland nous semble ainsi très intéressant conceptuellement même si son application effective n'a pas encore été réalisée. La machine multiniveau d'automates de Di Martino nous semble également intéressante dans les rapports qui pourraient être développés entre cette architecture et les réseaux connexionnistes. Nous avions nous même envisagé de réaliser un modèle de neurone à récurrence locale qui soit capable de modéliser, localement, un automate. Cette modélisation pourrait, par exemple, respecter l'architecture présentée à la figure 8.3. Un des membres du jury nous a, un jour, gentiment fait remarquer que cette architecture pourrait être, elle aussi, qualifiée de colonne corticale et cette remarque nous semble tout à fait justifiée dans le cas d'un réseau constitué de telles unités. Y aurait-il donc un intérêt à réinventer la roue ?

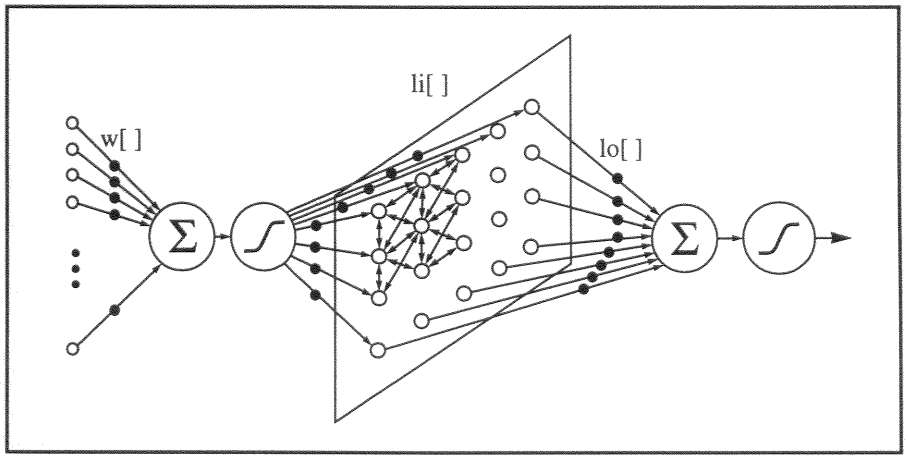


Figure 8.3 : Implantation d'un automate dans un neurone.

En outre, nous n'osons même plus, au terme de cette thèse, essayé d'imaginer ce que devrait être la procédure d'apprentissage dans ce modèle car il nous semble évident qu'une méthode basée sur la rétropropagation du gradient d'erreur serait incapable de découvrir un ensemble de poids acceptable vis-à-vis d'une tâche quelconque, le nombre de gradients devant être déterminés par récurrence

étant, ici, très grand.

Plutôt que de penser à une modélisation sous forme d'automate, il est également possible de penser à une modélisation basée sur un générateur d'impulsions. Ainsi, un système apte à générer des signaux sinusoïdaux à plusieurs fréquences et plusieurs phases pourrait servir de base à une méthode de caractérisation du signal ambiant. La mise en adéquation de ce système avec un bruit passerait alors par la définition de coefficients (synaptiques ?) effectuant une sorte de transformée de Fourier inverse pour minimiser la différence existant entre le signal de bruit et le générateur de signaux. Cette idée nous semble par ailleurs en adéquation avec la formule de Takens (cf chapitre 7, équation 7.3), la définition de τ permettant de définir une fréquence de base tandis que les harmoniques, et la complexité générale de la reconstruction, seraient définis par d. Quelques remarques existant dans la littérature sur les relations qu'entretiennent les réseaux récurrents et les systèmes générant des oscillations de manière induite nous laissent par ailleurs penser que le générateur de signaux peut être défini par apprentissage, les oscillations étant produites par évolution dans un système connexionniste dynamique (cf. chapitre 6, paragraphe 6.3.8)

Signalons enfin que les derniers développements des réseaux d'ondelettes, réseaux mariant les paradigmes du connexionnisme et de la théorie des ondelettes, semblent également très intéressant pour la reconnaissance de la parole dans le bruit. Les ondelettes se révèlent en effet être de très bonnes bases d'analyse du signal, permettant une représentation temps-fréquence plus exacte que celle obtenue par transformée de Fourier et permettant, par ailleurs, une reconstruction du signal presque parfaite, à l'inverse de la transformée de Fourier. Ce paradigme d'ondelettes a été appliqué, seul, au problème du débruitage de signaux avec de très bons résultats dans le cas de bruits stationnaires. Et le problème de la détermination correcte des coefficients d'ondelettes semble être assez facile à faire avec des réseaux connexionnistes, bien que nous n'ayons pas vraiment essayé et faisions totalement confiance à la littérature de ce domaine. Mais il s'agit là d'une autre thèse...

8.3/ Le mot de la fin

Nous espérons que la lecture a été enrichissante, que les fautes d'orthographe et de grammaire n'étaient pas trop nombreuses, que les références bibliographiques ne comportent plus trop de fautes et que les idées développées dans cette thèse sont intéressantes.

En tout cas bravo, la lecture est enfin finie, aux trois annexes près !

PARTIE 3

ANNEXES

Annexe 1 : Équations d'apprentissage

A1.1 Introduction

A1.1.1/ Présentation des équations

Cette annexe nous permet de présenter une partie des règles d'apprentissage que nous avons écrit au cours de cette thèse pour différents modèles connexionnistes. Nous ne présentons ici que les règles d'apprentissage pour les perceptrons multicouches standard et pour les réseaux gamma. D'autres réseaux, étudiés théoriquement, n'ont pas été implantés du fait des problèmes que nous a posé la procédure d'apprentissage gamma. Le modèle gamma étant le plus simple de tous les modèles envisagés, il nous a semblé que les mauvaises capacités d'apprentissage devaient être résolues avant tout autre développement architectural, une complexification de l'architecture d'un réseau n'étant pas synonyme de simplification de l'algorithme d'apprentissage.

La première méthode d'apprentissage que nous présenterons ici est la méthode d'apprentissage utilisée dans les perceptrons multicouches [lecun85]. Ceci permettra de facilement comparer la rétropropagation du gradient d'erreur une fois qu'elle est adaptée au modèle gamma.

L'algorithme d'apprentissage dans le modèle gamma comprend lui deux parties. La première partie de l'algorithme est une adaptation de la rétropropagation du gradient d'erreur adaptée à une architecture gamma. Cette première partie permet donc d'adapter les poids connexionnistes du réseau en fonction de la tâche à apprendre. La deuxième partie de l'algorithme, la plus importante et la plus délicate, permet d'adapter les coefficients de récurrence gamma. C'est cette partie de l'algorithme qui permet au réseau d'avoir une mémoire à plus ou moins long terme.

A1.1.2/ Notations

Les notations utilisées dans les paragraphes suivant respectent les notations standard du domaine connexionniste. Ainsi, w_{ij} représente la connexion synaptique existant entre d'un neurone j vers un neurone i. $net_{i,t}$ représente l'activation interne du neurone i au temps t, l'activation interne d'un neurone correspondant à la somme des activités des neurones afférents pondérées par les poids des connexions synaptiques relatives à ces afférences, avant toute modification par la fonction non linéaire. Cette valeur est rebaptisée $f(net_{i,t})$ une fois que la fonction non linéaire a été appliquée à $net_{i,t}$, f étant une fonction non linéaire de type sigmoïde. Cette valeur correspond donc à la sortie normale d'un perceptron multicouche. Dans le cas où le réseau considéré est un réseau multicouche, la valeur $f(net_{i,t})$ correspond à la sortie du neurone et peut être baptisée $y_{i,t}$. Cette dernière notation correspond à une valeur différente dans le cas où le neurone possède une unité gamma à la sortie de la fonction non linéaire. Dans ce cas, $y_{i,t}$ correspond à la valeur de sortie du filtre gamma qui a pris en compte la valeur de $f(net_{i,t})$.

Au sein de toutes ces équations, Δ représente le gradient d'erreur qui devra être rétropropagé pour permettre au réseau de converger vers un espace des poids permettant de résoudre le problème posé. $E(t)$ représente l'erreur globale en sortie à un instant t. Cette erreur est calculée à partir de $y_{i,t}$, valeur obtenue en sortie, et de $d_{i,t}$, valeur désirée et cible à atteindre par le processus d'apprentissage. Les valeurs d'entrée du réseau, correspondant le plus souvent à des valeurs issues d'un processus de prétraitement, correspondent à un vecteur. Nous avons donc baptisée $VE_{i,t}$ la valeur de la composante i du vecteur au temps t. Enfin, α représente le coefficient d'apprentissage.

Certaines des notations qui viennent d'être citées ont été replacées en contexte dans le schéma

d'un neurone gamma, figure A1.1. Ce schéma représente une unité en couche cachée et regroupe un neurone standard, tel qu'il peut en être trouvé en couche de sortie, et une unité gamma qui, utilisée isolément, permet de définir les unités de la couche d'entrée.

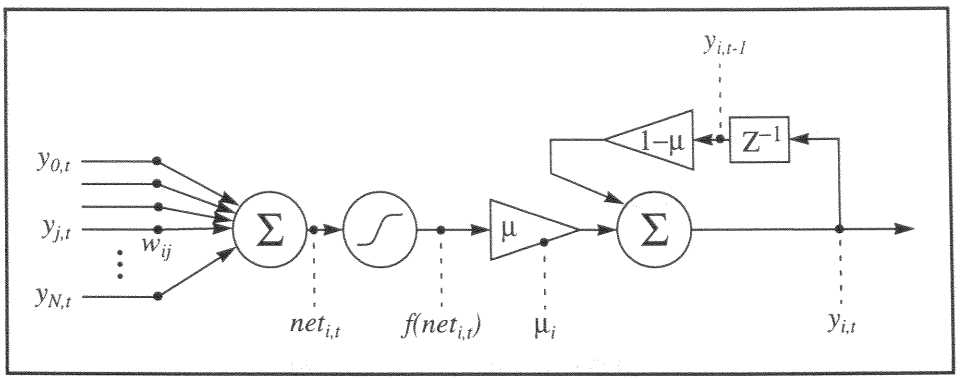


Figure A1.1 : Un neurone gamma et les différentes notations qui y sont associées.

A1.2 Mise à jour des poids connexionnistes

L'apprentissage des poids connexionnistes est la partie la plus importante de la définition d'un réseau connexionniste. C'est cet apprentissage qui permet de mettre en adéquation l'architecture du réseau avec la tâche qui lui est présentée. Cette procédure d'apprentissage ne permet pas d'obtenir, dans la grande majorité des cas, des poids identiques lorsque l'apprentissage est itéré. Les poids connexionnistes correspondent presque toujours à des approximations de l'espace des poids idéal. Seules les applications très limitées utilisant des architectures très simples peuvent voir différents apprentissages converger vers un même espace.

Cet apprentissage se fait lui même par parcours itératif du corpus d'apprentissage. Une forme est présentée en entrée du réseau puis propagée jusqu'à la couche de sortie. Une fois la valeur des unités de sortie calculée, il est possible de calculer l'erreur d'approximation effectuée par le réseau par comparaison aux valeurs de sortie désirées. Cette erreur est ensuite rétropropagée à partir de chaque unité de la couche de sortie par rétropropagation du gradient. Ce gradient va permettre de calculer la valeur de la modification à apporter à chacun des poids du réseau pour permettre à celui-ci de minimiser son erreur à la prochaine présentation de la forme.

Rappelons que la mise à jour des poids peut se faire en mode *online*, chaque fois qu'une erreur est calculée, ou en mode *batch*, à chaque fin de présentation de l'ensemble d'apprentissage. Dans ce dernier cas, l'erreur servant au calcul du gradient correspond à la somme de toutes les erreurs calculées sur le corpus.

A1.2.1/ Cas standard

Le cas que nous qualifions ici de standard est la procédure de rétropropagation du gradient d'erreur dans le cadre des perceptrons multicouches. Nous l'avons incluse ici à titre de rappel mais également pour faciliter la comparaison avec la méthode d'apprentissage dans les réseaux connexionnistes gamma.

A1.2.1.1/ Équation générale

L'erreur de sortie du réseau, pour une forme du corpus d'apprentissage, est calculée comme suit :

$$E(t) = \frac{1}{2} \sum_{i=1}^{\text{sorties(réseau)}} (y_{i,t} - d_{i,t})^2$$

Cette erreur est quadratique.

L'erreur globale, calculée sur l'ensemble du corpus d'apprentissage est, elle, définie comme suit,

en fonction du nombre N de formes dans le corpus et de la durée $t_{i,T}$ de chacune des formes dans le cas d'un corpus de formes temporelles (ou séquences) :

$$E = \sum_{i=1}^{N} \sum_{t=t_{i,0}}^{t_{i,T}} E(t)$$

La mise à jour des poids se fait à partir du gradient d'erreur qui est calculé comme suit, avec la pondération d'un coefficient d'apprentissage α, l'indice i représentant les neurones de la couche de sortie du réseau :

$$\Delta w_{i,j} = -\alpha \cdot \frac{\partial E(t)}{\partial w_{i,j}}$$

$$\Delta w_{i,j} = \alpha \cdot \left(-\frac{\partial E(t)}{\partial w_{i,j}} \right)$$

$$-\frac{\partial E(t)}{\partial w_{i,j}} = \left(-\frac{\partial E(t)}{\partial y_{i,t}} \right) \cdot \frac{\partial y_{i,t}}{\partial w_{i,j}}$$

Les deux parties de l'équation précédente peuvent se réécrire comme suit. Le premier terme de la partie droite de l'égalité correspond à l'erreur effectuée à la sortie d'un neurone. Ce coefficient d'erreur est baptisé delta d'erreur, δ_i.

$$-\frac{\partial E(t)}{\partial y_{i,t}} = \delta_i$$

Le deuxième terme de la partie droite de l'égalité correspond à la variation de la valeur d'activation d'un neurone en fonction de la variation des poids de pondération des neurones afférents. Ce terme se développe comme suit :

$$\frac{\partial y_{i,t}}{\partial w_{i,j}} = \frac{\partial \left(f \left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t} \right) \right)}{\partial w_{i,j}}$$

$$\frac{\partial y_{i,t}}{\partial w_{i,j}} = f' \left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t} \right) \cdot \frac{\partial \left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t} \right)}{\partial w_{i,j}}$$

$$\frac{\partial y_{i,t}}{\partial w_{i,j}} = f' \left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t} \right) \cdot y_{j,t}$$

$$\frac{\partial y_{i,t}}{\partial w_{i,j}} = f'(net_{i,t}) \cdot y_{j,t}$$

$$f'(net_{i,t}), = y'_{i,t}$$

$$\frac{\partial y_{i,t}}{\partial w_{i,j}} = y'_{i,t} \cdot y_{j,t}$$

On obtient de la réécriture de ces deux termes la valeur exacte de l'erreur :

$$-\frac{\partial E(t)}{\partial w_{i,j}} = \delta_i \cdot y'_{i,t} \cdot y_{j,t}$$

Cette valeur se calcule différemment en fonction de la position du neurone dans l'architecture. Deux cas doivent être considérés : soit le neurone est une des unités de la couche de sortie, soit le neurone appartient à une des couches cachée du réseau.

A1.2.1.2/ Équation pour un neurone de la couche de sortie

Lorsqu'un neurone appartient à la couche de sortie, l'erreur relative au neurone est simple à calculer puisqu'il s'agit de la différence entre la valeur obtenue et la valeur désirée :

$$\delta_i = y_{i,t} - d_{i,t}$$

Le gradient d'erreur se réécrit donc comme suit :

$$\Delta w_{i,j} = \alpha \cdot y'_{i,t} \cdot y_{j,t} \cdot (y_{i,t} - d_{i,t})$$

A1.2.1.3/ Équation pour un neurone en couche cachée

Dans le cas où le neurone est en couche cachée, la valeur de l'erreur relative à ce neurone doit être calculée en fonction de l'ensemble des erreurs effectuées par les neurones auxquels le neurone considéré est connecté en aval. L'idée du processus qui suit est de considérer que le neurone dont on veut calculer l'erreur est responsable de l'erreur d'une unité en aval respectivement par rapport à sa propre valeur d'activation mais également par rapport à la valeur de la connexion synaptique.

L'erreur des unités aval est donc rétropropagée en tenant compte des valeurs des poids de connexion.

$$\delta_i = -\frac{\partial E(t)}{\partial y_{i,t}} = \sum_{k=1}^{\text{sorties(i)}} \left(-\frac{\partial E(t)}{\partial y_{k,t}}\right) \cdot \frac{\partial y_{k,t}}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot \frac{\partial y_{k,t}}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot \frac{\partial \left(f\left(\sum_{l=1}^{\text{entrées(k)}, i \subset \{l\}} w_{k,l} \cdot y_{l,t} \right) \right)}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot f'\left(\sum_{l=1}^{\text{entrées(k)}} w_{k,l} \cdot y_{l,t}\right) \cdot \frac{\partial\left(\sum_{l=1}^{\text{entrées(k)},\, i \subset \{l\}} w_{k,l} \cdot y_{l,t}\right)}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot y'_{k,t} \cdot w_{k,i}$$

La valeur du gradient d'erreur se calcule donc comme suit :

$$\Delta w_{i,j} = \alpha \cdot y'_{i,t} \cdot y_{j,t} \cdot \left(\sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot y'_{k,t} \cdot w_{k,i}\right)$$

A1.2.2/ Cas gamma

A1.2.2.1/ Présentation

Les équations que nous présentons maintenant correspondent à l'algorithme d'apprentissage dans le modèle gamma. Le principe de calcul de la rétropropagation de l'erreur est ici le même que pour un perceptron multicouche. Mais l'architecture du neurone lui-même impose de totalement réécrire les équations pour que le mécanisme gamma placé après la fonction non linéaire soit pris en compte.

Trois cas différents se posent cependant. Nous avons premièrement décidé de ne pas mettre de mécanisme gamma à la sortie des neurones de la couche cachée. Ce choix a été fait car il nous a semblé qu'un mécanisme gamma placé à cet endroit allait à l'encontre du principe de la classification des formes. Les neurones de la couche de sortie sont des neurones d'architecture standard, identiques à ceux des perceptrons multicouches.

Le mécanisme gamma est par contre présent en couche cachée, comme cela a été présenté à la figure A1.1. Cette modification impose de profonds changements dans le calcul du gradient d'erreur puisqu'une fonction linéaire récursive est ajoutée après la fonction non linéaire.

Enfin, le troisième cas concerne la présence des seules unités gamma dans la couche d'entrée. Il n'y a plus alors de sommes pondérées en entrée d'une unité mais simplement un passage de l'impulsion d'une unité à l'unité suivante, selon le principe des lignes de délai.

Ces trois cas sont pris en compte dans les équations suivantes.

A1.2.2.2/ Équation générale

Ces équations sont équivalentes à celles données au paragraphe A1.2.1.1 mais tiennent compte de la nouvelle architecture des unités. Le gradient d'erreur se décompose comme précédemment.

$$\Delta w_{i,j} = -\alpha \frac{\partial E(t)}{\partial w_{i,j}}$$

$$\Delta w_{i,j} = -\alpha \cdot \frac{\partial E(t)}{\partial y_{i,t}} \cdot \frac{\partial y_{i,t}}{\partial w_{i,j}}$$

$$\Delta w_{i,j} = \alpha \cdot \left(-\frac{\partial E(t)}{\partial y_{i,t}}\right) \cdot \frac{\partial y_{i,t}}{\partial w_{i,j}}$$

$$\Delta w_{i,j} = \dot{\alpha} \cdot \delta_i \cdot \left(\frac{\partial \left((1-\mu_i) \cdot y_{i,t-1} + \mu_i \cdot f(net_{i,t}) \right)}{\partial w_{i,j}} \right)$$

$$\Delta w_{i,j} = \alpha \cdot \delta_i \cdot \left(\frac{\partial (1-\mu_i) \cdot y_{i,t-1}}{\partial w_{i,j}} + \frac{\partial (\mu_i \cdot f(net_{i,t}))}{\partial w_{i,j}} \right)$$

Le troisième terme de la partie droite de l'égalité peut être décomposé. La première partie de la somme peut être réécrite comme suit :

$$\frac{\partial (1-\mu_i) \cdot y_{i,t-1}}{\partial w_{i,j}} = \frac{\partial y_{i,t-1}}{\partial w_{i,j}} - \frac{\partial \mu_i \cdot y_{i,t-1}}{\partial w_{i,j}}$$

Ce qui permet de réécrire un premier terme :

$$\frac{\partial \mu_i \cdot y_{i,t-1}}{\partial w_{i,j}} = \mu_i \cdot \frac{\partial y_{i,t-1}}{\partial w_{i,j}} + \frac{\partial \mu_i}{\partial w_{i,j}} \cdot y_{i,t-1}$$

$$\frac{\partial \mu_i \cdot y_{i,t-1}}{\partial w_{i,j}} = \mu_i \cdot \frac{\partial y_{i,t-1}}{\partial w_{i,j}}$$

$$\frac{\partial (1-\mu_i) \cdot y_{i,t-1}}{\partial w_{i,j}} = (1-\mu_i) \cdot \frac{\partial y_{i,t-1}}{\partial w_{i,j}}$$

Le premier terme se calcule donc récursivement. Le deuxième terme se réécrit, lui, comme suit :

$$\frac{\partial (\mu_i \cdot f(net_{i,t}))}{\partial w_{i,j}} = \mu_i \cdot \frac{\partial (f(net_{i,t}))}{\partial w_{i,j}} + \frac{\partial \mu_i}{\partial w_{i,j}} \cdot f(net_{i,t})$$

$$\frac{\partial (\mu_i \cdot f(net_{i,t}))}{\partial w_{i,j}} = \mu_i \cdot \frac{\partial f(net_{i,t})}{\partial w_{i,j}}$$

$$\frac{\partial (\mu_i \cdot f(net_{i,t}))}{\partial w_{i,j}} = \mu_i \cdot f'(net_{i,t}) \cdot \frac{\partial net_{i,t}}{\partial w_{i,j}}$$

$$\frac{\partial (\mu_i \cdot f(net_{i,t}))}{\partial w_{i,j}} = \mu_i \cdot f'(net_{i,t}) \cdot y_{j,t}$$

Le calcul du gradient se fait donc selon l'équation suivante :

$$\Delta w_{i,j} = \alpha \cdot \delta_i \cdot \left((1-\mu_i) \cdot \frac{\partial y_{i,t-1}}{\partial w_{i,j}} + \mu_i \cdot f'(net_{i,t}) \cdot y_{j,t} \right)$$

A1.2.2.3/ Équation pour un neurone de la couche de sortie

Le delta des neurones en couche de sortie se calcule de la même manière que dans le cas du perceptron multicouche puisque qu'aucun mécanisme gamma n'est présent ici.

$$\delta_i = d_{i,t} - y_{i,t}$$

A1.2.2.4/ Équation pour un neurone en couche cachée

Le delta des neurones en couche cachée doit tenir compte de l'architecture du mécanisme gamma. La formule de calcul de delta s'écrit donc abstraitement de la même manière que dans le cas du perceptron multicouche :

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \left(-\frac{\partial E(t)}{\partial y_{k,t}} \right) \cdot \frac{\partial y_{k,t}}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot \frac{\partial y_{k,t}}{\partial y_{i,t}}$$

Mais la variation d'une unité par rapport à une de ses unités afférentes s'écrit de manière différente, comme suit :

$$\frac{\partial y_{k,t}}{\partial y_{i,t}} = \frac{\partial (1-\mu_k) \cdot y_{k,t-1}}{\partial y_{i,t}} + \frac{\partial \mu_k \cdot f(net_{k,t})}{\partial y_{i,t}}$$

Le premier terme de la partie droite se réécrit récursivement comme suit :

$$\frac{\partial (1-\mu_k) \cdot y_{k,t-1}}{\partial y_{i,t}} = (1-\mu_k) \cdot \frac{\partial y_{k,t-1}}{\partial y_{i,t}}$$

Tandis que le deuxième terme se réécrit comme suit :

$$\frac{\partial \mu_k \cdot f(net_{k,t})}{\partial w_{i,j}} = \mu_k \cdot f'(net_{k,t}) \cdot w_{k,i}$$

Ce qui permet de calculer la variation d'un neurone par rapport à une afférence selon la formule :

$$\frac{\partial y_{k,t}}{\partial y_{i,t}} = (1-\mu_k) \cdot \frac{\partial y_{k,t-1}}{\partial y_{i,t}} + \mu_k \cdot f'(net_{k,t}) \cdot w_{k,i}$$

Le delta des neurones en couche cachée peut donc s'écrire comme suit :

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot \left((1-\mu_k) \cdot \frac{\partial y_{k,t-1}}{\partial y_{i,t}} + \mu_k \cdot f'(net_{k,t}) \cdot w_{k,i} \right)$$

A1.3 Mise à jour des coefficients de récurrence

A1.3.1/ Présentation du problème

La mise à jour des poids connexionnistes n'est pas le seul processus d'apprentissage à prendre en compte dans un réseau constitué d'unités gamma. Les coefficients de récurrence, permettant de définir un rapport optimal entre résolution et profondeur, doivent également être déterminés en fonction de l'architecture du réseau et de la tâche à apprendre.

Il est bien sûr possible de fixer ces coefficients d'apprentissage à des valeurs constantes pendant

toute la phase d'apprentissage, en obligeant ainsi le réseau à utiliser un choix de compromis entre résolution et profondeur. Mais ce choix est difficile à réaliser. La tâche et l'architecture peuvent être assez complexes pour que l'utilisation d'un, de deux ou même de trois coefficients pour l'ensemble du réseau ne reflète que l'idée que le concepteur se fait de la tâche à apprendre. Cette idée peut, bien sûr, être totalement saugrenue. Il vaut mieux donc laisser au processus d'apprentissage le soin de régler ces paramètres.

Le processus d'apprentissage peut d'ailleurs être laissé libre de définir autant de coefficients de récurrence qu'il y a d'unités gamma, permettant ainsi une spécialisation poussée de chaque unité, chose qu'il est impossible de faire pour le concepteur. Ce type de choix crée cependant de nombreux degrés de liberté qu'il peut être difficile de contraindre si le réseau est surdimensionné par rapport à la tâche. À l'inverse, une obligation pour le processus d'apprentissage de définir une valeur partagée par l'ensemble des unités ou des neurones d'une couche peut pousser celui-ci à n'implanter aucune mémoire de manière à avoir une bonne résolution sur le signal en cours de traitement. Ce comportement est surtout vrai pour les couches cachées où nous avons plusieurs fois observé ce phénomène, les couches cachées n'implantant aucune mémoire alors qu'une augmentation du nombre de degrés de liberté se traduit par la création d'unités avec mémoire tandis que d'autres n'en implantent pas.

A1.3.2/ Équations

A1.3.2.1/ Coefficients de récurrence dans la plaque d'entrée

Le calcul du gradient d'erreur se fait, abstraitement, comme précédemment mais la variation n'est plus calculée par rapport au poids connexionniste $w_{i,j}$ comme nous l'avons fait dans tous le paragraphe A1.2. mais par rapport au coefficient de récurrence μ_i. C'est en effet ce coefficient qu'il s'agit désormais d'ajuster.

Le calcul du gradient en est le reflet :

$$\Delta\mu_i = -\alpha \cdot \frac{\partial E(t)}{\partial \mu_i}$$

$$\Delta\mu_i = -\alpha \cdot \frac{\partial E(t)}{\partial y_{i,t}} \cdot \frac{\partial y_{i,t}}{\partial \mu_i}$$

$$\Delta\mu_i = \alpha \cdot \delta_i \cdot \frac{\partial y_{i,t}}{\partial \mu_i}$$

A1.3.2.2/ Coefficients de récurrence de premier rang dans la plaque d'entrée

Le premier coefficient de récurrence d'une ligne de délai ne prend pas sa valeur d'entrée d'une unité de délai plus amont dans la ligne mais du vecteur d'entrée qui contient, le plus souvent, des valeurs issues d'un processus de prétraitement du signal d'entrée. $VE_{i,t}$ représente ici la $i^{\text{ème}}$ composante du vecteur d'entrée calculé au temps t.

Le calcul de la variation de l'activation du neurone par rapport au coefficient de récurrence se réécrit donc comme suit :

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \frac{\partial (\mu_i \cdot VE_{i,t} + (1-\mu_i) \cdot y_{i,t-1})}{\partial \mu_i}$$

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \frac{\partial (\mu_i \cdot VE_{i,t})}{\partial \mu_i} + \frac{\partial y_{i,t-1}}{\partial \mu_i} - \frac{\partial (\mu_i \cdot y_{i,t-1})}{\partial \mu_i}$$

$$\frac{\partial(\mu_i \cdot VE_{i,t})}{\partial \mu_i} = \mu_i \cdot \frac{\partial VE_{i,t}}{\partial \mu_i} + VE_{i,t}$$

$$\frac{\partial(\mu_i \cdot VE_{i,t})}{\partial \mu_i} = VE_{i,t}$$

$$\frac{\partial(\mu_i \cdot y_{i,t-1})}{\partial \mu_i} = \mu_i \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} + y_{i,t-1}$$

$$\frac{\partial y_{i,t}}{\partial \mu_i} = VE_{i,t} + (1-\mu_i) \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} - y_{i,t-1}$$

Ce qui permet d'obtenir un gradient qui se calcule selon la formule suivante :

$$\Delta\mu_i = \alpha \cdot \delta_i \cdot \left(VE_{i,t} + (1-\mu_i) \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} - y_{i,t-1} \right)$$

Nous n'avons pas développé le mode de calcul du delta car les unités de la couche supérieure peuvent être des neurones d'architecture standard tout autant que des neurones gamma :

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \left(-\frac{\partial E(t)}{\partial y_{k,t}} \right) \cdot \frac{\partial y_{k,t}}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot \text{(partie dépendante de l'architecture)}$$

A1.3.2.3/ Coefficients de récurrence de rang suivant dans la plaque d'entrée

Les unités de délai qui ne sont pas en tête de ligne ne prennent pas leurs valeurs d'entrée dans un vecteur issu d'un prétraitement mais de l'unité directement en amont dans la ligne. La variation de l'activité d'une unité par rapport à la variation du coefficient d'apprentissage s'écrit donc comme de la manière suivante :

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \frac{\partial(\mu_i \cdot y_{i-1,t-1} + (1-\mu_i) \cdot y_{i,t-1})}{\partial \mu_i}$$

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \frac{\partial(\mu_i \cdot y_{i-1,t-1})}{\partial \mu_i} + \frac{\partial y_{i,t-1}}{\partial \mu_i} - \frac{\partial(\mu_i \cdot y_{i,t-1})}{\partial \mu_i}$$

Le premier terme de la partie droite se réécrit :

$$\frac{\partial(\mu_i \cdot y_{i-1,t-1})}{\partial \mu_i} = \mu_i \cdot \frac{\partial y_{i-1,t-1}}{\partial \mu_i} + y_{i-1,t-1}$$

Le deuxième terme de la partie droite de l'égalité se réécrit en :

$$\frac{\partial(\mu_i \cdot y_{i,t-1})}{\partial \mu_i} = \mu_i \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} + y_{i,t-1}$$

La dérivée partielle que nous cherchons è calculer se réécrit donc de la manière suivante :

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \mu_i \cdot \frac{\partial y_{i-1,t-1}}{\partial \mu_i} + (1-\mu_i) \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} + y_{i-1,t-1} - y_{i,t-1}$$

Ce qui nous conduit à calculer le gradient selon la formule :

$$\Delta \mu_i = \alpha \cdot \delta_i \cdot \left(\mu_i \cdot \frac{\partial y_{i-1,t-1}}{\partial \mu_i} + (1-\mu_i) \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} + y_{i-1,t-1} - y_{i,t-1} \right)$$

Comme précédemment, le calcul du delta n'est pas développé :

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \left(-\frac{\partial E(t)}{\partial y_{k,t}} \right) \cdot \frac{\partial y_{k,t}}{\partial y_{i,t}}$$

$$\delta_i = \sum_{k=1}^{\text{sorties(i)}} \delta_k \cdot \text{(partie dépendante de l'architecture)}$$

A1.3.2.4/ Cas gamma en couches supérieures

Le deuxième cas pouvant se produire dans le calcul des gradients d'erreur sur les coefficients de récurrence est le cas où le mécanisme gamma se trouve en sortie d'un neurone de la couche cachée, après la fonction non linéaire. Le gradient d'erreur s'écrit toujours de la même manière à un niveau abstrait :

$$\Delta \mu_i = -\alpha \cdot \frac{\partial E(t)}{\partial \mu_i}$$

$$\Delta \mu_i = -\alpha \cdot \frac{\partial E(t)}{\partial y_{i,t}} \cdot \frac{\partial y_{i,t}}{\partial \mu_i}$$

$$\Delta \mu_i = \alpha \cdot \delta_i \cdot \frac{\partial y_{i,t}}{\partial \mu_i}$$

Le développement de la partie de cette équation concernant la variation de l'activation d'une unité par rapport à la variation du coefficient de récurrence s'écrit de la manière suivante, lorsque l'architecture de l'unité gamma est prise en compte :

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \frac{\partial \left[(1-\mu_i) \cdot y_{i,t-1} + \mu_i \cdot f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t} \right) \right]}{\partial \mu_i}$$

$$\frac{\partial y_{i,t}}{\partial \mu_i} = \frac{\partial \left((1-\mu_i) \cdot y_{i,t-1} \right)}{\partial \mu_i} + \frac{\partial \left(\mu_i \cdot f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t} \right) \right)}{\partial \mu_i}$$

Le premier terme de la partie droite de l'égalité se réécrit selon :

$$\frac{\partial\left(\left(1-\mu_i\right)\cdot y_{i,\,t-1}\right)}{\partial\mu_i} = \frac{\partial\left(1-\mu_i\right)}{\partial\mu_i}\cdot y_{i,\,t-1} + \left(1-\mu_i\right)\cdot\frac{\partial y_{i,\,t-1}}{\partial\mu_i}$$

$$\frac{\partial\left(1-\mu_i\right)}{\partial\mu_i}\cdot y_{i,\,t-1} = \frac{\partial 1}{\partial\mu_i}\cdot y_{i,\,t-1} + \frac{\partial\left(-\mu_i\right)}{\partial\mu_i}\cdot y_{i,\,t-1}$$

$$\frac{\partial\left(1-\mu_i\right)}{\partial\mu_i}\cdot y_{i,\,t-1} = -y_{i,\,t-1}$$

Tandis que le deuxième terme se réécrit de la manière suivante :

$$\frac{\partial\left(\mu_i\cdot f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)\right)}{\partial\mu_i} = \frac{\partial\mu_i}{\partial\mu_i}\cdot f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right) + \mu_i\cdot\frac{\partial f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i}$$

Le premier terme du deuxième terme se réécrit :

$$\frac{\partial\mu_i}{\partial\mu_i}\cdot f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right) = f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)$$

Tandis que le deuxième terme du deuxième terme se réécrit :

$$\mu_i\cdot\frac{\partial f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i} = \mu_i\cdot f'\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)\cdot\frac{\partial\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i}$$

Avec :

$$\frac{\partial\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i} = \sum_{j=1}^{\text{entrées(i)}}\frac{\partial\left(w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i}$$

$$\frac{\partial\left(w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i} = w_{i,j}\cdot\frac{\partial y_{j,\,t}}{\partial\mu_i} + \frac{\partial w_{i,j}}{\partial\mu_i}\cdot y_{j,\,t}$$

$$\frac{\partial\left(w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i} = w_{i,j}\cdot\frac{\partial y_{j,\,t}}{\partial\mu_i}$$

$$\frac{\partial\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot y_{j,\,t}\right)}{\partial\mu_i} = \sum_{j=1}^{\text{entrées(i)}} w_{i,j}\cdot\frac{\partial y_{j,\,t}}{\partial\mu_i}$$

$$\frac{\partial y_{j,t}}{\partial \mu_i} = 0'$$

$$\frac{\partial\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t}\right)}{\partial \mu_i} = 0$$

$$\mu_i \cdot \frac{\partial f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t}\right)}{\partial \mu_i} = 0$$

Ceci permet d'obtenir la formule suivante :

$$\frac{\partial y_{i,t}}{\partial \mu_i} = -y_{i,t-1} + (1-\mu) \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} + f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t}\right)$$

Le gradient doit donc être calculé selon la formule suivante :

$$\Delta \mu_i = \alpha \cdot \delta_i \cdot \left(-y_{i,t-1} + (1-\mu) \cdot \frac{\partial y_{i,t-1}}{\partial \mu_i} + f\left(\sum_{j=1}^{\text{entrées(i)}} w_{i,j} \cdot y_{j,t}\right)\right)$$

A1.4 Types d'apprentissage

L'algorithme d'apprentissage que nous venons de présenter n'est pas le seul possible. Deux grands paradigmes s'affrontent à l'heure actuelle pour l'apprentissage dans les réseaux récurrents. Il semble que, bien qu'aucun des deux ne soit parfaitement efficace, l'algorithme que nous venons de présenter est trop approximatif.

A1.4.1/ Apprentissage récurrent temps-réel

L'apprentissage présenté dans les paragraphes précédents, paragraphe A1.2 pour les poids connexionnistes et paragraphe A1.3 pour les coefficients de récurrence, est un apprentissage d'un type identique à celui présenté dans [williams89a], à la suite des travaux présentés dans [almeida87] et [pineda87] et faisant suite à une première ébauche présentée dans [robinson89]. Il s'agit d'un apprentissage récurrent temps réel qui est, avant tout, fait pour un apprentissage *online* de la tâche. Chaque présentation d'une nouvelle forme en entrée

Bien sûr, cette constance de la mise à jour des poids peut être relâchée et les poids peuvent être mis à jour à des intervalles de temps plus longs que les intervalles élémentaires [catfolis93]. Mais cette technique, malgré son bien fondé et toute la théorie qui peut y être associée [takens81], ne permet pas un apprentissage optimal des poids dans le cas de notre réseau, et d'autres...

A1.4.2/ Rétropropagation dans le temps

La rétropropagation dans le temps correspond à une vision d'un apprentissage dans les réseaux récurrents qui se ferait en mode *batch*. L'apprentissage en temps réel est donc dual par rapport à la rétropropagation dans le temps [beaufays94]. En fait, la rétropropagation dans le temps a comme avantage principal de ne considérer l'erreur qu'à la fin de la présentation d'une forme temporelle qui est donc considérée comme une véritable séquence et non plus comme une simple suite de formes dont la classification est identique. Cette méthode a tout d'abord été présentée dans [werbos90].

L'apprentissage se fait en gardant la trace des activations internes du réseau tant que la fin de la séquence n'èst pas atteinte. Une fois la fin de la séquence atteinte, la valeur de sortie désirée est présentée et les gradients d'erreurs sont rétropropagés à travers tout le réseau non seulement dans le plan architectural mais également dans le plan temporel. Le gros avantage de cette méthode est de travailler constamment avec des gradients donc le calcul se fait à partir des valeurs exactes. Ces gradients ne sont donc plus approximés comme ils peuvent l'être dans l'apprentissage en temps réel. Le désavantage de cette méthode est, bien évidemment, qu'aucune sortie désirée n'est présentée pendant toute la durée de présentation de la séquence, du premier à l'avant-dernier élément. Les poids allant de la dernière couche cachée à la couche de sortie sont donc sous spécifiés puisqu'ils ne sont pas mis à jour avec la même fréquence que les poids connexionnistes des couches internes et les coefficients de récurrence.

L'algorithme d'apprentissage et, surtout, l'implantation informatique de cet algorithme, impose de revoir quelque peu les équations que nous avons donné précédemment. En effet, le gradient n'est plus calculé à partir des gradients des instants précédents mais, la rétropropagation dans le temps étant ce qu'elle est, ces gradients sont calculés à partir des gradients des couches supérieures et des gradients des couches futures.

Schématiquement, cela peut être résumé en considérant deux fois la même unité, un neurone gamma, à deux pas de temps successifs. L'algorithme de mise à jour des poids suit le principe exposé (figure A1.2).

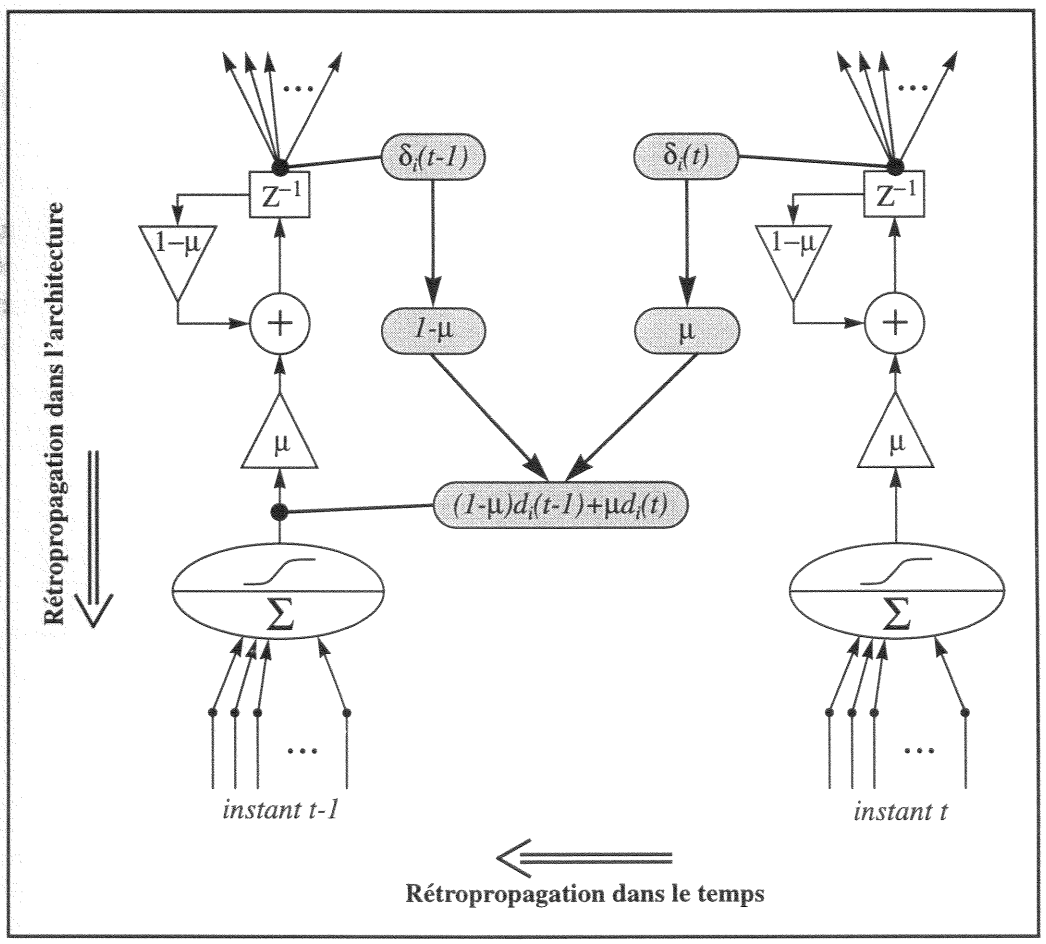


Figure A1.2 : Schéma de principe de la rétropropagation dans le temps pour des neurones gamma.

L'algorithme découle très simplement de ce schéma de principe.

Annexe 2 : Réponses des filtres gamma

A2.1 Présentation

Les graphiques présentés dans cette annexe permettront au lecteur de se représenter le comportement d'un filtre gamma (paragraphe A2.2) tout autant que d'une ligne de délais encastrés (*tapped delay line*) constituée d'unités gamma (paragraphe A2.3), les filtres gamma étant des intégrateurs à déperdition (ou intégrateurs à perte, *leaky integrator*).

A2.2 Réponses des filtres

Cette première partie de l'annexe se veut être une petite extension du paragraphe 6.1.3 du chapitre 6. Nous avions alors brièvement présenté les réponses de divers filtres, dont le filtre à décroissance exponentielle et le filtre gamma.

Rappelons tout d'abord qu'un filtre gamma placé en tête dans une ligne de délais se comporte à la manière d'un filtre à décroissance exponentielle bien que la réponse exacte obtenue soit différente du fait des définitions dissemblables de ces deux filtres. Cette similarité s'explique facilement par le fait que le filtre en tête de ligne ne peut pas, après le passage d'une impulsion, être alimenté par la trace résiduelle de cette impulsion qui serait fournie par des filtres se trouvant plus en amont dans la ligne de délais.

L'équation du filtre gamma est rappelée dans l'équation A2.1. Le filtre ainsi défini calcule une partie de son activité en prenant en compte l'activité lui arrivant de l'extérieur, cette activité étant pondérée d'un coefficient, μ. L'autre partie de l'activité est déterminée par la récurrence interne du filtre qui pondère la valeur par le complément à 1 du facteur de pondération portant sur l'activité externe. Cette activité interne est donc pondérée par 1 - μ. La somme de ces deux valeurs définit la nouvelle intensité d'activation du filtre.

$$y_{i,\,t+1} = (\mu \times y_{i-1,\,t}) + ((1-\mu) \times y_{i,\,t}) \qquad \text{(Éq. A2.1)}$$

L'avantage de cette définition réside dans la présence d'un seul coefficient, μ, qui permet de définir l'ensemble du comportement du filtre. La valeur de ce coefficient est, de plus, restreinte à l'intervalle]0,2[, cette restriction assurant la stabilité du filtre. Cette intervalle peut être décomposé en trois sous intervalles, chacun correspondant à un cas particulier de comportement du filtre. Le filtre est passe-bas pour des valeurs de μ comprises entre 0 et 1, bornes exclues. Ce même filtre sera passe-haut pour des valeurs de μ comprises entre 1 et 2, bornes exclues. Le cas particulier où μ est égal à 1, enfin, correspond au cas où le filtre agit comme une simple unité de mémorisation.

En restreignant notre étude à l'intervalle]0,1], il est possible de dégager deux notions supplémentaires qui ont déjà été exposées au chapitre 7, paragraphe 7.2.2.3 : la profondeur et la résolution. Ces notions sont antagonistes au sens où une bonne profondeur interdit d'avoir une bonne résolution et vice versa. La profondeur sera bonne lorsque la valeur de μ sera proche de 0. Dans ce cas, la trace de l'impulsion d'entrée est conservée sur un long intervalle de temps grâce à la récurrence. La trace initiale de cette impulsion est cependant dégradée par le coefficient μ d'entrée du filtre, dégradant d'autant la résolution, synonyme de qualité de conservation de l'impulsion. À l'inverse, lorsque μ est proche de 1, l'impulsion ne sera quasiment pas modifiée mais sa trace sera très faible en valeur relativement à l'impulsion initiale et sera, de plus, de très courte durée. La résolution peut alors être qualifiée de bonne et la profondeur de mauvaise.

Les figures présentées dans les paragraphes suivants ont toutes été calculées de la même manière :

une impulsion initiale égale à 1 a été fournie au premier délai de la ligne qui n'a ensuite plus reçu que des impulsions à 0. Les figures présentées permettent d'étudier l'évolution de la réponse des filtres placés, d'une part, en tête de ligne (graphiques de gauche dans les figures) et, d'autre part, au 6ème rang de la ligne (graphiques de droite dans les figures).

A2.2.1/ Filtre gamma passe-bas

Les cas où le filtre gamma se comporte comme un filtre passe-bas est le cas qui nous a tout particulièrement intéressé lors de notre thèse. Le filtre gamma permet alors d'implanter une mémoire à décroissance plus ou moins rapide en fonction de la valeur de μ. C'est cette capacité qui nous a poussé à utiliser le filtre gamma pour simuler les phénomènes d'excitation et d'inhibition retardées.

Un filtre gamma sera passe-bas lorsque la valeur de μ sera comprise entre 0 et 1, bornes exclues. Au sein de cet intervalle peuvent être isolés plusieurs sous intervalles permettant une mémorisation plus ou moins forte.

Un premier sous intervalle peut ainsi être défini entre 0 et 0,3. Le filtre gamma présente alors des réponses très faibles pendant de très longues durées, comme le montre la figure A2.1 et la figure A2.2. Dans ces deux cas, le filtre en tête de ligne perd son activité de manière relativement lente et les filtres plus en aval dans la ligne de délais, qui répercutent les traces amont du signal, voient leurs activités, faibles en intensité, perdurées sur des intervalles de temps de plus en plus conséquents.

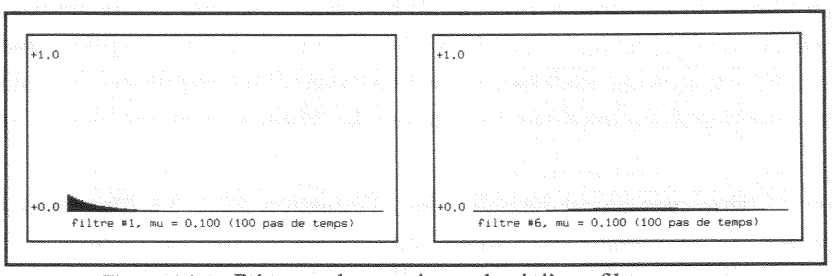


Figure A2.1 : Réponse du premier et du sixième filtre gamma d'une ligne de délais encastrés pour $\mu = 0,1$

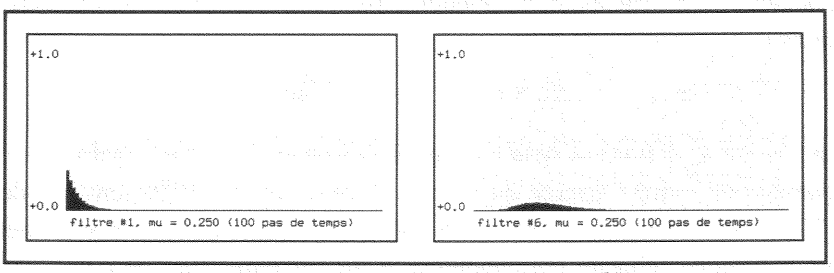


Figure A2.2 : Réponse du premier et du sixième filtre gamma d'une ligne de délais encastrés pour $\mu = 0,25$

Un deuxième sous intervalle peut être isolé entre 0,3 et 0,6. Dans ce cas, la réponse des filtres se trouvant dans la ligne de délais présente une forme de bosse de montée et de descente assez fortes. Dans ce cas, la valeur de l'unité en tête de ligne décroît plus rapidement que précédemment mais l'intensité de la trace est plus forte. Des remarques similaires peuvent être faites sur les délais en aval de la tête de ligne.

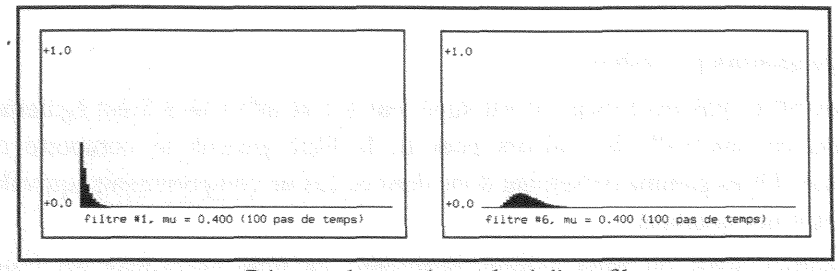


Figure A2.3 : Réponse du premier et du sixième filtre gamma
d'une ligne de délais encastrés pour $\mu = 0,4$

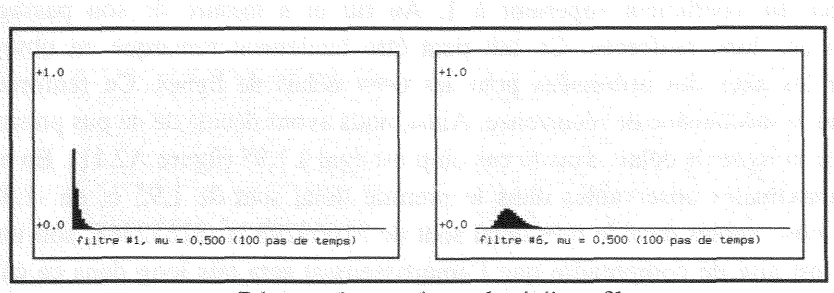


Figure A2.4 : Réponse du premier et du sixième filtre gamma
d'une ligne de délais encastrés pour $\mu = 0,5$

Le dernier sous intervalle, pour des valeurs de μ comprises entre 0,6 et 1 exclu, permet d'obtenir des réponses de forte intensité dont la décroissance est très rapide. Ce phénomène est bien visible sur la figure A2.5, la trace du signal dans le sixième délai évoluant à la manière de la trace de signal dans le premier délai de la ligne de la figure A2.3.

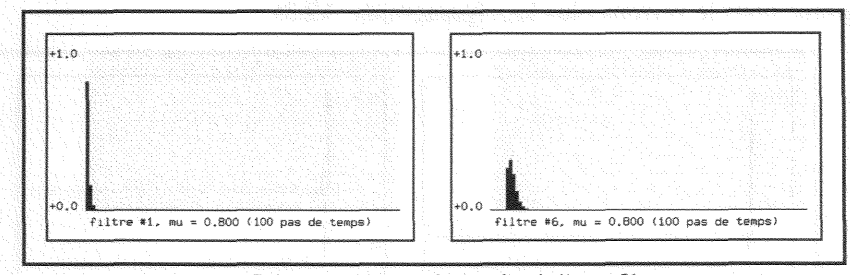


Figure A2.5 : Réponse du premier et du sixième filtre gamma
d'une ligne de délais encastrés pour $\mu = 0,8$

A2.2.2/ Filtre gamma équivalent à un délai simple

Dans le cas où μ est égal à 1, le filtre gamma se comporte comme un simple délai, tel qu'il est possible d'en trouver dans un TDNN par exemple [waibel89]. Le filtre n'effectue alors plus aucune mémorisation à long terme, se contentant de stocker l'impulsion telle qu'elle lui est fournie par le délai de rang immédiatement inférieur à un certain pas de temps pour la redonner, telle qu'elle a été reçue, au délai suivant lors du pas de temps suivant.

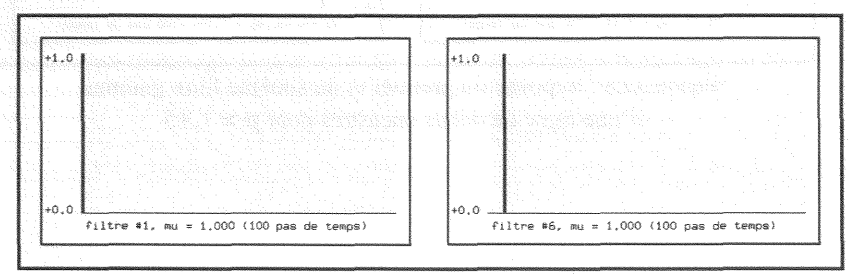


Figure A2.6 : Réponse du premier et du sixième filtre gamma
d'une ligne de délais encastrés pour $\mu = 1,0$

L'impulsion initiale n'étant absolument pas modifiée, la résolution est dans ce cas maximum et la

profondeur nulle.

A2.2.3/ Filtre gamma passe-haut

Le cas des filtres gamma lorsque μ est supérieur à 1 et inférieur à 2 est également intéressant à étudier. Dans cet intervalle de valeurs pour μ, le filtre gamma se comporte comme un filtre passe-haut. Les filtres gamma présentent donc dans ce cas un comportement équivalent à un système oscillant avec amortissement.

Un des particularités du filtre gamma positionné en filtre passe-haut est l'ampleur que peut prendre une impulsion. En effet, à chaque passage de l'impulsion initiale dans un délai, celle-ci est pondérée par un coefficient supérieur à 1. Au fur et à mesure de son passage dans la ligne, l'impulsion est donc renforcée. Ce fait peut être facilement remarqué en observant les valeurs portées sur les axes des ordonnées pour les 6èmes délais de lignes. Ce renforcement est encore accentué par le mécanisme de récurrence. Ainsi, nous avons décidé de ne pas présenter la réponse de 6ème délai de la ligne de délais dans le cas où μ est égal à 1,95 (figure A2.11). En effet, alors que les intensités maximales observables dans le premier délai sont de 1,95 et de -1,852, les intensités maximales observables dans le 6ème délai sont de 31669298 et -31652488, soit un écart de plus de $6{,}3\times10^7$. Il est aisé de comprendre que l'amortissement sera très long dans ce cas et que de telles valeurs sont à proscrire dans le cadre d'une implantation connexionniste.

Comme dans les cas où le filtre gamma est passe-bas, il est ici aussi possible de matérialiser plusieurs sous intervalles. Il est ainsi possible d'en définir deux en fonction de la rapidité d'amortissement qui est fonction de la valeur de μ. Le premier intervalle, correspondant à un amortissement rapide, pourrait être défini entre 1 et 1,2 alors que le deuxième pourrait être défini entre 1,2 et 2. Cette rapidité d'amortissement est fonction de μ mais également du rang du délai dans la ligne, comme nous le verrons plus loin (paragraphe A2.3).

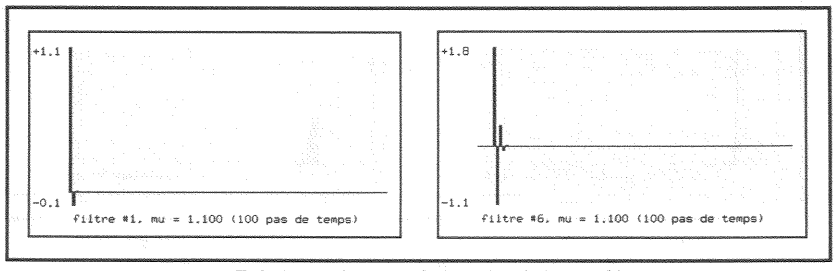


Figure A2.7 : Réponse du premier et du sixième filtre gamma d'une ligne de délais encastrés pour μ = 1,1

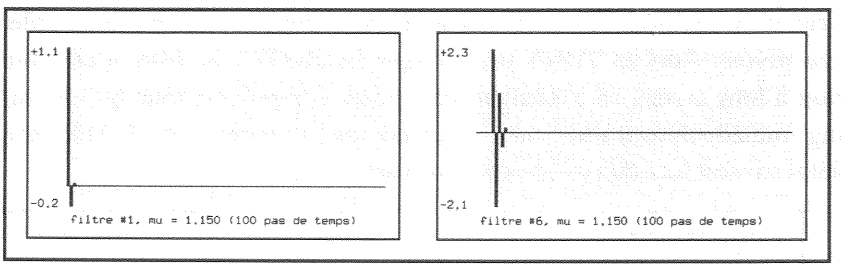


Figure A2.8 : Réponse du premier et du sixième filtre gamma d'une ligne de délais encastrés pour μ = 1,15

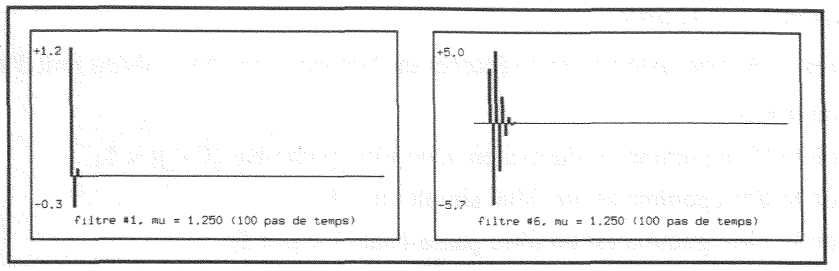


Figure A2.9 : Réponse du premier et du sixième filtre gamma d'une ligne de délais encastrés pour $\mu = 1{,}25$

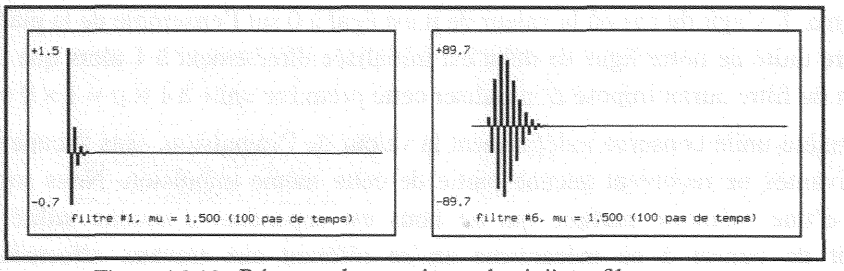


Figure A2.10 : Réponse du premier et du sixième filtre gamma d'une ligne de délais encastrés pour $\mu = 1{,}5$

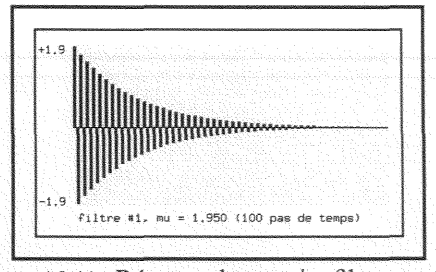


Figure A2.11 : Réponse du premier filtre gamma d'une ligne de délais encastrés pour $\mu = 1{,}95$

A2.3 Réponses des lignes de délais

Toutes les figures présentées dans ce paragraphe résultent d'un calcul identique : la ligne de délais est constituée de 200 unités gamma partageant le même facteur de gain. L'impulsion initiale est égale à 1 et les impulsions suivantes sont égales à 0. Les figures permettent d'observer la conservation de l'impulsion initiale sur les 400 premiers pas de temps.

Il faut cependant noter que la première unité gamma de la ligne de délais est une unité partielle : si la valeur mémorisée en interne est bien pondérée à chaque itération par $(1 - \mu)$, l'unité à été initialisée à 1 (l'impulsion initiale) et non à $(1 \times \mu)$ comme cela aurait dû être le cas (équation A2.1). Hormis la première unité de la ligne de délais, toutes les autres unités respectent l'équation standard définissant un filtre gamma telle que présentée dans l'équation A2.1.

Cette initialisation à 1 de la première unité a cependant été choisie pour l'ensemble des figures de manière à présenter du mieux possible la déperdition de l'impulsion lors de son transfert sur la ligne de délais. Le résultat est intéressant à observer bien qu'il ne puisse pas avoir lieu dans le cadre d'une utilisation normale. Cet artifice permet également de présenter clairement le cas limite où $\mu = 0$, chose qui n'aurait pas été possible si l'équation gamma avait été respectée dès le premier pas de temps.

Chaque figure doit être interprétée comme suit : l'impulsion initiale est visible en haut à gauche du graphique et sa trace, plus ou moins forte, est représentée par la partie du graphique où les tons de gris sont plus foncés. La valeur 0 correspond au fond du graphique qui a été grisé dans un souci de facilité de lecture. Le lecteur peut se reporter aux niveaux de gris à gauche de chaque graphique pour

juger de l'intensité de la trace.

Les figures ci-dessous peuvent être séparées en 4 groupes que nous allons tout d'abord énumérés :

- cas où $\mu = 0$,
- cas où le filtre gamma est équivalent à un filtre passe-bas ($0 < \mu < 1$),
- cas où le filtre gamma est un délai simple ($\mu = 1$),
- cas où le filtre gamma est un filtre passe-haut ($1 < \mu < 2$).

A2.3.1/ Cas hors limites

Le premier cas que nous présentons ici n'a, en fait, pas lieu d'être dans une utilisation normale du filtre gamma. Il s'agit du cas où la valeur de μ est égal à 0 sur l'ensemble de la plaque de délais et où la première unité de notre ligne de délai est initialisée directement à 1 alors que le respect strict de l'équation du filtre aurait imposé d'initialiser cette première unité à $1 \times \mu = 1 \times 0 = 0$.

La première unité conserve indéfiniment la valeur de l'impulsion, sans aucune déperdition, et les unités suivantes ne reçoivent aucune partie de cette même impulsion. Nous sommes donc ici en présence d'une mémoire parfaite qui ne nous est cependant d'aucune utilité. Il est cependant intéressant de penser à ce mécanisme en se référant aux travaux effectués sur l'estimation orthogonale des poids [pican95] ou aux travaux sur la triade synaptique [changeux96] puisque ce filtre pourrait servir de rétenseur d'activité dans des tâches nécessitant des mémorisations à long et très long terme.

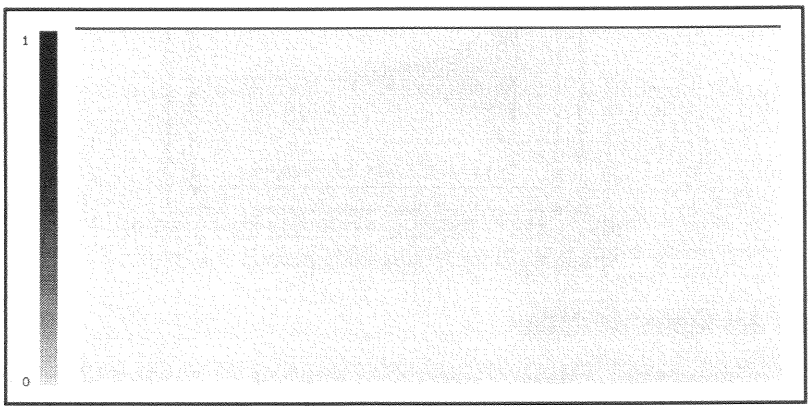


Figure A2.12 : Réponses des filtres gamma pour $\mu = 0{,}00$

A2.3.2/ Ligne de délais gamma encastrés passe-bas

Un filtre gamma agira comme un filtre passe-bas lorsque la valeur de μ sera comprise entre 0 et 1, bornes exclues. Les filtres gamma permettent d'implanter dans ce cas des mémoires de plus ou moins grande profondeur, c'est à dire à plus ou moins long terme, et qui se caractérisent par une décroissance plus ou moins rapide de leur activité. Le paramètre μ permet de paramétrer la vitesse de la décroissance de l'activité de l'unité. Ainsi, plus μ sera proche de 0 et plus la décroissance de l'activité sera faible, entraînant donc la mise en place d'une mémoire de très grande profondeur. À l'inverse, lorsque μ sera proche de 1, la mémoire sera de très faible profondeur et la décroissance de l'activité sera très rapide.

Un fait intéressant à observer dans les figures suivantes est la pente de la pseudo droite obtenue par concaténation de l'activité des 200 délais de la lignes sur les 400 pas de temps observés. La figure la plus intéressante pour cette observation est la figure A2.16 qui correspond au cas où $\mu = 0{,}5$. Considérons la droite partant du coin en haut à gauche du graphique, le point de départ, et qui arrive au coin en bas à droite, qui correspond au $200^{ème}$ et dernier délai de la ligne lors de la $400^{ème}$ et dernière observation. Cette droite a une pente de 0,5, pente obtenue en divisant le nombre de délais par le nombre de pas de temps d'observation. La figure A2.16 montre bien que cette pente est également la pente, en moyenne, de la trace de mémoire au sein de la plaque dans le cas où $\mu = 0{,}5$.

Ce fait, évident à observer sur la figure A2.16, peut, en fait, être observé sur tous les autres graphiques dans le cas où le filtre gamma implante un filtre passe-bas. Il s'explique facilement grâce à l'équation 7.12 du chapitre 7. Cette équation donne l'instant t_p de passage du pic d'une impulsion dans une unité de rang k et s'écrit de la manière suivante :

$$t_p = \frac{k-1}{\mu} \qquad \text{(Éq. 2.2)}$$

Cette équation donne, en quelque sorte, une abscisse en fonction d'une ordonnée. Elle peut donc se réécrire de la manière suivante pour fournir une ordonnée en fonction d'une abscisse :

$$k = (\mu \cdot t_p) + 1 \qquad \text{(Éq. 2.3)}$$

Cette dernière équation se dérive très facilement et permet d'obtenir la pente de la courbe, pente qui est égale à μ, c.q.f.d.

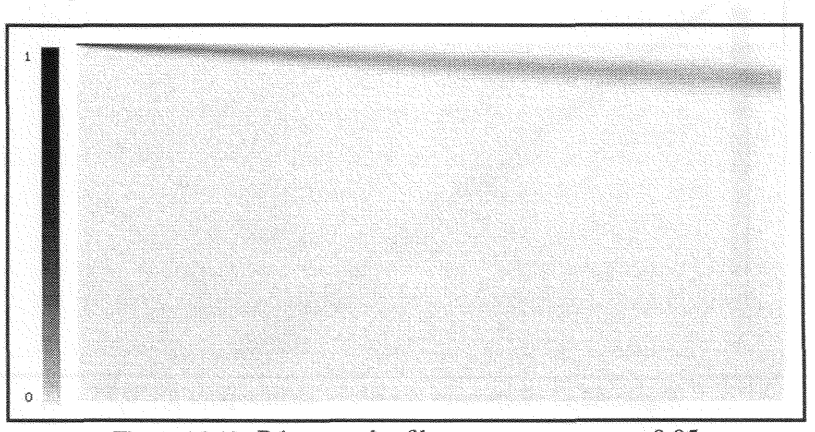


Figure A2.13 : Réponses des filtres gamma pour $\mu = 0{,}05$

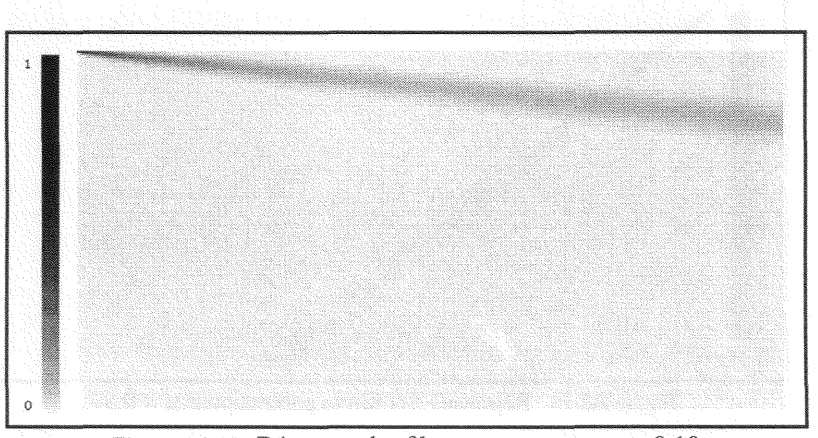


Figure A2.14 : Réponses des filtres gamma pour $\mu = 0{,}10$

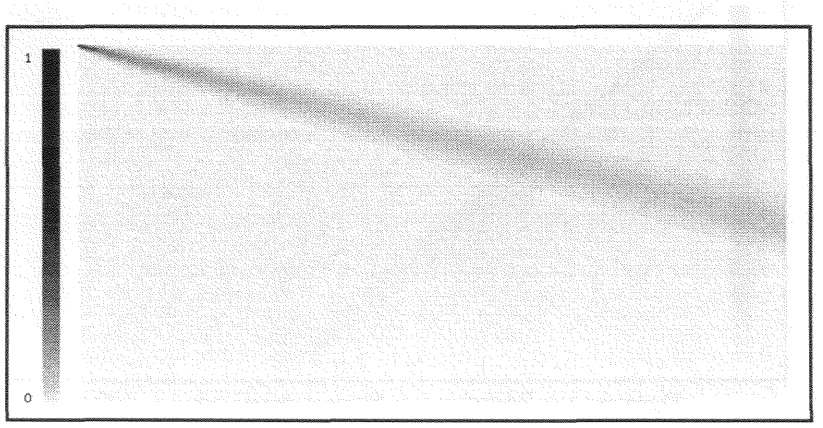


Figure A2.15 : Réponses des filtres gamma pour $\mu = 0{,}25$

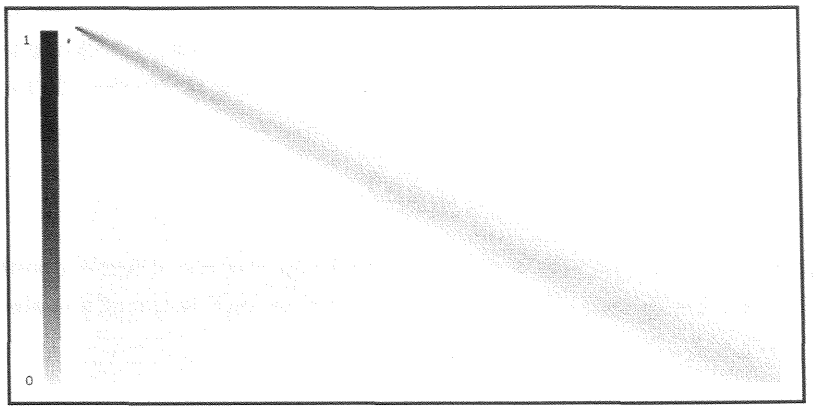


Figure A2.16 : Réponses des filtres gamma pour $\mu = 0,50$

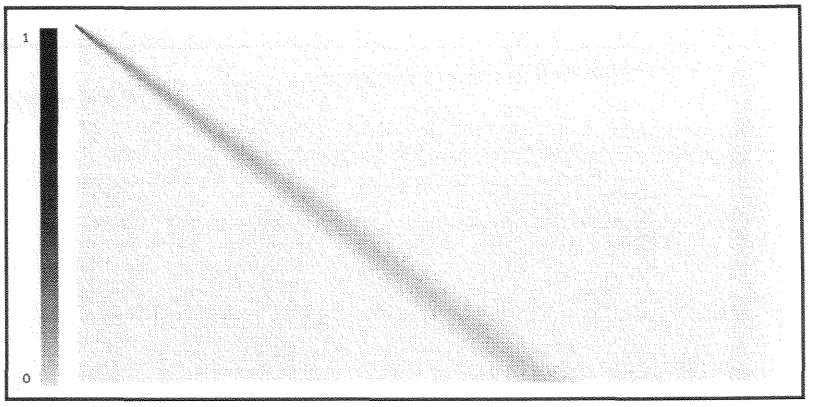


Figure A2.17 : Réponses des filtres gamma pour $\mu = 0,75$

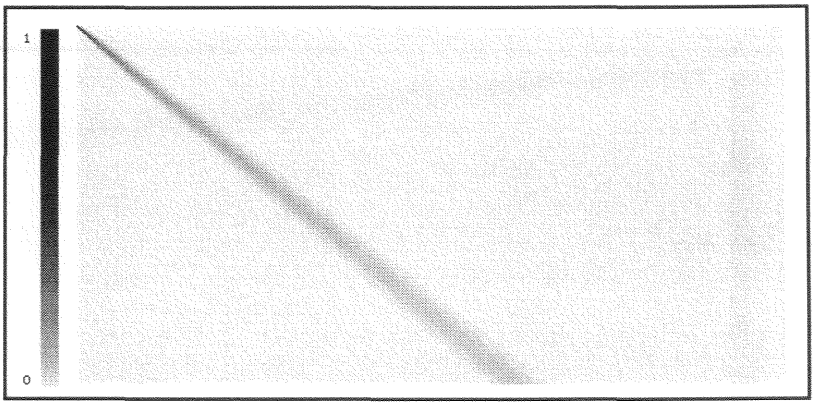


Figure A2.18 : Réponses des filtres gamma pour $\mu = 0,80$

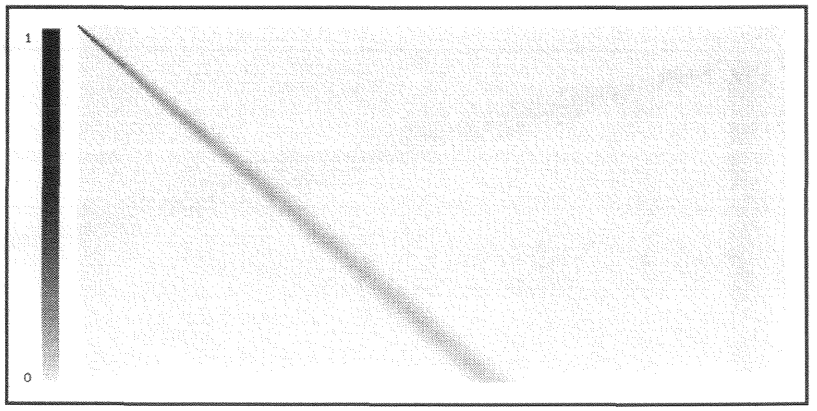


Figure A2.19 : Réponses des filtres gamma pour $\mu = 0,85$

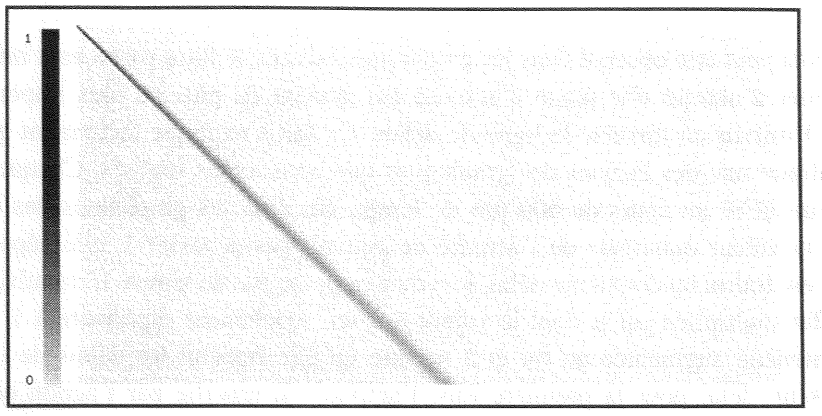


Figure A2.20 : Réponses des filtres gamma pour $\mu = 0{,}95$

A2.3.3/ Ligne de délais gamma encastrés équivalent à des délais simples

Le cas où le filtre gamma se comporte comme un délai simple est le cas où $\mu = 1$. Dans ce cas, si x est la valeur d'activation d'une unité gamma, la valeur de la récurrence est égale à $(1 - \mu) \times x = (1 - 1) \times x = 0$.

Les filtres de la ligne de délais n'implante donc pas de mémoire et l'impulsion est transférée, tout au long de cette ligne, à la manière d'un jeton. Il ne reste aucune donc aucune trace de l'impulsion une fois que celle-ci a franchi le dernier délai de la ligne. C'est ce mécanisme de mémorisation qui est mis en œuvre dans les réseaux à représentation du temps par mécanisme externe (cf. chapitre 6, paragraphe 6.1.2) tel que le réseau TDNN, *Time Delay Neural Network*, par exemple.

Une fois encore, on peut constater dans la figure A2.21 que la pente de la courbe est égale à la valeur de μ.

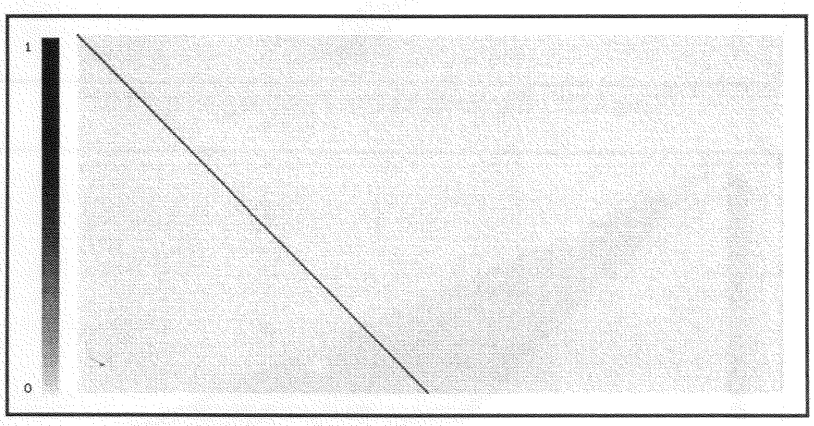


Figure A2.21 : Réponses des filtres gamma pour $\mu = 1{,}00$

A2.3.4/ Ligne de délais gamma encastrés passe-haut

Un filtre gamma dont le coefficient μ est compris entre 1 et 2, bornes exclues, agit comme un filtre passe-haut. Ce cas ne nous a pas intéressé pendant notre thèse quoiqu'il implante également un mécanisme de mémorisation. Ce mécanisme est cependant difficile à interpréter puisqu'il fait apparaître un système oscillant.

Prenons par exemple le cas où $\mu = 1{,}5$. Dans ce cas, une unité gamma dont le potentiel d'activité est de 1 au départ, verra cette activité osciller suivant la séquence : 1 ; -0,5 ; 0,25 ; -0,125 ; 0,0625 ; -0,03125. L'activité de l'unité gamma va donc en décroissant, en valeur absolue, et les valeurs de cette unité sont alternativement positives et négatives. Ce mécanisme pourrait être vu comme permettant de simuler un amortissement oscillant de l'activation de l'unité. Ce mécanisme semble cependant difficile à interpréter dans le cadre de la définition d'une unité à mémoire. Cette raison nous a donc poussé à interdire cette possibilité tout au long des études et essais que nous avons

réalisés.

Comme cela peut être observé dans les graphiques suivant, le filtre passe-haut implanté par l'unité gamma permet d'obtenir une plage d'activité qui devient de plus en plus importante en durée à mesure que l'impulsion traverse la ligne de délais. Ce fait s'explique facilement grâce à l'équation A2.1 et souligne une des lacunes des graphiques que nous avons réalisés. Chaque ligne représente l'activité d'un délai au cours de 400 pas de temps. Si, dans les graphiques des trois paragraphes précédents, la valeur maximale de l'activité ne pouvait pas dépasser 1, elle dépasse ici 1 dès que l'impulsion est fourni au deuxième délai, lors du deuxième pas de temps. L'impulsion, initialement à 1, est en effet multipliée par μ dont la valeur est, ici, strictement supérieure à 1. L'impulsion voit donc son intensité augmentée au fur et à mesure qu'elle franchit les délais de la ligne. Une fois arrivée dans un délai pour la première fois, l'activité est amortie par l'action combinée du délai considéré, qui oscille comme précédemment, et du délai précédent, dont l'activité oscille également.

La valeur de μ va donc ici définir le temps qui sera nécessaire pour que l'activité d'une cellule soit totalement amortie. La durée de cet amortissement sera, bien évidemment, fonction de la valeur de μ mais également fonction du rang du délai considéré.

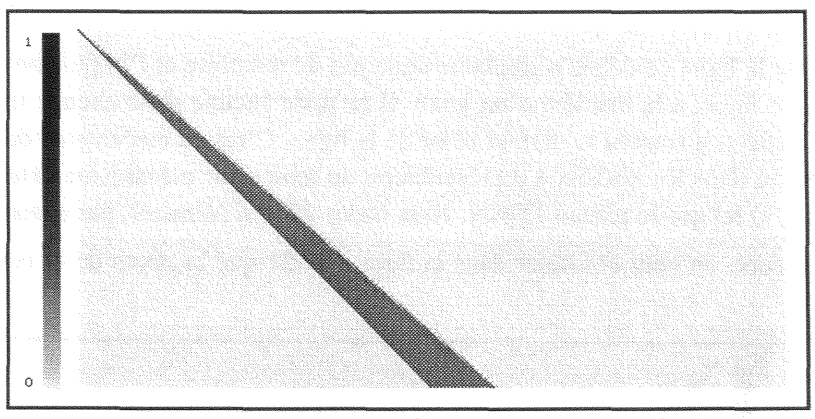


Figure A2.22 : Réponses des filtres gamma pour $\mu = 1{,}05$

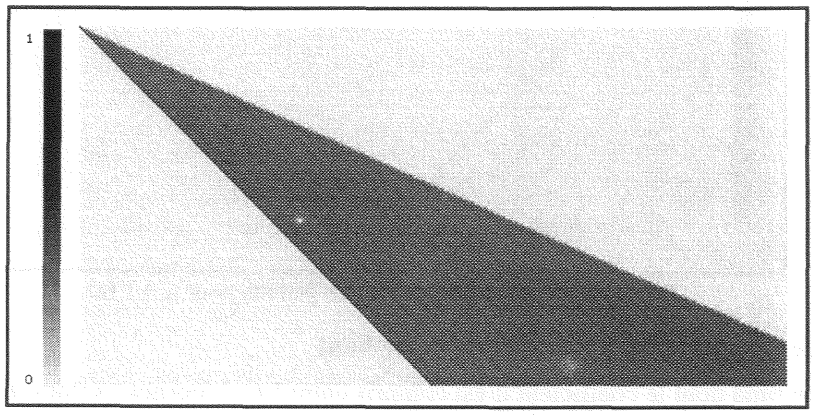


Figure A2.23 : Réponses des filtres gamma pour $\mu = 1{,}25$

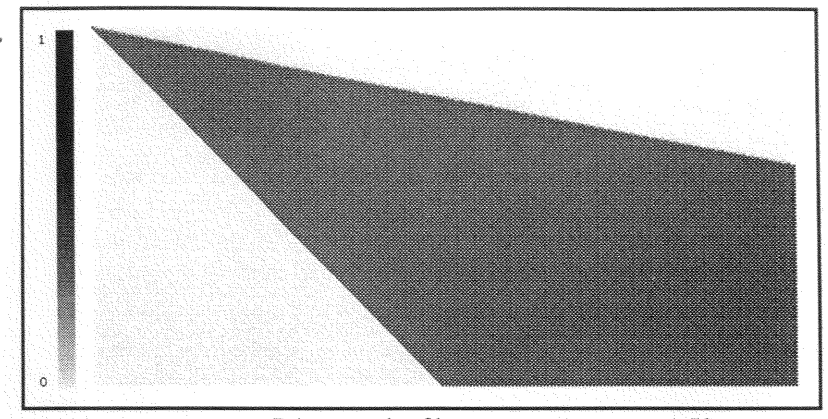


Figure A2.24 : Réponses des filtres gamma pour $\mu = 1{,}50$

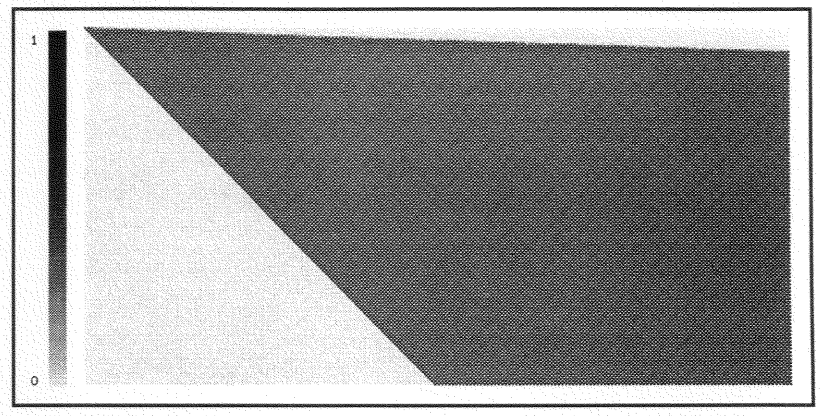


Figure A2.25 : Réponses des filtres gamma pour $\mu = 1{,}85$

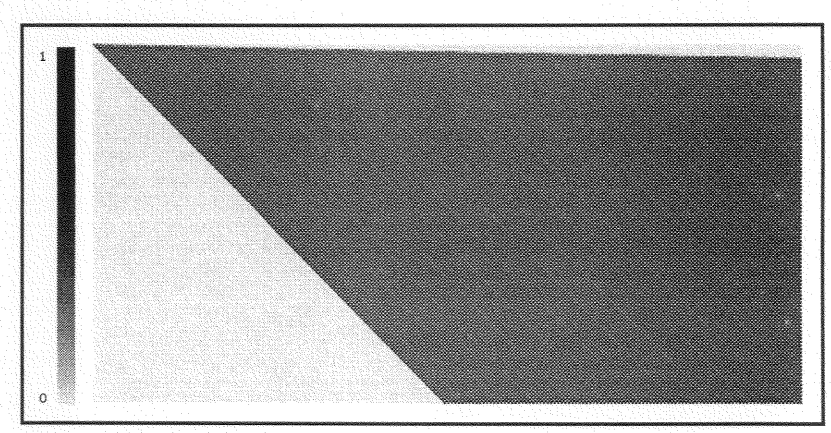


Figure A2.26 : Réponses des filtres gamma pour $\mu = 1{,}90$

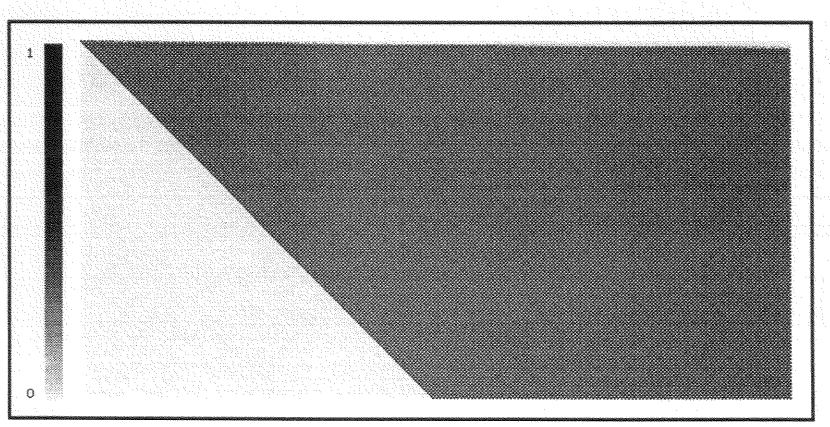
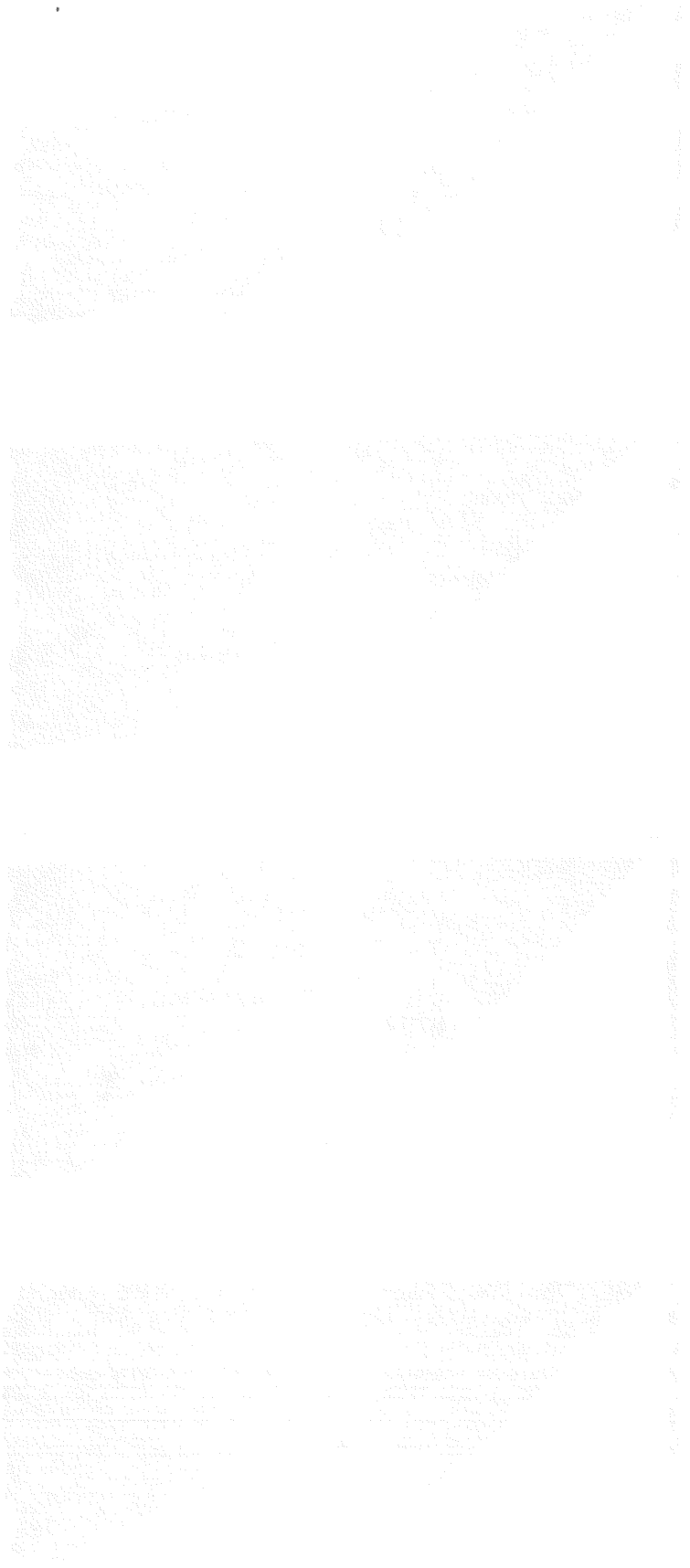


Figure A2.27 : Réponses des filtres gamma pour $\mu = 1{,}95$



Annexe 3 : Le corpus de bruits NOISEX

A3.1/ Introduction

L'objectif de cette annexe est de présenter brièvement le corpus de bruit NOISEX avec lequel nous avons travaillé lors de cette thèse. Le but de ce corpus est de fournir un ensemble de bruits standards pouvant servir de base de comparaison pour les différentes méthodes de traitement et de reconnaissance de la parole dans le bruit.

A3.2/ Le corpus Noise-Rom-0

Le premier corpus développé, le corpus Noise-Rom-0, comprend 24 bruits. Ce corpus a été développé en 1988 par l'Institut TNO pour l'étude de la perception, à Soesterberg aux Pays-Bas, avec l'appui du groupe de recherche et d'étude *Speech Processing* (RSG 10) de l'OTAN (Organisation du Traité de l'Atlantique Nord). Ces 24 bruits sont d'origine très diverses comme le montre la liste complète ci-dessous, liste donnée dans l'ordre croissant des numéros de bruit.

- bruit généré par une sinusoïde ayant une fréquence de 1000 Hz, bruit Noise-Rom-0 n° 1,
- bruit rose, bruit Noise-Rom-0 n° 2,
- bruit blanc, bruit Noise-Rom-0 n° 3,
- bruit blanc atténué de 6 décibels par octave de 250 Hz, bruit Noise-Rom-0 n° 4,
- bruit blanc atténué de 12 décibels par octave de 250 Hz, bruit Noise-Rom-0 n° 5,
- bruit de parole (*speech noise*), bruit synthétique ayant les propriétés de masquage d'un environnement bruité par de la parole, bruit Noise-Rom-0 n° 6,
- bruit du M 109 à 30 km/h, bruit Noise-Rom-0 n° 7,
- bruit de Buccaneer à 190 nœuds et 1000 pieds, bruit Noise-Rom-0 n° 8,
- bruit du Leopard 2 à 70 km/h, bruit Noise-Rom-0 n° 9,
- bruit de véhicule de transport à roues à 50-60 km/h, bruit Noise-Rom-0 n° 10,
- bruit de Buccaneer à 450 nœuds et 300 pieds, bruit Noise-Rom-0 n° 11,
- bruit de l'hélicoptère Lynx sur plate-forme, bruit Noise-Rom-0 n° 12,
- bruit du Leopard 1 à 70 km/h, bruit Noise-Rom-0 n° 13,
- Salle de commandement d'un contre-torpilleur, bruit Noise-Rom-0 n° 14,
- Salle des machines d'un contre-torpilleur, bruit Noise-Rom-0 n° 15,
- bruit de mitrailleuse, bruit Noise-Rom-0 n° 16,
- bruit de canal radio hautes-fréquences, bruit Noise-Rom-0 n° 17,
- signal de test du bateau STITEL, bruit Noise-Rom-0 n° 18,
- bruit de parole (*voice babble, canteen, 100 people*), bruit réel, bruit Noise-Rom-0 n° 19,
- bruit d'un chasseur F16 biplace à 500 nœuds et 300-600 pieds en place copilote, bruit Noise-Rom-0 n° 20,
- bruit d'une usine de production de voitures : bruits de soudures électriques lors de

l'assemblage du bas de caisse (*car floor production*), bruit Noise-Rom-0 n° 21,

- bruit d'une usine de production de voitures : bruits du hall d'assemblage (*car production hall*), bruit Noise-Rom-0 n° 22,
- bruit de voiture en déplacement : Volvo 340 à 120 km/h en 4ème vitesse sur une route goudronnée, bruit Noise-Rom-0 n° 23,
- bruit de voiture en déplacement : Volvo 340 à 50 km/h en 3ème vitesse sur une route pavée, bruit Noise-Rom-0 n° 24.

Tous ces bruits peuvent être regroupés en trois catégories différentes en fonction de leur univers de rattachement : bruits de référence propres au domaine du décodage acoustico-phonétique, bruits d'origine industrielle (ou plutôt civile...) et bruits d'origine militaire.

Les bruits propres au domaine de DAP regroupent les bruits synthétiques et les bruits de parole bien que cette dernière catégorie puisse, à elle seule, constituer une classe. Les bruits synthétiques sont, généralement, générés à partir de fonctions sinusoïdales ou gaussiennes et possèdent des spectres très stables. Dans cette première sous-catégories se trouvent les bruits 1, 2, 3, 4 et 5 du corpus Noise-Rom-0. Les bruits de parole, la deuxième sous-catégorie de ce groupe, sont les bruits 6 et 19 du corpus. Cependant, une grande différence existe entre ces deux bruits : alors que le bruit numéro 6 est un bruit synthétique permettant d'obtenir les caractéristiques moyennes de masquage produit par la parole de tierces personnes vis-à-vis de l'utilisateur d'un système de RAP, le bruit 19 est un bruit réel enregistré dans un local fréquenté par une centaine de personnes. La ressemblance "en moyenne" de ces deux bruits et leurs fortes différences au niveau de la stabilité peuvent être observées en comparant la figure A3.2 à la figure A3.7.

La deuxième catégorie regroupe les bruits d'origine industrielle et, plus généralement, civile. Cette catégorie est composée de deux fois deux bruits : bruits de véhicules en déplacement (numéros 23 et 24) et bruits d'atelier de fabrication (bruit 21 et 22). Alors que les deux premiers sont stationnaires, les deux derniers ne le sont pas.

La dernière catégorie regroupe tous les bruits d'origine militaire, que ces bruits soient stationnaires ou non. Ces bruits ont été enregistrés à bord de véhicules des trois "armes" (terre, air et mer) des armées de l'OTAN. Les bruits de canal radio hautes-fréquences (bruit 17) et de rafales de mitrailleuse (bruit 16) sont les seules exceptions à cette règle. L'expérimentateur dispose ainsi de bruits de véhicules terrestres (bruits 7, 9, 10 et 13), de bruits enregistrés à bord d'unités navales (bruits 14, 15 et 18) et de bruits originaires du monde aéronautique (bruits 8, 11, 12, et 20).

Tous ces bruits sont enregistrés à une fréquence d'échantillonnage de 20 MHz.

L'ensemble des bruits contenu dans le corpus n'a pas pour but d'être exhaustif. Il constitue simplement un ensemble de référence pour l'étude du comportement des méthodes de reconnaissance automatique de la parole lorsqu'elles sont mises en présence de bruits additifs. Les bruits présents dans le corpus sont cependant variés et permettent de bruiter un signal de parole avec des bruits stationnaires ou non. Ces bruits sont fournis de manière isolés et aucun signal de parole bruité n'est fourni en complément alors qu'un bruitage effectué a priori aurait permis de disposer d'une base de comparaison encore mieux définie. Pour palier à cet inconvénient, un autre corpus de bruits, comprenant également des signaux de parole bruité selon un certain nombre de rapports signal-sur-bruit, a été défini.

A3.3/ Le corpus Noisex-92

Le corpus Noisex-92 a été conjointement mis au point, en 1992, à partir du corpus Noise-Rom-0 par l'Institut TNO pour l'étude de la perception et par l'équipe de recherche sur la parole de la *Defense Research Agency* anglaise. Seuls certains bruits ont été sélectionnés par rapport à l'ensemble

de ceux disponibles dans le premier corpus. De plus, ces bruits ont été rééchantillonnés de manière à être compatibles avec des signaux de parole préalablement enregistrés à une fréquence de 16 MHz. Il est à noter que les spectrogrammes présentés ci-après correspondent cependant aux bruits du corpus Noise-Rom-0, échantillonné à 20 MHz, et non aux bruits du corpus NOISEX-92. En complément de ces bruits sont fournis des signaux de parole dans différentes conditions de bruits et, ce, pour tous les bruits du corpus : parole non bruitée et parole bruitée à des rapports signal-sur-bruit de 18, 12, 6, 0 et -6 décibels.

Plutôt que de classer les différents bruits sélectionnés en fonction de leur origine (synthétique, civile ou militaire), nous avons ici essayé de les classer en fonction de leurs caractéristiques de stationnarité. Nous avons ainsi définis trois classes : celle des bruits stationnaires, celles des bruits rythmiques et celles des bruits aléatoires. Ici, nous n'entendons pas par aléatoire que le signal est lui-même aléatoire mais plutôt que le processus qui l'a engendré peut être considéré comme aléatoire.

Les spectrogrammes de bruit présentés dans cette annexe ont été calculés de manière à ce que le graphique présente des valeurs minimales de 20 décibels et des valeurs maximales de 90 décibels.

A3.3.1/ Bruits stationnaires ou quasi stationnaires

Les bruits stationnaires de NOISEX sont aux nombre de quatre. Ils présentent des difficultés diverses et plus ou moins importantes.

Le bruit de voiture (bruit d'une voiture Volvo 340 à 120 kilomètres à l'heure sur une route goudronnée et en 4ème vitesse) n'est pas fondamentalement problématique à de forts rapports signal sur bruit (RSSB). Il peut cependant devenir très incommodant pour un système de RAP à de faibles RSSB puisque l'énergie de ce bruit est concentrée dans les basses fréquences, fréquences où se trouvent les formants des voyelles (figure A3.1).

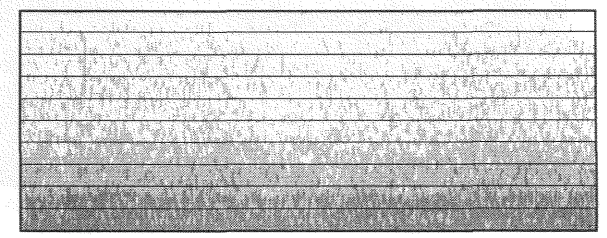


Figure A3.1 : Spectrogramme du bruit de voiture.

Le bruit synthétique de parole (*speech noise*) présente la même répartition spectrale que le bruit de voiture mais est cependant plus énergétique (figure A3.2). Ce bruit, plus que le bruit de la voiture Volvo, peut présenter de réels inconvénients pour un système de RAP puisque l'énergie de se bruit est concentré en basses fréquences, en dessous de 4 MHz. Il est cependant loin d'avoir le caractère problématique du *voice babble*, qui est un véritable bruit de parole (figure A3.7).

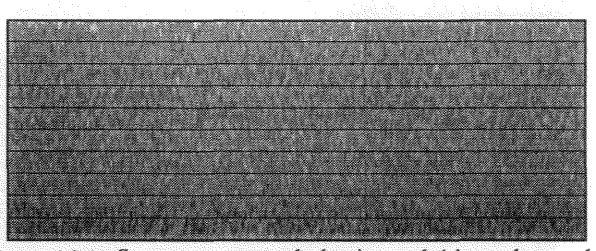


Figure A3.2 : Spectrogramme du bruit synthétique de parole.

Les deux bruits que nous venons de présenter peuvent nettement amoindrir les capacités des systèmes de RAP mais possèdent des répartitions spectrales assez différentes de celles de la parole véritable. D'autres bruits stationnaires possèdent des caractéristiques qui les font plus ressembler à de la parole, tout au moins à des voyelles : ces bruits possèdent des traces formantiques dans leur

spectre.

Le premier de des deux bruits qui rentrent dans la catégorie des bruits stationnaires de nature formantique est le bruit de l'avion F16. Ce bruit a été enregistré en place arrière d'un F16 biplace volant à une vitesse de 500 nœuds et à une altitude comprise entre 300 et 600 pieds. Ce bruit est, comme le montre son spectrogramme (figure A3.3), de nature formantique. Ces formants correspondent à la vitesse de fonctionnement des organes moteur de l'avion, en l'occurrence du réacteur. Ces formants, en se superposant à la parole, peuvent créer l'illusion d'une présence de voyelles pour certains systèmes élémentaires de DAP. Leur stationnarité, dans un régime de vol donné, permet cependant de les isoler assez facilement.

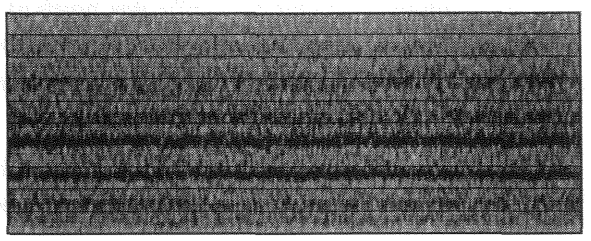


Figure A3.3 : Spectrogramme du bruit de l'avion F16.

Le deuxième bruit stationnaire de nature formantique est le bruit de l'hélicoptère Lynx enregistré sur une plate-forme. Les caractéristiques générales de ce bruit (figure A3.4) sont similaires aux caractéristiques du bruit du F16 (figure A3.3). Les organes moteur fonctionnant cependant moins vite, les traces formantiques sont de fréquences moins élevées. Cette dernière caractéristique fait que le bruit de l'hélicoptère Lynx dégradera plus facilement les voyelles que ne le fait le bruit de l'avion F16.

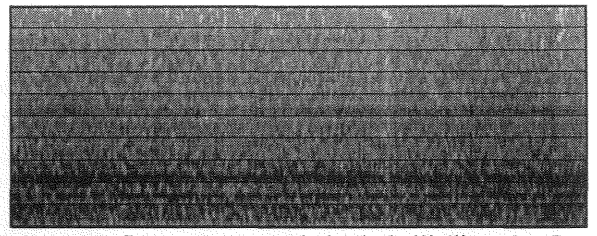


Figure A3.4 : Spectrogramme du bruit de l'hélicoptère Lynx.

A3.3.2/ Bruits rythmiques

Le premier bruit rythmique de NOISEX correspond au bruit enregistré dans un salle d'opérations d'un contre-torpilleur. Ce bruit, dont le spectrogramme est présenté dans la figure A3.5, présente un rythme simple qui est visible dans la succession des raies spectrales. Ce bruit est assez intense mais l'énergie des raies n'est pas beaucoup plus importante que l'énergie du bruit de fond. Il est, en outre, possible d'observer la présence de bruits aléatoires supplémentaires en basses fréquences, entre 0 et 3000 Hz. Ce bruit est cependant très diffus.

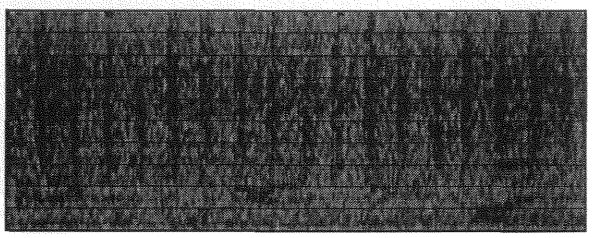


Figure A3.5 : Spectrogramme du bruit d'une salle d'opérations d'un contre-torpilleur.

Un deuxième bruit rythmique présent dans la base NOISEX est le bruit de test du bateau STITEL. Le rythme est ici établi sur quatre bandes de fréquences différentes. Ces bandes s'échelonnent respectivement de 1000 à 1500 Hz, de 1500 à 3000 Hz, de 3000 à 5500Hz et de 5500 à 10000 Hz.

Elles constituent un bruit qu'il serait possible de traiter aisément avec un système ayant appris les différents intervalles de temps de présence et d'absence d'énergie dans ces quatre bandes de fréquences. Une observant de la figure A3.6 permettra en outre au lecteur de se rendre compte de l'indépendance des rythmes dans les quatre bandes considérées. Il n'existe cependant encore aucun modèle permettant de modéliser ces rythmes différents, modélisation qui pourrait permettre d'éliminer ce bruit avec les techniques actuelles de compensation ou avec toute autre technique conduisant à des résultats satisfaisant.

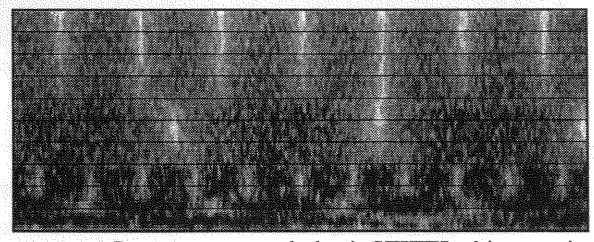


Figure A3.6 : Spectrogramme du bruit STITEL ship test signal.

A3.3.3/ Bruits aléatoires

Rappelons tous d'abord que nous qualifions ici de bruits aléatoires les bruits générés par des processus aléatoires de haut niveau, ces processus pouvant correspondre à une discussion entre plusieurs personnes, à une séance de travail en atelier ou dans un bureau. Ces bruits ne correspondent donc pas à des signaux temporels générés aléatoirement, cette génération aléatoire étant un processus de bas niveau.

Le bruit de parole *voice babble*, appelé également bruit de cocktail party, est un bruit qui génère de grandes difficultés dans les systèmes de RAP. Le *voice babble* de NOISEX a, par exemple, été enregistré dans un réfectoire où se trouvait une centaine de personnes. Le bruit est donc composé de différents signaux de parole se superposant plus ou moins les uns aux autres en fonction de la distance du locuteur au microphone à un instant donné. Ce bruit devient très problématique lorsque le rapport signal sur bruit devient négatif. Il présente en effet strictement les même caractéristiques que le signal de parole à analyser et peut donc conduire à de très nombreuses erreurs. Ces bruits semblent difficiles à traiter avec les techniques actuelles de la RAP puisque l'homme, en tant qu'auditeur, semble utiliser toutes ses facultés pour surmonter ce type de problème avec l'utilisation de la binauralité et de la connaissance du timbre de la voix du locuteur, connaissance qui permet d'effectuer un "suivi" de la voix.

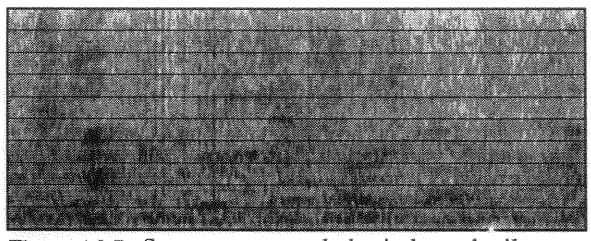


Figure A3.7 : Spectrogramme du bruit de cocktail party.

Les bruits associés à des ateliers de fabrication ou à des usines peuvent également générer de nombreuses erreurs dans les systèmes de RAP. L'univers de la production est généralement très bruyant (cf. chapitre 3, paragraphe 3.2.3.1) et le bruit, qui en constitue l'environnement sonore, est généralement très élevé. Au niveau de la caractérisation de ces bruits, il faut tout d'abord considérer le cas des bruits rythmiques correspondant aux automates mis en place dans une unité de production. Mais ces bruits ne sont pas les seuls. L'ouvrier, qui est avant tout être humain (...), ne travaille pas en suivant le rythme d'une horloge à la manière d'un robot et produit donc des bruits à intervalles irréguliers. Ce fait peut conduire à la génération de bruits tel que celui dont le spectrogramme est donné en figure A3.8. Ce bruit correspond à l'enregistrement des bruits de soudures électriques dans

un atelier de montage de bas de caisse de voitures. Des bruits de friction (raies verticales) correspondant aux instants de soudure y sont visibles, à intervalles irréguliers.

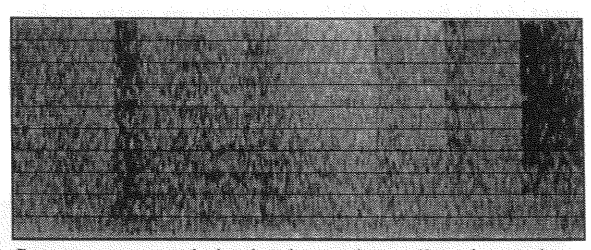


Figure A3.8 : Spectrogramme de bruits de soudures électriques dans un atelier de montage de bas de caisse.

Le dernier bruit que nous avons classé dans les bruits aléatoires est le bruit de rafales de mitrailleuse. Ce bruit est aléatoire car, comme on peut le voir sur le spectrogramme de la figure A3.9, l'intervalle de temps entre deux détonations successives peut, approximativement, varier du simple au triple. Ce bruit est également très problématique car il peut créer pendant de courts instants de très fortes perturbations dans le signal de parole.

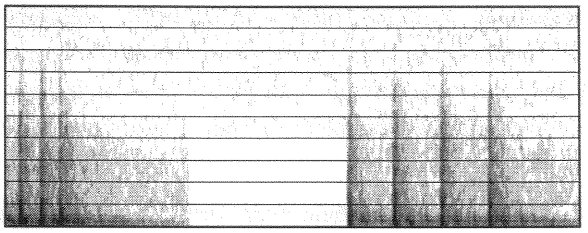


Figure A3.9 : Spectrogramme du bruit de rafales de mitrailleuse.

Annexe 4 : Bibliographie

[abarbanel93] H. D. I. Abarbanel, R. Brown, J. J. Sidorowich et L. S. Tsimring. The analysis of observed chaotic data in physical systems. Reviews of Modern Physics, vol. 65, no 4, pp 1331-1392, 1993.

[abbott90] L. F. Abbott et T. B. Kepler. Model neurons: from Hodgkin-Huxley to Hopfield. Statistical mechanics of neural networks, pp 5-18, Springer-Verlag, 1990.

[acero90a] A. Acero. Acoustical and environmental robustness in automatic speech recognition. Thèse de doctorat mention informatique, 208 pp, Université Carnegie Mellon, Pittsburgh (PA, États-Unis), 1990. (et Kluwer Academic Press, 1992)

[acero90b] A. Acero et R. M. Stern. Environmental robustness in automatic speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 849-852, 1990.

[ackley85] D. H. Ackley, G. E. Hinton et T. J. Sejnowski. A learning algorithm for Boltzmann machines. Cognitive Science, vol. 9, pp 147-169, 1985.

[aho74] A. V. Aho, J. E. Hopcroft et J. D. Ullman. The design and analysis of computer algorithms, chapitre 7 : The fast Fourier transform and its application, pp 251-276, 1974.

[aihara90] K. Aihara, T. Takabe, M. Toyoda. Chaotic neural networks. Physics Letters A, vol. 144, nos 6-7, pp 333-340, 1990.

[aktas90] A. Aktas, O. Schmidbauer, K.-H. Maier et W. H. Feix. Classification of coarse phonetic categories in continuous speech: statistical classifiers vs. temporal flow connectionist network. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 89-92, 1990.

[alexandre90] F. Alexandre. Une modélisation fonctionnelle du cortex : la colonne corticale - Aspects visuels et moteurs. Thèse de doctorat mention informatique, 157 pp, Université de Nancy 1, Nancy (France), 1990.

[alexandre93] P. Alexandre, J. Boudy et P. Lockwood. Evaluation of car noise reduction/compensation techniques for digit recognition in a speaker-independent context. Proceedings of the European Conference on Speech Communication and Technology, vol. 2, pp 1255–1258, 1993.

[allen83] J. F. Allen. Maintaining knowledge about temporal intervals. Communications of the Association for Computing Machinery, vol. 26, no 11, pp 832-843, 1983.

[allen84] J. F. Allen. Towards a general theory of action and time. Artificial Intelligence, vol. 23, pp 123-154, 1984.

[allen94] J. B. Allen. How do humans process and recognize speech? IEEE Transactions on Speech and Audio Processing, vol. 2, no 4, pp 567-577, 1994.

[almeida87] L. B. Almeida. A learning rule for asynchronous perceptrons with feedback in a combinatorial environment. Proceedings of the International Conference on Neural Networks, vol. 2, pp 609-618, 1987.

[alpaydin91] E. Alpaydin. GAL: networks that grow when they learn and shrink when they forget. Rapport technique 91-032 de l'International Computer Science Institute, Berkeley (CA, États-Unis), 1991.

[altman96] J. Altman. Epilogue: Models of decision-making. [damasio96a], pp 201-206.

[altosaar91] T. Altosaar et M. Karjalainen. Event-based recognition and analysis of speech by neural networks. Proceedings of the European Conference on Speech Communication and Technology, pp 1031-1034, 1991.

[amari92] S.-I. Amari, N. Fujita et S. Shinomoto. Four types of learning curves. Neural Computation, vol. 4, pp 605-618, 1992.

[anderson91] T. R. Anderson. Speaker independent phoneme recognition with an auditory model and a neural network: a comparison with traditional techniques. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 149-152, 1991.

[anglade92a] Y. Anglade, D. Fohr et J.-C. Junqua. Selectively trained neural networks for the discrimination of normal and Lombard speech. Proceedings of the International Conference on Spoken Language Processing, pp 595-598, 1992.

[anglade92b] Y. Anglade, D. Fohr et J.-M. Pierrel. Reconnaissance de vocabulaires difficiles à l'aide de réseaux de neuronaux. Article non publié, 6pp.

[anglade93] Y. Anglade, D. Fohr et J.-C. Junqua. Speech discrimination in adverse conditions using acoustic knowledge and selectively trained neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 279-282, 1993.

[anglade94] Y. Anglade. Robustesse de la reconnaissance automatique de la parole : étude et application dans un système d'aide vocal pour une standardiste mal-voyante. Thèse de doctorat mention informatique, 232 pp, Université de Nancy 1, Nancy (France), 1994.

[ansis35] ANSI S3.5: Methods for the calculation of the articulation index. American National Standards Institute, New-York (NY, États-Unis), 1969.

[atlan92] H. Atlan. L'organisation biologique et la théorie de l'information. 300 pp, Collection Actualités scientifiques et industrielles, no 1432, Éditions Hermann, 1992.

[azencott90] R. Azencott. Synchronous Boltzmann machines and Gibbs fields: learning algorithms. Neurocomputing, série NATO ASI, vol. F68, pp 51-62, Springer-Verlag, 1990.

[azencott92a] R. Azencott. Sequential simulated annealing: speed of convergence and acceleration techniques. Simulated annealing: Parallelization Techniques, pp 1-10, Wiley Interscience, 1992.

[azencott92b] R. Azencott. Parallel simulated annealing: an overview of basic techniques. Simulated annealing: Parallelization Techniques, pp 37-46, Wiley Interscience, 1992.

[azencott92c] R. Azencott. Boltzmann machines: higher-order interactions and synchronous learning. Lecture Notes in statistics: Stochastic models in image analysis, 32 pp, Springer-Verlag, 1992.

[azencott93] R. Azencott, A. Doutriaux et L. Younes. Synchronous Boltzmann machines and curve identification tasks. Article de revue distribué à l'École d'Été d'Analyse Numérique CEA-INRIA-EDF Réseaux de Neurones et Applications, pp 461-480, 1993.

[azencott94] R. Azencott. Machines de Boltzmann. Manuel de cours, 21 pp, École d'Été d'Analyse Numérique CEA-INRIA-EDF Réseaux de Neurones et Applications, 1994.

[back91] A. D. Back et A. C. Tsoi. FIR and IIR synapses, a new neural network architecture for time series modelling. Neural Computation, vol. 3, no 3, pp 375-385, 1991.

[back95] A. D. Back et A. C. Tsoi. A comparison of discrete time operators for nonlinear system identification. Advances in Neural Information Processing Systems, 8 pp, 1995.

[bahl86] L. R. Bahl, P. F. Brown, P. V. de Souza et R. L. Mercer. Maximum mutual information estimation of hidden Markov models for automatic speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 49-52, 1986.

[baldi94] P. Baldi et A. Atiya. How delays affect neural dynamics and learning. Rapport technique, 29 pp, Jet Propulsion Laboratory, Pasadena (CA, États-Unis), 1994.

[baldi95] P. Baldi. Gradient descent learning algorithm overview: a general dynamical systems perspective. IEEE Transactions on Neural Networks, vol. 6, no 1, pp 182-195, 1995.

[baum67] L. E. Baum et J. A. Egon. An inequality with applications to statistical estimation for probabilistic functions of a Markov process and to a model for ecology. Bulletin of the American Meteorological Society, vol. 73, pp 360-363, 1967.

[beattie92] V. L. Beattie et S. J. Young. Hidden Markov model state-based cepstral noise compensation. Proceedings of the International Conference on Spoken Language Processing, pp 519-522, 1992.

[beaufays94] F. Beaufays et E. A. Wan. Relating real-time backpropagation and backpropagation through time: an application of flow graph interreciprocity. Neural Computation, vol. 6, pp 296-306, 1994.

[bell93] A. J. Bell. Self-organisation in real neurons: anti-Hebb in "channel space". Advances in Neural Information Processing Systems, pp 59-66, 1993.

[bell95] A. J. Bell et T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. Neural Computation, vol. 7, pp 1129-1159, 1995.

[bellman57] R. E. Bellman. Dynamic Programming. Princeton University Press, 1957.

[bellman58] R. E. Bellman. On a routing problem. Quaterly Journal of Applied Mathematics, vol. 16, pp 87-90, 1958.

[bendiksen90] A. Bendiksen et K Steiglitz. Neural networks for voiced/unvoiced speech classification. Proceedings of the International Conference on Acoustic, Speech and Signal Processing, pp 521-524, 1990.

[bengio93a] Y. Bengio, P. Frasconi, M. Gori et G. Soda. Recurrent neural networks for adaptive temporal processing. Proceedings of the 6th Italian Workshop on Neural Networks (WIRN93), pp 1183-1195, 1993.

[bengio93b] Y. Bengio, P. Frasconi et P. Simard. The problem of learning long-term dependencies in recurrent networks. Proceedings of the IEEE International Conference on Neural Networks, pp 1183-1188, 1993.

[bengio94a] Y. Bengio, P. Simard et P. Frasconi. Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 157-166, 1994.

[bengio94b] Y. Bengio et P. Frasconi. Credit assignment through time: alternative to backpropagation, Advances in Neural Information Processing Systems, 8 pp, 1994.

[bengio96] Y. Bengio. Neural networks for speech and sequence recognition. International Thomson Computer Press, 167 pp, 1996.

[bennani91a] Y. Bennani et P. Gallinari. On the use of TDNN-extracted features information in talker identification. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 385-388, 1991.

[bennani91b] Y. Bennani, N. Chaourar, P. Gallinari et A. Mellouk. Validation of neural net architectures on speech recognition tasks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 97-100, 1991.

[beranek54] L. L. Beranek. Acoustics. 481 pp, McGraw-Hill, 1954.

[berthold94] M. R. Berthold. A time delay radial basis function network for phoneme recognition. Proceedings of the International Conference on Neural Networks, 4 pp, 1994.

[berthommier92] F. Berthommier. Intégration neuronale dans le système auditif - Modélisation de réseaux neuronaux temporo-dépendants. Thèse de doctorat mention informatique, Université Joseph Fourier - Grenoble I, Grenoble (France), 127 pp, 1992.

[bestougeff89] H. Bestougeff et G. Ligozat. Outils logiques pour le traitement du temps - De la linguistique à l'intelligence artificielle. 272 pp, Collection Études et Recherches en Informatique. Masson, 1989.

[bianchini94] M. Bianchini, M. Gori et M. Maggini. On the problem of local minima in recurrent neural networks. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 167-177, 1994.

[biehl94] M. Biehl et H. Schwarze. Learning by online gradient descent. Rapport technique LU-TP-94-10, 21 pp, Université de Lund, Lund (Suède), 1994.

[bimbot90] F. Bimbot, G. Chollet et J.-P. Tubach. Phonetic features extraction using time-delay neural networks. Proceedings of the International Conference on Spoken Language Processing, pp 665- 668, 1990.

[bimbot90] F. Bimbot, G. Chollet et J.-P. Tubach. TDNN for phonetic features extraction : a visual exploration. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 73-76, 1991.

[bishop95] C. M. Bishop. Neural networks for pattern recognition. 499 pp, Oxford University Press, 1995.

[bodenhausen91a] U. Bodenhausen et A. Waibel. The tempo 2 algorithm: adjusting time delays by supervised learning. Advances in Neural Information Processing Systems, pp 155-161, 1991.

[bodenhausen91b] U. Bodenhausen et A. Waibel. Learning the architecture of neural networks for speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 117-120, 1991.

[boe94] L.-J. Boë et P. Iranzo. Invention médiatique et découverte scientifique - Un sérieux cas d'amnésie vocale. Les cahiers de l'ICP, collection monographies, 120 pp, 1994. Étude du cas "Martine Kempf", 1984-1985.

[bourlard90a] H. Bourlard et N. Morgan. A continuous speech recognition system embedding MLP into HMM. Advances in Neural Information Processing Systems, pp 186-193, 1990.

[bourlard90b] H. Bourlard et C. J. Wellekens. Links between Markov models and multilayer perceptrons. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no 12, pp 1167-1178, 1990.

[bourlard93] H. Bourlard et N. Morgan. Continuous speech recognition by connectionnist statistical methods. IEEE Transactions on Neural Networks, vol. 4, no 6, pp 893-908, 1993.

[bourlard95a] H. Bourlard. Towards increasing speech recognition error rates. Proceedings of the European Conference on Speech Communication and Technology, vol. 2, pp 883-894, 1995.

[bourlard95b] H. Bourlard et N. Morgan. Connectionist techniques. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 11.5, pp 412-418, 1995.

[bourlard96a] H. Bourlard. Reconnaissance automatique de la parole : modélisation ou description ?. Actes des 21èmes Journées d'Étude sur la Parole, pp 263-272, 1996.

[bourlard96b] H. Bourlard, H. Hermansky et N. Morgan. Towards increasing speech recognition error rates. Speech Communication, vol. 18, pp 205-231, 1996.

[brauer89] P. Brauer et P. Knagenhjelm. Infrastructure in Kohonen maps. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 647-650, 1989.

[bresslof93] P. C. Bresslof. Complex dynamics of a discrete time model of a neuron. Neural networks dynamics, pp 103-121, Springer Verlag, 1993.

[bridle95] J. S. Bridle. Optimization and search in speech and language processing. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 11.7, pp 423-428, 1995.

[brinker92] A. C. Brinker et J. A. Roufs. Evidence for a generalized Laguerre transform of temporal events by the visual system. Biological Cybernetics, no 67, pp 395-402, 1992.

[brooks91] R. A. Brooks. Intelligence without reason. Rapport technique AI memo 1293, 27 pp, Massachusetts Institute of Technology, Cambridge (MA, États-Unis), 1991.

[brooks93] R. A. Brooks et L. A. Stein. Building brains for bodies. Rapport technique AI memo 1439, 15 pp, Massachusetts Institute of Technology, Cambridge (MA, États-Unis), 1993.

[broome65] P. Broome. Discrete orthogonal sequences. Journal of the Association for Computing Machinery, vol. 12, no 12, pp 151-168, 1965.

[buhrke91] E. R. Buhrke et J. L. Lo Cicero. Speech recognition with neural networks and network fusion. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 157-160, 1991.

[bullier83] J. Bullier. Les cartes du cerveau. La recherche, vol. 14, no 148, pp 1202-1214, 1983.

[buniet91] L. Buniet. Contribution à la reconnaissance des consonnes sonantes du français. Mémoire de DEA informatique, 74 pp, Université de Nancy-1, Nancy (France), 1991.

[buniet93a] L. Buniet, D. Fohr, Y. Anglade, J.-C. Junqua, J.-M. Pierrel. Selectively trained neural networks for connected word recognition in noisy environments. Proceedings of the European Conference on Speech Communication and Technology, vol. 2, pp 841-845, 1993.

[buniet93b] D. Bechet, L. Buniet, P. Deville, Y. Lahlou, Y. Lallement, B. Simonnot, P. Subtil. Intelligence artificielle : perspectives historiques. Rapport Interne CRIN 93-R-149, 30 pp, Centre de Recherche en Informatique de Nancy, Nancy (France), 1993.

[buniet94] L. Buniet, D. Fohr, J.-C. Junqua. Une méthode connexionniste pour la reconnaissance de mots enchaînés en milieu bruité. Actes de la Conférence Neurosciences et Sciences de l'Ingénieur (NSI), pp 27-30, 1994.

[buniet95a] L. Buniet, S. Durand. Méthodes connexionnistes en parole. Actes de la l'École de printemps Neurosciences et Sciences de l'Ingénieur (NSI), 54 pp, 1995.

[buniet95b] L. Buniet, D. Fohr. Continuous speech segmentation with the gamma memory model. Proceedings of the European Conference on Speech Communication and Technology, vol. 3, pp 1685-1688, 1995.

[buniet96] L. Buniet, D. Fohr, J.-M. Pierrel. Mise en oeuvre des réseaux de neurones gamma pour la segmentation de la parole continue. Actes des Journées d'Étude sur la Parole (XXIèmes JEP), pp 309-312, 1996.

[burnod88] Y. Burnod. An adaptive neural network: the cerebral cortex. Masson, 367 pp, 1988.

[burr92] D. J. Burr. Comparison of gaussian and neural network classifiers on vowel recognition using the discrete cosine transform. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 365-368, 1992.

[burrows93] T. L. Burrows et M. Niranjan. The use of feed-forward and recurrent neural networks for system identification. Rapport technique CUED/F-INFENG/TR158 du département d'ingénirie de l'Université de Cambridge, Cambridge (Angleterre), 17 pp, 1993.

[bush87] M. A. Bush et G. E. Kopec. Network-based connected digit recognition. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 5, no 10, pp 1401-1413, 1987.

[caianiello61] E. R. Caianiello. Outline of a theory of thought processes and thinking machines. Journal of Theorical Biology, vol. 1, pp 204-235, 1961.

[calliope89] Calliope (ouvrage collectif). La parole et son traitement automatique. Collection technique et scientifique des télécommunications, CNET - ENST, Masson, 718 pp, 1989.

[cammarata95] G. Cammarata, D. Fichera, S. Graziani et L. Marletta. Fuzzy logic for urban traffic noise prediction. Journal of the Acoustical Society of America, vol. 98, no 5, pp 2607–2612, 1995.

[carey91] M. J. Carey et E. S. Parris. Adapting input transformations using alpha-nets for whole word speech recognition. Proceedings of the European Conference on Speech Communication and Technology, pp 555-558, 1991.

[carpenter88] G. A. Carpenter et S. Grossberg. The ART of adaptive pattern recognition by a self organizing neural network. Computer, vol. 21, pp 77-88, 1988.

[carpinteiro96] O. A. S. Carpinteiro. A connectionnist approach in music perception. Thèse de doctorat mention informatique, 108 pp, Université du Sussex, Brighton (Angleterre), 1996.

[carre84] R. Carré, R. Descout, M. Eskenazi, J. Mariani et M. Rossi. The french language database: defining, planning and recording a large database. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 4 pp, 1984.

[carre91] R. Carré, J.-F. Dégremont, M. Gross, J.-M. Pierrel et G. Sabah. Langage humain et machine. 300 pp, Collection CNRS Plus, Presses du CNRS, Paris (France), 1991.

[carroll90] Lewis Carroll. Œuvres. 1980 pp, Bibliothèque de la Pléiade, NRF, Gallimard, 1990.

[casdagli89] M. Casdagli. Nonlinear prediction of chaotic time series. Physica D, vol. 35, pp 335–356, 1989.

[casdagli91] M. Casdagli, S. Eubank, J. D. Farmer et J. Gibson. State space reconstruction in the presence of noise. Physica D, vol. 51, pp 52–98, 1991.

[casey95] M. Casey. How discrete-time recurrent neural networks work. Rapport technique INC-9503, 41 pp, Institute for Neural Computation, Université de Californie à San Diego, San Diego (CA, États-Unis), 1995.

[cateau92] H. Câteau, T. Nakajima, H. Nunokawa et N. Fuchikami. Power law in the human memory and in neural network model. Rapport technique UT-616, 18 pp, Université de Tokyo, Tokyo (Japon), 1992.

[catfolis93] T. Catfolis. A method for improving the real-time recurrent learning algorithm. Neural Networks, vol. 6, pp 807-821, 1993.

[cerf69] V. Cerf. ASCII format for network interchange. Internet Request For Comment no 20 (rfc20), 6 pp, 1969. http://www.internic.net/

[celebi94] S. Celebi et J. C. Principe. Analysis of spectral feature extraction using the gamma filter. Proceedings of the IEEE International Conference on Neural Networks, vol. 7, pp 4497-4501, 1994.

[celebi95a] S. Celebi. Representations of locally stationary signals using lowpass moments. Thèse de doctorat mention informatique, 156 pp, Université de Floride à Gainesville, Gainesville (FL, États-Unis), 1995.

[celebi95b] S. Celebi et J. C. Principe. Parametric least squares approximation using gamma bases. IEEE Transactions on Signal Processing, vol. 43, no 3, pp 781-784, 1995.

[celebi95c] S. Celebi et J. C. Principe. Magnitude spectral estimation via Poisson moments with application to speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 393-396, 1995.

[chalmers91] D. J. Chalmers, R. M. French et D. R. Hofstadter. High-level perception, representation and analogy: a critique of artificial intelligence methodology. Rapport technique CRCC-49, 32 pp, Université de l'Indiana, Bloomington (IN, États-Unis), 1991.

[changeux89] J.-P. Changeux et S. Dehæne. Neuronal models of cognitive functions. Cognition, vol. 33, pp 63-109, 1989.

[changeux96] J.-P. Changeux et S. Dehæne. Neuronal models of cognitive functions associated with the prefrontal cortex. [damasio96a], pp 125-144.

[chappelier94] J.-C. Chappelier et A. Grumbach. Time in neural networks. SIGART Bulletin, vol. 5, no 3, pp 3–11, 1994.

[chappell93] G. J. Chappell et J. G. Taylor. The temporal Kohonen map. Neural Networks, vol. 6, pp 441-445, 1993.

[chiba90] S. Chiba et K. Asai. A new method of consonant detection and classification using neural networks. Proceedings of the International Conference on Spoken Language Processing, pp 1065-1068, 1990.

[cho90] Y. D. Cho, K. C. Kim, H. S. Yoon, S. R. Meng et J. W. Cho. Extended Elman's recurrent neural network for syllable recognition. Proceedings of the International Conference on Spoken Language Processing, pp 1057-1060, 1990.

[chomsky57] N. Chomsky. Syntactic structures. 118 pp, Montoun, 1957.

[churchland96] P. S. Churchland. Feeling reasons. [damasio96a], pp 181-199.

[cohn91] R. P. Cohn. Robust voiced/unvoiced speech classification using a neural net. Proceedings of the International Conference on Acoustic, Speech and Signal Processing, pp 437-440, 1991.

[cole92] R. A. Cole et Y. K. Muthusamy. Perceptual studies on vowels excised from continuous speech. Proceedings of the International Conference on Spoken Language Processing, pp 1091-1093, 1992.

[colla85] A. M. Colla et D. Sciarra. Automatic generation of linguistic, phonetic and acoustic knowledge for a diphone-based continuous speech recognition system. NATO ASI Series, vol. F16 : New systems and architectures for automatic speech recognition and synthesis, 628 pp, Springer-Verlag, 1985.

[comper90] D. van Compernolle, F. Xie et M. van Diest. Speech recognition in noisy environments with the aid of microphone arrays. Speech Communication, vol. 9, no 5, pp 433–442, 1990.

[comper95] D. van Compernolle. Speech enhancement. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 10.3, pp 380-384, 1995.

[connor94] J. T. Connor, R. D. Martin et L. E. Atlas. Recurrent neural networks and robust time series prediction. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 240-254, 1994.

[cooke93] M. Cooke, S. Beet et M. Crawford Éds. Visual representations of speech signals. 385 pp, Collection Wiley professional computing, John Wiley and Sons,1993.

[copelli96] M. Copelli, R. Eichhorn, O. Kinouchi, M. Biehl, R. Simonetti, P. Riegler et N. Caticha. Noise robustness in multilayer neural networks. Rapport technique WUE-ITP-96-022, 8 pp, Institut de Physique Théorique, Université de Würzburg, Würzburg (Allemagne), 1996.

[cottrell91] G. W. Cottrell et K. Plunkett. Learning the past tense in a recurrent network: acquiring the mapping from meaning to sounds. Cognitive Science, vol. 15, 6 pp, 1991.

[crick91] F. Crick et C. Koch. Towards a neurobiological theory of conciousness. Rapport technique CNS Memo 9, 23 pp, Computation and Neural Systems program, California Institute of Technology, Pasadena (CA, États-Unis), 1991.

[cummins93] F. Cummins. Representation of temporal patterns in recurrent networks. Proceedings of the 15th Annual Conference of the Cognitive Science Society, 6 pp, 1993.

[cummins94] F. Cummins, R. F. Port, J. D. McAuley et S. Anderson. Temporal pattern recognition with fully recurrent networks. Rapport technique no 93, 14 pp, Cognitive Science Program, Université de l'Indiana, Bloomington (IN, États-Unis), 1994.

[cybenko89] G. Cybenko. Approximation by superposition of a sigmoidal function. Mathematics of Control, Signals and Systems, no 2, pp 303-314, 1989.

[daly88] N. A. Daly. Acoustic and lexical study of the spelling task. Speech communication group working papers, vol. 6, pp 136-161, Reserarch Laboratory of Electronics, Massachussets Institute of Technology, Cambridge (MA, États-Unis), 1988.

[damasio94] A. R. Damasio. Descartes's error - Emotion, reason, and the human brain. Putnam, 1994. Traduction française : "L'erreur de Descartes : la raison des émotions". 365 pp, Éditions Odile Jacob, 1995.

[damasio96a] A. R. Damasio, H. Damasio et Y. Christen. Neurobiology of decision-making. 208 pp, Collection Research and perspectives in neurosciences, Spinger-Verlag, 1996.

[damasio96b] H. Damasio. Human neuroanatomy relevant to decision-making. [damasio96a], pp 1-12.

[dancer92] A. Dancer, T. Vaillant et P. Borredon. Environnement acoustique et opérateur humain. Actes des entretiens Science et Défense, vol. 2, pp 93-110, 1992.

[dancer95a] A. Dancer, D. Plessiet et T. Vaillant. Les limites d'exposition aux bruits dans les armées. Établissement Technique de Bourges, Journées du Comité Bruits d'Armes, Bourges (France), 1995. DGA CR 241/CT/DTB/1995. cité dans [dancer95b].

[dancer95b] A. Dancer et R. Franke. Hearing hazard from impulse noise: a comparative study of two classical criteria for weapon noises (Pfander criterion and Smoorenburg criterion) and the Laeq8 method. Acta Acustica, vol. 3, no 6, pp 539, 1995.

[davis80] S. B. Davis et P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 28, no 4, pp 357-366, 1980.

[dehæne87] S. Dehæne, J.-P. Changeux et J.-P. Nadal. Neural networks that learn temporal sequences by selection. Proceedings of the National Academy of Sciences of the USA, vol. 84, pp 2727-2731, 1987.

[dehæne89] S. Dehæne, J.-P. Changeux. A simple model of prefrontal cortex function in delayed-response tasks. Journal of Cognitive Neuroscience, vol. 1, pp 244-261, 1989.

[dempster77] A. P. Dempster, N. M. Laird et D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, vol. 39, no 1, pp 1-38, 1977.

[deng94] L. Deng et D. X. Sun. A statistical approach to automatic speech recognition using the atomic speech units constructed from overlapping articulatory features. Journal of the Acoustical Society of America, vol. 95, no 5, pt. 1, pp 2702-2719, 1994.

[dennis94] S. J. Dennis. The integration of learning into models of human memory. Thèse de doctorat mention informatique, 196 pp, Université du Queensland, St Lucia (Australie), 1994.

[depuydt90] L. Depuydt, J.-P. Martens, L. van Immerseel et N. Weymære. Improved broad phonetic classification and segmentation with a neural network and a new auditory model. Proceedings of the International Conference on Spoken Language Processing, pp 1041-1044, 1990.

[dermody92] Phillip Dermody. Human capabilities for speech processing in noise. ESCA Technical Research Workshop: Speech processing in adverse conditions, pages 11–19, 1992.

[derou94] D. Derou et L. Héraut. Pulsed neural network and perceptive grouping. Proceedings of the European Conference on Computer Vision, 6 pp, 1994.

[dersch63] W. C. Dersch. Speech operates safety switch - Speech recognition circuit switches off machinery on command. Electronics, no 6, pp 78-82, 1963.

[digalakis95] V. Digalakis et L. Neumeyer. Speaker adaptation using combined transformation and bayesian methods. Proceedings of the International Conference on Acoustic, Speech and Signal Processing, pp 680-683, 1995.

[dimartino87] J. Di Martino. On multi-level machines for continuous speech recognition. Proceedings of the International Joint Conference on Artificial Intelligence, pp 836-839, 1987.

[dimartino93] J. Di Martino. A cognitive machine for automatic speech recognition. Rapport technique CRIN 93-R-084, 73 pp, Centre de Recherche en Informatique de Nancy, Nancy (France), 1993.

[dimartino94] J. Di Martino, J. F. Mari, B. Mathieu, K. Perot, K. Smaili. Which model for future speech recognition systems: hidden Markov models or finite-state automata?. Proceedings of the International Conference on Acoustic, Speech and Signal Processing, 4 pp, 1994.

[dingle93] A. A. Dingle, J. H. Andreæ et R. D. Jones. A chaotic neural unit. Proceedings of the IEEE World Congress on Computational Intelligence, pp 335-340, 1993.

[djezzar95] L. Djezzar. Contribution à l'étude acoustico-perceptive des occlusives du français. Thèse de doctorat mention informatique, 151 pp, Université Henri Poincaré - Nancy 1, Nancy (France), 1995.

[dobler92] S. Dobler, P. Meyer et H. W. Ruehl. A robust connected-words recognizer. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 245-248, 1992.

[dod93] U.S. Department of Defense. LPC-10 2400 bps voice coder. Release 1.0, 1993. (fichier readme.txt du package).

[dogaru95] R. Dogaru et A. T. Murgan. Chaotic resonance theory, a new approach for pattern storage and retrieval in neural networks. Proceedings of the International Conference on Neural Networks, vol. 6, pp 3048-3052, 1995.

[dore90] M. Doré. Application de réseaux neuronaux à la reconnaissance de caractères manuscrits. Rapport de La Poste, no RA3354-RVA/90/12, 89 pp, 1990.

[dorizzi92] B. Dorizzi, E. Guigon et Y. Burnod. A neural network model of frontal cortical circuits for learning conditional sequences. Proceedings of the IEEE International Conference on Neural Networks, vol. 2, pp 63-68, 1992.

[dowling86] W. J. Dowling et D. L. Harwood. Music cognition. Academic Press, 1986.

[dreher57] J. Dreher et J. O'Neill. Effects of ambient noise on speaker intelligibility for words and phrases. Journal of the Acoustical Society of America, vol. 29, pp 1320–1323, 1957.

[dtat83] Direction Terrestre des Armements Terrestres. Recommandation on evaluating the possible harmful effects of noise on hearing, AT-83/27/28, 14 pp, 1983. (présentation du LAeq8(90dB)).

[dubois91] D. Dubois. Comparison of time-dependent acoustic features for for a speaker-independent speech recognition system. Proceedings of the European Conference on Speech Communication and Technology, pp 935-938, 1991.

[ducassou91a] D. Ducassou. Cours d'acoustique. Cours de 2ème année de médecine, Université de Nancy 1, 1991.

[ducassou91b] D. Ducassou. Cours de psychoacoustique : les phénomènes subjectifs de l'audition. Cours de 2ème année de médecine, Université de Nancy 1, 1991.

[ducassou91c] D. Ducassou. Cours sur les organes des sens. Cours de 2ème année de médecine, Université de Nancy 1, 1991.

[durand95] S. Durand. TOM, une architecture connexionniste de traitement de séquences - Application à la reconnaissance de la parole. Thèse de doctorat mention informatique, 167 pp, Université Henri Poincaré - Nancy 1, Nancy (France), 1995.

[eckmann85] J.-P. Eckmann et D. Ruelle. Ergodic theory of chaos and strange attractors. Reviews of Modern Physics, vol. 57, no 3, pt. 1, pp 617–656, 1985.

[eckmann86] J.-P. Eckmann, S. O. Kamphorst, D. Ruelle et S. Ciliberto. Lyapunov exponents from time series. Journal of Physical Review A, vol. 34, no 6, pp 4971–4979, 1986.

[edelman78] G. M. Edelman. Group selection and phasic reentrant signaling: a theory of higher brain function. In: The mindful brain, pp 51-100, MIT Press, 1978.

[elenius91] K. Elenius et G. Takacs. Phoneme recognition with an artificial neural network. Proceedings of the European Conference on Speech Communication and Technology, pp 121-124, 1991.

[elman88] J. L. Elman et D. Zipser. Learning the hidden structure of speech. Journal of the Acoustical Society of America, vol. 83, pp 1615-1626, 1988.

[elman90] J. L. Elman. Finding structure in time. Cognitive Science, vol. 14, pp 179-211, 1990.

[english92] T. M. English et L. C. Bogges. Back-propagation training of a neural network for word spotting. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 357-360, 1992.

[escande92] P. Escande. Reconnaissance de parole par un modèle connexionniste à détection de coïncidences. Thèse de doctorat mention informatique, 202 pp, Université Paris XI Orsay (France), document LIMSI 93-09, 1992.

[fahlman90] S. E. Fahlman et C. Lebiere. The cascade-correlation learning architecture. Rapport technique CMU-CS-90-100, 13 pp, Université Carnegie Mellon, Pittsburgh (PA, États-Unis), 1990.

[fairbanks58] G. Fairbanks. Test of phonetic differentiation: the rhyme test. Journal of the Acoustical Society of America, vol. 30, pp 596–600, 1958.

[fallside90] F. Fallside, H. Lucke, T. P. Marsland, P. J. O'Shea, M. St J. Owen, R. W. Prager, A. J. Robinson et N. H. Russel. Continuous speech recognition for the TIMIT database using neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal, pp 445-448, 1990.

[fanty90] M. Fanty et R. Cole. Speaker-independent english alphabet recognition: experiments with the E-set. Proceedings of the International Conference on Spoken Language Processing, pp 1361-1364, 1990.

[fels94] S. S. Fels. Glove-Talk II: mapping hand gestures to speech using neural networks - An approach to building adaptive interfaces. Thèse de doctorat mention informatique, 78 pp, Université de Toronto, Toronto (Canada), 1994.

[feng91] G. Feng, N. Achab et P. Combescure. On-line speech segmentation using adaptive models: application to variable rate speech coding. Proceedings of the European Conference on Speech Communication and Technology, pp 705-708, 1991.

[fessant94] F. Fessant. La prédiction de l'activité solaire par perceptrons multicouches - Utilisation en télécommunications. Journées de l'ACTH, 8 pp, 1994.

[flake93] G. W. Flake, G.-Z. Sun, Y.-C. Lee et H.-H. Chen. Exploiting chaos to control the future. Rapport technique, 13 pp, Institute for Advance Computer Studies, Université du Maryland, College Park (MD, États-Unis), 1993.

[flandrin93] P. Flandrin. Temps-fréquence. 394 pp, Traité des nouvelles technologies, série Traitement du signal, Hermès, 1993.

[foucault13] M. Foucault. Annales de psychologie, vol. 19, p 218, 1913. Cité dans [cateau92].

[franzini89] M. A. Franzini et al. A connectionnist approach to continuous speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 425-428, 1989.

[franzini92] M. A. Franzini. A new connectionist architecture for word spotting. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 361-364, 1989.

[frasconi92] P. Frasconi, M. Gori et G. Soda. Local feedback multilayered networks. Neural Computation, vol. 4, pp 120-130, 1992.

[frasconi93] P. Frasconi, M. Gori et G. Soda. Injecting nondeterministic finite state automata into recurrent neural networks. Rapport technique, 40 pp, Université de Florence, Florence (Italie), 1993.

[fraser86] A. M. Fraser et H. L. Swinney. Independent coordinates for strange attractors from mutual information. Journal of Physical Review A, vol. 33, no 2, pp 1134-1140, 1986.

[freeman87] W. J. Freeman. Simulation of chaotic EEG patterns with a dynamic model of the olfactory system. Biological Cybernetics, vol. 56, pp 139-150, 1987.

[french47] N. R. French et J. C. Steinberg. Factors governing the intelligibility of speech sounds. Journal of the Acoustical Society of America, vol. 19, pp 90–119, 1947.

[fry95] R. L. Fry. Rational neural models based on information theory. Proceedings of the 15th International Workshop on Maximum Entropy and Bayesian Methods, 6 pp, 1995.

[furui81] S. Furui. Cepstral analysis technique for automatic speaker verification. IEEE Transations on Acoustics, Speech and Signal Processing, vol. 29, pp 254-272, 1981.

[gagnon92] L. Gagnon. A noise reduction approach for non-stationary additive interference. ESCA Technical Research Workshop: Speech processing in adverse conditions, pages 139–142, 1992.

[gales93] M. J. F. Gales et S. J. Young. HMM recognition in noise using parallel model combination. Proceedings of the European Conference on Speech Communication and Technology, vol. 2, pp 837–840, 1993.

[gales95] M. J. F. Gales. Model-based techniques for noise robust speech recognition. Thèse de doctorat mention informatique, 119 pp, Université de Cambridge, Cambridge (Angleterre), 1995.

[galiano91] I. Galiano, F. Casacuberta et E. Sanchis. On the structure of subword units for a speaker independent continuous speech task. Proceedings of the European Conference on Speech Communication and Technology, pp 675-678, 1991.

[gat78] I. Gat et R. Keith. An effect of linguistic experience. Audiology, vol. 17, pp 339–345, 1978.

[geva92] S. Geva et J. Sitte. A constructive method for multivariate function approximation by multilayer perceptrons. IEEE Transactions on Neural Networks, vol. 3, no 4, pp 621-631, 1992.

[ghiselli91] T. Ghiselli-Crippa et A. El-Jaroudi. A fast neural net training algorithm and its application to voiced-unvoiced-silence classification of speech. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 441-444, 1991.

[giles94a] C. L. Giles, G. M. Kuhn et R. J. Williams. Dynamic recurrent neural networks: theory and applications. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 153-156, 1994.

[giles94b] C. L. Giles, B. G. Horne et T. Lin. Learning a class of large finite state machines with a recurrent neural network. Rapport technique UMIACS-TR-94-94, 24 pp, Institute for Advanced Computer Studies, Université du Maryland, College Park (MD, États-unis), 24 pp, 1994.

[giles95] C. L. Giles et C. W. Omlin. Learning, representation and synthesis of discrete dynamical systems in continuous recurrent neural networks. Proceedings of the IEEE Workshop on Architectures for Semiotic Modeling and Analysis in Large Complex Systems, 8 pp, 1995.

[goh91] T. H. Goh, P. Z. Wang et H. C. Lui. Learning algorithm for the enhanced fuzzy perceptron. Rapport technique, 6 pp, Université Nationale de Singapoure (Singapoure), 1991.

[gong91] Y. Gong et J.-P. Haton. Comparing two phoneme identification methods using a continuous speech recognizer. Proceedings of the European Conference on Speech Communication and Technology, pp 417-420, 1991.

[gong93] Y. Gong et W. C. Treurniet. Speech recognition in noisy environments: a survey. Rapport technique CRC-TN-93-002, 35 pp, Communication Research Center, Toronto (Canada), 1993.

[gong95] Y. Gong. Speech recognition in noisy environments: a survey. Speech Communication, vol. 16, no 3, pp 261–291, 1995.

[gori89] M. Gori, Y. Bengio, R. de Mori. BPS: A learning algorithm for capturing the dynamic nature of speech. Proceedings of the International Joint Conference on Neural Networks, vol. 2, pp 417-423, 1989.

[greco91] F. Greco, A. Paoloni et G. Ravaioli. A recurrent time-delay neural network for improved phoneme recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 81-84, 1991.

[grossberg83] S. Grossberg et M. Cohen. Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. IEEE Transcactions on Systems, Man and Cybernetics, vol. 13, pp 815-826, 1983.

[grumbach92] A. Grumbach. Percept, concept, symbole, artificiels. Actes de la Conférence Neurosciencès et Sciences de l'Ingénieur, 4 pp, 1992.

[gu91] H. Gu, C. Tseng et L. Lee. Isolated-utterance speech recognition using hidden Markov models with bounded state duration. IEEE Transactions on Signal Processing, vol. 39, no 8, pp 1743-1752, 1991.

[gupta91] V. N. Gupta, M. Lennig, P. Mermelstein, P. Kenny, F. Seitz et D. O'Shaughnessy. Using phoneme duration and energy contour information to improve large vocabulary isolated-word recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 341-344, 1991.

[guyot90] F. Guyot. Une modélisation fonctionnelle du cortex : la colonne corticale - Aspect auditifs et visuels. Thèse de doctorat mention informatique, 180 pp, Université de Nancy 1, Nancy (France), 1990.

[haffner91a] P. Haffner et A. H. Waibel. Time-delay neural network embedding time alignment: a performance analysis. Proceedings of the European Conference on Speech Communication and Technology, pp 1415-1418, 1991.

[haffner91b] P. Haffner, M. Franzini et A. H. Waibel. Integrating time alignment and neural networks for high performance continuous speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 105-108, 1991.

[haffner92a] P. Haffner et A. H. Waibel. Multi-state time delay neural network for continuous speech recognition. Advances in Neural Information Processing Systems, pp 135-142, 1992.

[haffner92b] P. Haffner. Connectionnist word-level classification in speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 621-624, 1991.

[hansen90] L. K. Hansen P. Salamon. Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no 10, pp 993-1001, 1990.

[hanson96] C. Hanson et S. J. Hanson. Development of schemata during event parsing: Neisser's perceptual cycle as a recurrent connectionnist network. Journal of Cognitive Neuroscience, vol. 8, no 2, pp 119-134, 1996.

[harnad90] S. Harnad. The symbol grouding problem. Physica D, vol. 42, pp 335-346, 1990.

[harris78] F. J. Harris. On the use of windows for harmonic analysis with the discrete Fourier transform. Proceedings of the IEEE, vol. 66, no 1, pp 51-83, 1978.

[harrison89] T. D. Harrison et F. Fallside. A connectionnist model for phoneme recognition in continuous speech. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 417-420, 1989.

[hataoka90] N. Hataoka, A. Amano, T. Aritsuka et A. Ichikawa. Large vocabulary speech recognition using neural-fuzzy and concept networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 513-516, 1990.

[hataoka91] N. Hataoka et A. H. Waibel. Evaluation of speaker-independent phoneme recognition on TIMIT database using TDNNs. Proceedings of the European Conference on Speech Communication and Technology, pp 105-108, 1991.

[haton91] J.-P. Haton, J.-M. Pierrel, G. Pérennou, J. Caelen et J.-L. Gauvain. Reconnaissance automatique de la parole. 239 pp, Collection AFCET - Dunod informatique, Dunod, 1991.

[hattori92] H. Hattori. Text-idependent speaker recognition using neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 153-156, 1992.

[haykin91] S. Haykin. Adaptive filter theory. Prentice Hall, 1991.

[hebb49] D. O. Hebb. The organization of behavior. John Wiley and Sons, 1949.

[hermansky85] H. Hermansky, B. A. Hanson et H. Wakita. Low-dimensional representations of vowels based on all-pole modeling in the psychophysical domain. Speech Communication, vol. 4, pp 181-187, 1985.

[hermansky90] H. Hermansky. Perceptual linear predictive analysis of speech. Journal of the Acoustical Society of America, vol. 87, no 4, pp 1738-1752, 1990.

[hermansky91a] H. Hermansky, N. Morgan, A. Bayya et P. Kohn. Compensation for the effect of the communication channel in auditory-like analysis of speech (RASTA-PLP). Proceedings of the European Conference on Speech Communication and Technology, pp 1367-1370, 1991.

[hermansky91b] H. Hermansky, N. Morgan, A. Bayya et P. Kohn. RASTA-PLP speech analysis. Rapport technique TR-91-069, 6 pp, International Computer Science Institute, Berkeley (CA, États-Unis), 1991.

[hermansky92] H. Hermansky, N. Morgan, A. Bayya et P. Kohn. RASTA-PLP speech analysis technique. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 121-124, 1992.

[hermansky94] H. Hermansky et N. Morgan. RASTA processing of speech. IEEE Transactions on Speech and Audio Processing, vol. 2, no 4, pp 578–589, 1994.

[hermansky95a] H. Hermansky, S. Greenberg et M. Pavel. A brief (100-200 ms) history of time in feature extraction for speech (temporal processing for speech signal). Proceedings of the 15th Annual Speech Research Symposium, John Hopkins University, Baltimore (MD, États-Unis), 8 pp, 1995.

[hermansky95b] H. Hermansky. Exploring temporal domain for robustness in speech recognition. Proceedings of the 15th International Congress on Acoustics, pp 61-64, 1995.

[hernando94] J. Hernando et C. Nadeu. Speech recognition in noisy car environment based on OSALPC representation and robust similarity measuring techniques. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 91-94, 1994.

[hewitt90] M. J. Hewitt et R. Meddis. Implementation details of a computation model of the inner hair-cell/auditory nerve synapse. Journal of the Acoustical Society of America, vol. 87, no 4, pp 1813-1816, 1990.

[hihi96] S. El Hihi et Y. Bengio. Hierarchical recurrent neural networks for long-term dependencies. Advances in Neural Information Processing Systems, 8 pp, 1996.

[hinton84] G. E. Hinton, T. J. Sejnowski et D. H. Ackley. Boltzmann machines: constraint satisfaction networks that learn. Rapport technique TR-CMU-CS-84-199, Université Carnegie Mellon, Pittsburgh (PA, États-Unis), 1984. Cité dans [ackley85].

[hkp91] J. A. Hertz, A. S. Krogh et R. G. Palmer. Introduction to the theory of neural computation. Lecture Notes in the Santa Fe Institute Studies in the Sciences of Complexity, vol. 1, 352 pp, Addison-Wesley, 1991.

[hochreiter95] S. Hochreiter et J. Schmidhuber. Long short term memory. Rapport technique FKI 207-95, 8 pp, Université Technique de Munich & IDSIA, Munich (Allemagne) & Lugano (Suisse), 1995.

[hodgkin52] A. J. Hodgkin et A. F. Huxley. Journal of Physiology, vol. 117, pp 500, 1952.

[hoehfeld91] M. Hoehfeld et S. E. Fahlman. Learning with limited numerical precision using the cascade-correlation algorithm. Rapport technique CMU-CS-91-130, 17 pp, Université Carnegie Mellon, Pittsburg (PA, États-Unis), 1991.

[hogg90] T. Hogg et B. A. Huberman. Controlling chaos in distributed systems. Rapport technique, 21 pp, Xerox PARC, Palo Alto (CA, États-Unis), 1990.

[holmes86] J. N. Holmes et N. C. Sedgwick. Noise compensation for speech recognition with degraded and undegraded speech. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 741-744, 1986.

[hopfield82] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences, États-Unis, vol. 79, pp 2554–2558, 1982.

[horn91] D. Horn et M. Usher. Parallel activation of memories in oscillatory neural network. Neural Computation, vol. 3, no 1, pp 31-43, 1991.

[horne95] B. G. Horne et C. L. Giles. An experimental comparison of recurrent neural networks. Advances in Neural Information Processing Systems, pp 697-705, 1995.

[hubbard95] B. B. Hubbard. Ondes et ondelettes - La saga d'un outil mathématique. 235 pp, Collection Sciences d'avenir, Éditions Belin - Pour La Science, 1995.

[hunt78] E. Hunt. Mechanics of verbal ability. Psychological Review, vol. 85, pp 109-130, 1978.

[hunt89] M. J. Hunt et C. Lefebre. A comparison of several acoustic representations for speech recognition with degraded and undegraded speech. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 262-265, 1989.

[hunt95] M. J. Hunt. A robust method of detecting the presence of voiced speech. Proceedings of the 15th International Congress on Acoustics, pp 65-68, 1995.

[hunt96] M. J. Hunt. Real speech recognition technology doesn't need hype. Voice +, vol. 3, no 3, pp 35-37, 1996.

[hush93] D. R. Hush et B. G. Horne. Progress in supervised neural networks - What's new since Lippmann? IEEE Signal Processing Magazine, no 1, pp 8-39, 1993.

[husoy91] P. O. Husøy et T. Svendsen. ANN-based speech recognition using a preprocessor for non-linear time compression. Proceedings of the European Conference on Speech Communication and Technology, pp 563-566, 1991.

[imbert83] M. Imbert. La neurobiologie de l'image. La recherche, vol. 14, no 144, pp 600-613, 1983.

[ingber81] L. Ingber. Attention, physics and teaching. Journal Social Biological Structures, vol. 4, pp 225-235, 1981. Disponible sur www.ingber.com (!).

[ingber82] L. Ingber. Statistical mechanics of neocortical interactions - Basic formulation. Physica D, vol. 5, pp 83-107, 1982.

[ingber95a] L. Ingber. Statistical mechanics of neocortical interactions: high resolution path-integral calculation of short-term memory. Journal of Physical Review E, vol. 51, no 5, 1995.

[ingber95b] L. Ingber. Statistical mechanics of neocortical interactions: constraints on 40 Hz models of short-term memory. Journal of Physical Review E, vol. 52, no 4, pp 4561-4563, 1995.

[iso90] K.-I. Iso et T. Watanabe. Speaker-independent word recognition using a neural prediction model. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 441-444, 1990.

[iso91] K.-I. Iso et T. Watanabe. Large vocabulary speech recognition using a neural prediction model. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 57-60, 1991.

[iso1999] International Standard Organisation. Acoustics - Determination of occupational noise exposure and estimation of noise-induced hearing impairment - ISO R-1999, Genève (Suisse), 1990.

[iso3741] International Standard Organisation. Acoustics - Determination of sound power levels of noise sources - ISO 3741, Genève (Suisse), 1975.

[iso3742] International Standard Organisation. Acoustics - Determination of sound power levels of noise sources - ISO 3742, Genève (Suisse), 1975.

[iso3745] International Standard Organisation. Acoustics - Determination of sound power levels of noise sources - ISO 3745, Genève (Suisse), 1977.

[itakura75] F. Itakura. Minimum production residual principle applied to speech recognition. IEEE Transaction on Acoustics, Speech an Signal Processing, vol. 23, pp 67-72, 1975.

[james81] W. James. The principles of psychology. Harvard University Press, 1981.

[jerison55] H. J. Jerison et A. K. Smith. Effect of acoustic noise on time judgment. Technical Report WADC-TR-55-358, AD 99641, US Air Force, Wright Air Development Center, Wright-Patterson Air Force Base, Ohio, États-Unis, 1955. Cité dans [kryter70].

[jerison57] H. J. Jerison. Performance on a simple vigilance task in noise and quiet. Journal of the Acoustical Society of America, vol. 29, no 11, pp 1163- 1165, 1957.

[jerison59] H. J. Jerison. Effects of noise on human performance. Journal of Applied Psychology, vol. 43, no 2, pp 96-101, 1959.

[jervis93] T. T. Jervis et W. J. Fitzgerald. Optimization schemes for neural networks. Rapport technique CUED/F-INFENG/TR144, 33 pp, Université de Cambridge, Cambridge (Angleterre), 1993.

[jodouin90] J.-F. Jodouin. Présentation des modèles connexionnistes. Intellectica, nos 9-10, pp 9-39, 1990.

[jodouin93] J.-F. Jodouin. Putting the simple recurrent network to the test. Proceedings of the IEEE International Conference on Neural Networks, pp 1141-1146, 1993.

[jordan94a] M. I. Jordan. Forward models: supervised learning with a distal teacher. Rapport technique occasional paper no 40, 51 pp, Center for Cognitive science, Massachussets Institute of Technology, Cambridge (MA, États-Unis), 1994.

[jordan94b] M. I. Jordan. Lectures on neural networks. Manuel de cours, 104 pp, École d'Été d'Analyse Numérique CEA-INRIA-EDF Réseaux de Neurones et Applications, 1994.

[jordan86] M. I. Jordan. Attractor dynamics and parallelism in a connectionnist sequential machine. Proceedings of the Eighth Conference of the Cognitive Science Society, pp 531-546, 1986.

[junqua92] J.-C. Junqua. The variability of speech produced in noise. ESCA Technical Research Workshop: Speech processing in adverse conditions, pp 43–52, 1992.

[junqua94a] J.-C. Junqua. A duration study of speech vowels produced in noise. Proceedings of the International Conference on Spoken Language Processing, pp 419-422, 1994.

[junqua94b] J.-C. Junqua, B. Mak et B. Reaves. A robust algorithm for word boundary detection in the presence of noise. IEEE Transactions on Speech and Audio Processing, vol. 2, no 3, pp 406-412, 1994.

[junqua95] J.-C. Junqua et J.-P. Haton. Robustness in automatic speech recognition: fundamentals and applications. 440 pp, Kluwer Academic Publishers, 1995.

[jurik91] M. Jurik. Backpercolation: assigning local error in feedforward perceptron networks. Rapport technique, 34 pp, Jurik Research and Consulting, 1991.

[kaneko90] K. Kaneko. Clustering, coding, switching, hierarchical ordering and control in a network of chaotic elements. Physica D, vol. 41, pp 137-172, 1990.

[kangas91] J. Kangas. Phoneme recognition using time-dependant versions of self-organising maps. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 101-104, 1991.

[kangas92] J. Kangas, K. Torkkola et M. Kokkonen. Using SOMs as feature extractors for speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 341-344, 1992.

[kangas94] J. Kangas. On the analysis of pattern sequences by self-organizing maps. Thèse de doctorat de technologie, 84 pp, Université de technologie d'Helsinki, Helsinki (Finlande), 1994.

[kaplan95] R. M. Kaplan. Finite state technology. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 11.6, pp 419-422, 1995.

[kautz54] W. Kautz. Transient synthesis in the time domain. IRE Transactions on Circuit Theory, vol. 1, pp 29-39, 1954.

[keating92] P. Keating, B. Blankenship, D. Byrd, E. Flemming et Y. Todaka. Phonetic analyses of the TIMIT corpus of american english. Proceedings of the International Conference on Spoken Language Processing, pp 823-826, 1992.

[kechriotis94] G. Kechriotis, E. Zervas et E. S. Manolakos. Using recurrent neural networks for adaptive communication channel equalization. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 267-278, 1994.

kehagias91 A. Kehagias. Stochastic recurrent networks training by the local backward-forward algorithm. Rapport technique, 40 pp, département de mathématiques appliquées, Université Brown, Providence (RI, États-Unis), 1991.

[keller92] E. Keller. L'organisation temporelle de la parole. Bulletin de la Communication Parlée, no 2, pp 73-88, 1992.

[kenny91] P. Kenny, S. Parthasarathy, V. N. Gupta, M. Lennig, P. Mermelstein et D. O'Shaughnessy. Energy, duration and Markov models. Proceedings of the European Conference on Speech Communication and Technology, pp 655-658, 1991.

[kim92] S. Kim et M. B. Waldron. Spatiotemporal neural network using axodendritic chemical synapse model. Proceedings of the IEEE International Joint Conference on Neural Network, vol. 1, pp 389-394, 1992.

[klatt79] D. H. Klatt. A digital filterbank for spectral matching. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 573-576, 1979.

[kobayashi84] T. Kobayashi et S. Imai. Spectral analysis using generalised cepstrum. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 35, no 5, pp 1087-1089, 1984.

[kobayashi91] T. Kobayashi, M. Yagyu et K. Shirai. Application of neural networks to articulatory motion estimation. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 489-492, 1991.

[kohonen72] T. Kohonen. Correlation matrix memories. IEEE Transactions on Computers, vol. C21, no 4, pp 353-359, 1972.

[kohonen84] T. Kohonen, K. Mäkisara et T. Saramäki. Phonotopic maps - Insightful representation of phonological features for speech recognition. Proceedings of the IEEE International Conference on Pattern Recognition, pp 182-185, 1984.

[kohonen87] T. Kohonen. Self-organisation and associative memory. 312 pp, Springer series in information sciences, vol. 8, Springer-Verlag, Berlin (Allemagne), 2ème édition, 1987.

[kohonen88] T. Kohonen. The "neural" phonetic typewriter. IEEE Computer Magazine, no 3, pp 11-22, 1988.

[kohonen93] T. Kohonen. Things you haven't heard about the self-organising map. Proceedings of the IEEE International Conference on Neural Network, pp 1147-1156, 1993.

[koizumi94] T. Koizumi, S. Taniguchi, K.-I. Hattori et M. Mori. Simplified sub-neural-networks for accurate phoneme recognition. Proceedings of the International Conference on Spoken Language Processing, vol. 3, pp 1571-1574, 1994.

[kokar93] M. M. Kokar et S. A. Reveliotis. Reinforcement learning: architectures and algorithms. International Journal of Intelligent Systems, vol. 8, pp 875-894, 1993.

[kolen94] J. F. Kolen. Exploring the computational capabilities of recurrent neural networks. Thèse de doctorat mention informatique, 165 pp, Université d'état de l'Ohio, Columbus (OH, États-Unis), 1994.

[komori90] Y. Komori, K. Hatazaki, T. Tanaka et T. Kawabata. Combining phoneme identification neural networks into an expert system using spectrogram reading knowledge. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 505-508, 1990.

[komori91a] Y. Komori. Time-state neural networks for phoneme identification by considering temporal structure of phonemic features. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 125-128, 1991.

[komori91b] Y. Komori et K. Hatazaki. An integration of knowledge and neural networks toward a phoneme typewriter without a language model. Proceedings of the European Conference on Speech Communication and Technology, pp 1423-1426, 1991.

[komori92] Y. Komori. A neural fuzzy training approach for continuous speech recognition improvment. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 405-408, 1992.

[kowalski86] R. A. Kowalski et M. J. Sergeot. A logic-based calculus of events. New Generation Computing, vol. 4, pp 67-95, 1986.

[kryter70] K. D. Kryter. The effects of noise on man. 632 pp, Academic Press, 1970.

[kumar91] V. V. Kumar, S. C. Ahalt et A. K. Krishnamurthy. Phonetic to acoustic mapping using recurrent neural network. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 753-756, 1991.

[kuo93a] J.-M. Kuo. Nonlinear dynamic modeling with artificial neural networks. Thèse de doctorat mention informatique, 186 pp, Université de Floride à Gainesville, Gainesville (FL, États-Unis), 1993.

[kuo93b] J.-M. Kuo et J. C. Principe. Using the Poisson filter chain to reconstruct attractors. Proceedings of the SPIE Conference on Chaos and Non Linearities, pp 59-65, 1993.

[kuo94a] J.-M. Kuo et J. C. Principe. Noise reduction in state space using the focused gamma model. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 533-536, 1994.

[kuo94b] J.-M. Kuo et J. C. Principe. Reconstructed dynamics and chaotic time series modelling. Proceedings of the IEEE World Congress on Computational Intelligence, Orlando (FL, États-Unis), vol. 5, pp 3131-3136, 1994.

[kuo94c] J.-M. Kuo, S Celebi et J. C. Principe. Adaptation of memory depth in the gamma filter. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal, Adélaide (Australie), pp 373-376, 1994.

[laine94] A. Lainé et D. Béroule. Vers une perception active pour la reconnaissance automatique de la parole continue. Actes des Journées Neurosciences et Sciences de l'Ingénieur, pp 23-26, 1994.

[lang90] K. J. Lang, A. H. Waibel et G. E. Hinton. A time-delay neural network architecture for isolated word recognition. Neural Networks, vol. 3, pp 23-43, 1990.

[large95a] E. W. Large, C. Palmer et J. B. Pollack. Reduced memory representations for music. Cognitive science, vol. 19, no 1, pp 53–96, 1995.

[large95b] E. W. Large, J. F. Kolen. Resonance and the perception of musical meter. Connection Science, vol. 6, no 1, pp 177-208, 1995.

[lashley51] K. S. Lashley. The problem of serial order in behavior. In: Cerebral mechanisms in behavior: the Hixon symposium, Wiley, 1951.

[laubsch79] J. H. Laubsch. Interfacing a semantic net with an augmented transition network. Proceedings of the International Joint Conference on Artificial Intelligence, pp 516-518, 1979. Cité dans [pierrel81].

[lawrence95] S. Lawrence, S. Fong, C. L. Giles. On the applicability of neural network and machine learning methodologies to natural language processing. International Joint Conference on Artificial Intelligence, 8 pp, 1995.

[lawrence96] S. Lawrence, A. C. Tsoi et A. D. Back. The gamma MLP for speech phoneme recgnition. Advances in Neural Information Processing Systems, 7 pp, 1996.

[lecun85] Y. Le Cun. Une procédure d'apprentissage pour réseau à seuil asymétrique. Actes de la conférence Cognitiva, pp 599-604, 1985.

[lecun89a] Y. Le Cun, L. D. Jackel, B. Boser, J. S. Denker, H. P. Graf, I. Guyon, D. Henderson, R. E. Howard, W. Hubbard. Handwritten digit recognition: applications of neural network chips and automatic learning. IEEE Communications Magazine, no 11, pp 41-46, 1989.

[lecun89b] Y. Le Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard et L. D. Jackel. Handwritten digit recognition with a back-propagation network. Advances in Neural Information Processing systems, pp 396-404, 1989.

[lecun94] Y. Le Cun. Efficient learning and second-order methods. Manuel de cours, 75 pp, École d'Été d'Analyse Numérique CEA-INRIA-EDF Réseaux de Neurones et Applications, 1994.

[lefebvre93] W. C. Lefebvre et J. C. Principe. Object-oriented artificial neural network implementations. Proceedings of the World Conference on Neural Network, Portland (Oregon, États-Unis), vol. 4, pp 436–439, 1993.

[leigthon91] R. R. Leighton et B. B. Conrath. The autoregressive backpropagation algortihm. Proceedings of the International Joint Conference on Neural Networks, 9 pp, 1991.

[leigthon92] R. R. Leighton. The Aspirin/Migraines neural network software, user's manual. Rapport Technique MP-91W00050, MITRE Corporation, Bedford (MA, États-Unis), 1992.

[leroi92] A. Leroi-Gourhan. Le geste et la parole ; tome 1 : technique et langage. Collection sciences d'aujourd'hui, 324 pp, Éditions Albin Michel, 1992.

[lesser75] V. R. Lesser, R. D. Fennel, L. D. Erman et D. R. Reddy. Organization of the HEARSAY-II speech understanding system. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 11–23, 1975.

[leung90] H. C. Leung et V. W. Zue. Phonetic classification using multi-layer perceptrons. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 525–528, 1990.

[levin93] A. U. Levin et K. S. Narenda. Control of nonlinear dynamical systems using neural networks: controllability and stabilization. IEEE Transactions on Neural Networks, vol. 4, no 2, pp 192-206, 1993.

[levinson86] S. E. Levinson. Continuously variable duration hidden Markov models for automatic speech recognition. Computer Speech and Language, vol. 1, no 1, pp 29-45, 1986.

[lienard90] J. S. Liénard et C. d'Alessandro. Wavelets and granular analysis of speech. In: Wavelets - Time-frequency methods and phase space, 2nd edition, pp 158-163, Springer-Verlag, 1990.

[ligozat95] G. Ligozat. Representations of space and time. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 9.2, pp 343-347, 1995.

[lim79a] J. S. Lim. Spectral root homomorphic deconvolution system. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 27, no 3, pp 223-233, 1979.

[lim79b] J. S. Lim et A. V. Oppenheim. Enhancement and bandwidth compression of noisy speech. Proceedings of the IEEE, vol. 67, pp 1586-1604, 1979.

[lin95] T. Lin, B. G. Horne, P. Tino et C. L. Giles. Learnin long-term dependencies is not so difficult with NARX recurrent neural networks. Rapport technique UMIACS-TR-95-78, 23 pp, Institute for Advanced Computer Studies, Université du Maryland, College Park (MD, États-unis), 1995.

[lindsay80] P. H. Lindsay et D. A. Norman. Traitement de l'information et comportement humain - Une introduction à la psychologie. 754 pp, Éditions Études Vivantes, 1980.

[lippmann87] R. P. Lippmann. An introduction to computing with neural nets. IEEE ASSP Magazine, no 4, pp 4-22, 1987

[lipmann89] R. P. Lippmann. Review of neural networks for speech recognition. Neural Computation, vol. 1, pp 1-38, 1989.

[liu94] F.-H. Liu. Environmental adaptation for robust speech recognition. Thèse de doctorat mention informatique, 130 pp, Université Carnegie Mellon, Pittsburgh (PA, États-Unis), 1994.

[lockwood93] P. Alexandre, J. Boudy et P. Lockwood. Evaluation of car noise reduction/compensation techniques for digit recognition in a speaker-independent context. Proceedings of the European Conference on Speech Communication and Technology, pp 1255–1258, 1993.

[logar93] A. M. Logar, E. M. Corwin et W. J. B. Oldham. A comparison of recurrent neural network learning algorithms. Proceedings of the IEEE, pp 1129-1134, 1993.

[lombard11] E. Lombard. Le signe de l'élévation de la voix. Annales des maladies des oreilles, du larynx, du nez et du pharynx, vol. 37, pp 101–119, 1911.

[lonchamp88] F. Lonchamp. Étude sur la production et la perception de la parole. Première partie : les indices acoustiques de la nasalité vocalique.Thèse de doctorat d'état, Université de Nancy 2, 345 pp, 1988.

[lonchamp90] F. Lonchamp. Les sons du français. Rapport interne, 63 pp, Institut de phonétique, Université de Nancy 2, 1990.

[lonchamp91a] F. Lonchamp. Cours de phonétique et d'acoustique. Cours de licence de sciences du langage, Université de Nancy 2, 1991.

[lonchamp91b] F. Lonchamp. Cours sur le système phonatoire de l'être humain. Cours de licence de sciences du langage, Université de Nancy 2, 1991.

[lorento38] R. Lorento de No. The cerebral cortex: architecture, intracortical connections and motor projections. Physiology of the nervous system, Oxford University Press, pp 291-301, 1938.

[lorrain80] D. Lorrain. A panoply of stochastic "cannons". Computer Music Journal, no 3, pp 48-55, 1980.

[loughlin93] P. J. Loughlin, L. E. Atlas et J. W. Pitton. Advanced time-frequency representations for speech processing. [cooke93], pp 27-53, 1993.

[lucke92] H. Lucke et F. Fallside. Expanding the vocabulary of a connectionist recognizer trained on the DARPA resource management corpus. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 605–608, 1992.

[lumer92] E. D. Lumer et B. A. Huberman. Binding hierarchies: a basis for dynamical perceptual grouping. Neural Computation, vol. 4, pp 341-355, 1992.

[mackey77] M. Mackey et L. Glass. Oscillation and chaos in physiological control systems. Science, vol. 197, pp 287, 1977.

[maeda79] S. Maeda. An articulatory model of the tongue based on statistical analysis. Journal of the Acoustical Society of America, vol. 65, supp. no 1, page S22, 1979.

[mak92] B. Mak, J.-C. Junqua et B. Reaves. A robust speech/non-speech detection algorithm using time and frequency-based features. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 269–272, 1992.

[makhoul89] J. Makhoul, R. Schwartz et A. El-Jaroudi. Classification capabilities of two-layer neural nets. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 635–638, 1989.

[makhoul95] J. Makhoul. DSP techniques. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 11.3, pp 402-405, 1995.

[marchand88] P. Marchand. Cours de mathématiques pour l'informatique. Cours de licence d'informatique - Université de Nancy 1, 1988.

[mari96] J.-F. Mari. Perception de signaux complexes et interaction homme-machine. Mémoire d'Habilitation à Diriger les Recherches (spécialité informatique), 90 pp, Université Henri Poincaré - Nancy 1, 1996.

[markel76] J. D. Markel et A. H. Gray Jr. Linear prediction of speech. 288 pp, Springer-Verlag, 1976.

[martinetz93] T. M. Martinetz, S. G. Berkovich et K. J. Schulten. Neural-gas network for vector quantization and its application to time-series prediction. IEEE Transactions on Neural Networks, vol. 4, no 4, pp 558-569, 1993.

[massone94] L. L. E. Massone et T. Khoshaba. Local dynamic interactions in the collicular motor map: a neural network model. Rapport technique 4/94, 23 pp, Université du Nord-Ouest, Evanston (IL, États-Unis), 1994.

[massone95] L. L. E. Massone et T. Khoshaba. The role of initial conditions in dynamic pattern formation. Advances in Neural Information Processing Systems, 8 pp, 1995.

[maugis95] L. Maugis. VOCALISES speech interface: preliminary studies. Rapport technique CENA/N95-051/L.Maugis, version 1, 23 pp, Direction Générale de l'Aviation Civile, 1995.

[mcauley93] J. D. McAuley. Learning to perceive and produce rhythmic patterns in an artificial neural network. Rapport technique 371, 26 pp, département d'informatique, Université de l'Indiana, Bloomington (IN, États-Unis), 1993.

[mcauley94] J. D. McAuley et J. Stampfli. Analysis of the effect of noise on a model for the neural mechanism of short-term active memory. Neural computation, vol. 6, pp 668–678, 1994.

[mcclelland79] J. L. McClelland. On the time relations of mental processes: an examination of systems of processes in cascade. Psychological Review, vol. 86, no 4, pp 287–330, 1979.

[mcculloch43] W. S. McCulloch et W. Pitts. A logical calculus of the ideas imminent in the nervous activity. Bulletin of Mathematical Biophysics, no 5, pp 115-133, 1943.

[mcdermott82] D. McDermott. A temporal logic for reasoning about processes and plans. Cognitive Science, vol. 6, pp 101-155, 1982.

[mcdermott92] E. McDermott et S. Katagiri. Prototype-based discriminative training for various speech units. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 417-420, 1992.

[me94] L. Mé. Audit de sécurité par algorithmes génétiques. Thèse de doctorat mention informatique, no 1069, 143 pp, Université de Rennes 1 mention informatique, 1994.

[meddis86] R. Meddis. Simulation of mechanical to neural transduction in the auditory receptor. Journal of the Acoustical Society of America, vol. 79, no 3, pp 702-711, 1986.

[mella93] O. Mella. Contribution à l'identification automatique du locuteur sur des critères acoustiques et phonétiques. Thèse de doctorat mention informatique, 244 pp, Université de Nancy 1, Nancy (France), 1993.

[mellor93] B. A. Mellor et A. P. Varga. Noise masking in a transform domain. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 87–90, 1993.

[mery95] D. Méry. Cours sur l'algorithmique, les structures de données et la programmation. Cours de DEA de Chimie Informatique et Théorique, Université Henri Poincaré - Nancy 1. 1995.

[meyer92] Y. Meyer. Les ondelettes - Algorithmes et applications. 174 pp, Armand Colin Éditeur, 1992.

[mil1474b] Mil-Std 1474B (M2): Military standard: noise limits for army materiel. US Department of Defense, États-Unis, 1984.

[miller47] G. A. Miller. The masking of speech. Psychological Bulletin, vol. 44, pp 105–129, 1947.

[miller55] G. A. Miller et P. E. Nicely. An analysis of perceptual confusions among some english consonants. Journal of the Acoustical Society of America, vol. 27, pp 338–352, 1955.

[miller56] G. A. Miller. The magic number seven, plus or minus two. Psychological Review, vol. 63, pp 81-97, 1956. Cité dans [ingber95].

[minoux89] M. Minoux. Programmation mathématique - Théorie et algorithmes - Tome 1. 168 pp, CNET et ENST, Dunod, 1989.

[minsky69] M. Minsky et S. Papert. Perceptrons: an introduction to computational geometry. MIT Press, 1969.

[miptce94] Ministère français de l'Industrie, des Postes et Télécommunications et du Commerce Extérieur. Circulaire relative à l'emploi de la langue française au ministère. 8 pp, 1994.

[mirghafori95] N. Mirghafori, E. Fosler et N. Morgan. Fast speakers in large vocabulary continuous speech recognition: analysis and antidotes. Proceedings of the European Conference on Speech Communication and Technology, vol. 1, pp 491–495, 1995.

[miyatake90] M. Miyatake, H. Sawai, Y. Minami et K. Shikano. Integrated training for spotting japanese phonemes using large phonemic time-delay neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 449–452, 1990.

[moeschler95] J. Moeschler et A. Reboul. Dictionnaire encyclopédique de pragmatique. 562 pp, Éditions du Seuil, 1995.

[mokbel93] C. Mokbel, J. Monné et D. Jouvet. On line adaptation of a speech recognizer to variations in telephone line conditions. Proceedings of the European Conference on Speech Communication and Technology, pp 1247–1250, 1993.

[moles88] A. A. Moles. Théorie structurale de la communication et société. 296 pp, Collection technique et scientifique des télécommunications CNET-ENST, Masson, 1988.

[moore91] A. W. Moore. Variable resolution dynamic programming: efficiently learning action maps in multivariate real-valued state-spaces. Proceedings of the Eighth International Workshop on Machine Learning, pp 333-337, 1991.

[moreno94] P. Moreno et R. M. Stern. Sources of degradation of speech recognition in the telephone network. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 109–112, 1994.

[morgan91a] N. Morgan, H. Hermansky, H. Bourlard, P. Kohn et C. Wooters. Continuous speech recognition using PLP analysis with multilayer perceptrons. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 49-52, 1991.

[morgan91b] D. P. Morgan, C. L. Scofield et J. E. Adcock. Multiple neural network topologies applied to keyword spotting. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 313–316, 1991.

[morgan92] N. Morgan et H. Hermansky. RASTA extensions: robustness to additive and convolutional noise. ESCA Technical Research Workshop: Speech processing in adverse conditions, pp 115-118, 1992.

[morgan95a] N. Morgan et H. Bourlard. Continuous speech recognition. IEEE Signal Processing Magazine, pp 25-42, 1995.

[mori95] R. de Mori et F. Brugnara. HMM methods in speech recognition. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 1.5, pp 24-34, 1995.

[mougeot91] M. Mougeot et R. Azencott. Unsupervised learning for the visual cortex (layer 4): model and simulations. Proceedings of the International Joint Conference on Neural Networks, 6 pp, 1991.

[mountcastle78] V. B. Mountcastle. An organizing principle for cerebral function: the unit module and the distributed system. In: The mindful brain, pp 51-100, MIT Press, 1978.

[mousset94] E. Mousset. Les expression de la transformée en ondelettes discrète et de son inverse dans le formalisme des réseaux neuronaux multicouches. Revue Valgo, 15 pp, 1994.

[mozer93] M. C. Mozer. Neural net architectures for temporal sequence processing. In: Time series prediction: forecasting the future and understanding the past. SFI Studies in the sciences of complexity, Addison-Wesley Publishing, pp 243-264, 1993.

[mozer94] M. C. Mozer. Neural network music composition by prediction: exploring the benefits of psychoacoustic constraints and multiscale processing. Connection science, 32 pp, 1994.

[muller94] C. Muller. Perceprons multicouches : minimisation de la fonction coût et élagage - Cas de la série des consommations électriques journalières. Manuel de cours, 19 pp, École d'Été d'Analyse Numérique CEA-INRIA-EDF Réseaux de Neurones et Applications, 1994.

[murata92] N. Murata, S. Yoshizawa et S.-I. Amari. Network information criterion: determining the number of hidden units for an artificial neural network model. Rapport technique, 17 pp, Université de Tokyo, Tokyo (Japon), 1992.

[muthusamy90] Y. K. Muthusamy, R. A. Cole et M. Slaney. Speaker-independent vowel recognition: spectrograms versus cochleagrams. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 533-536, 1990.

[myers81a] C. S. Myers et L. R. Rabiner. Connected digit recognition using a level building DTW algorithm. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 29, pp 351-363, 1981.

[myers81b] C. S. Myers et L. R. Rabiner. A comparative study of several dynamic time-warping algorithms for connected-word recognition. The Bell System Technical Journal, vol. 60, no 7, pp 1389-1409, 1981.

[nadas88] A. Nadas, D. Nahamoo et M. Picheny. Speech recognition using noise-adaptive prototypes. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 517-520, 1988.

[nakamura90] M. Nakamura et S. Tamura. Vowel recognition by phoneme filter neural networks. Proceedings of the International Conference on Spoken Language Processing, pp 669-672, 1990.

[nakamura91] M. Nakamura, S. Tamura et S. Sagayama. Phoneme recognition by phoneme filter neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 85-88, 1991.

[nakamura92] S. Nakamura, H. Sawai et M. Sugiyama. Speaker-independent phoneme recognition using large scale neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 409-412, 1992.

[narenda90] K. S. Narenda et K. Parthasarathy. Gradient methods for optimization of dynamical systems containing neural networks. IEEE Transactions on Neural Networks, vol. 2, no 2, pp 252-262, 1991.

[nato87] NATO AC/243-(Panel 8/RSG 6) D/9. Research study group on the effects of impulsive noise. Organisation du Traité de l'Atlantique Nord, 1987.

[neisser67] U. Neisser. Cognitive psychology. Appleton-Century-Fox, 1967.

[nerrand94] O. Nerrand, P. Roussel-Gagot, D. Urbani, L. Personnaz et G. Dreyfus. Training recurrent neural networks: why and how? An illustration in dynamical process modeling. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 178-184, 1994.

[neumann92] J. von Neumann. L'ordinateur et le cerveau. 132 pp, Collection textes à l'appui, Série sciences cognitives, Éditions La Découverte, 1992.

[neumeyer94] L. Neumeyer et M. Wientraub. Probabilistic optimum filtering for robust speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 417-420, 1994.

[ney91] H. Ney. Speech recognition in a neural network framework: discriminative training of gaussian models and mixture densities as radial basis functions. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 573-576, 1991.

[nguyen90] D. Nguyen et B. Wodrow. The truck backer-upper: an example of self learning in neural networks. Proceedings of the International Conference on Neural Networks, vol. 2, pp 357-363, 1990.

[niles90] L. T. Niles, H. F. Silverman et M. A. Bush. Neural networks, maximum mutual information training and maximum likelihood training. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 493-496, 1990.

[niles92] L. T. Niles, L. D. Wilcox et M. A. Bush. Error-correcting training for phoneme spotting. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 425-428, 1992.

[nioche94] C. Nioche, J.-P. Tassin et D. Béroule. Vers une modélisation fonctionnelle de la neuromodulation par réseaux à propagation guidée. Actes des Journées Neurosciences et Sciences de l'Ingénieur, pp 231-234, 1994.

[nozalco93] J. A. Nozalco-Flores et S. J. Young. CSS-PMC: a combined enhancement/compensation scheme for continuous speech recognition in noise. Rapport technique CUED/F-INFENG/TR128, 31 pp, Université de Cambridge, Cambridge (Angleterre), 1993.

[ohkura91] K. Ohkura et M. Sugiyama. Speech recognition in noisy environment using a noise reduction neural network and a codebook mapping technique. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 929-932, 1991.

[olurotimi94] O. Olurotimi. Recurrent neural network training with feedforward complexity. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 185-197, 1994.

[omlin92] C. W. Omlin et C. L. Giles. Training second-order recurrent neural networks using hints. Proceedings of the 9th International Conference on Machine Learning, 6 pp, 1992.

[omlin94] C. W. Omlin, C. L. Giles, B. G. Horne, L. R. Leerink et T. Lin. Training recurrent neural networks with temporal input encodings. Proceedings of the International Conference on Neural Networks, pp 1267-1278, 1994.

[omlin95a] C. W. Omlin et C. L. Giles. Fault-tolerant implementation of finite-state automata in recurrent neural networks. Rapport technique RPI-CS-95-3, 32 pp, département d'informatique, Institut Polytechnique Rensselaer, Troy (NY, États-Unis), 1995.

[omlin95b] C. W. Omlin et C. L. Giles. Extraction of rules from discrete-time recurrent neural networks. Rapport technique CS-TR-3465, 18 pp, NEC Research Institute, Princeton (NJ, États-Unis), 1995.

[omlin96] C. W. Omlin, K. K. Thornber et C. L. Giles. Fuzzy finite-state automata can be deterministically encoded into recurrent neural networks. Rapport technique CS-TR-3599, 21 pp, NEC Research Institute, Princeton (NJ, États-Unis), 1996.

[ormesson96] J. d'Ormesson. Presque rien sur presque tout. 380 pp, Éditions Gallimard, 1996.

[palkar94] M. Palkar et J. C. Principe. Echo cancellation with the gamma filter. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 3, pp 369-372, 1994.

[pandit83] S. Pandit et S. Wu. Time series and system analysis with applications. Wiley, 1983.

[parks71] T. Parks. Choice of time scale in Laguerre approximations using signal measurements. IEEE Transactions on Optimal Control, vol. 16, pp 511-513, 1971.

[parlos94] A. G. Parlos, K. T. Chung et A. F. Atiya. Application of the recurrent multilayer perceptron in modeling complex process dynamics. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 255-264, 1994.

[pdp86] D. E. Rumelhart et J. L. McClelland. Parallel distributed processing: explorations in the microstructure of cognition. 2 volumes, 547 pp et 611 pp, MIT Press, 1986.

[pearlmutter89] B. A. Pearlmutter. Learning state space trajectories in recurrent neural networks. Neural Computation, vol. 1, pp 263-269, 1989.

[pearlmutter90] B. A. Pearlmutter. Dynamic recurrent neural networks. Rapport technique CMU-CS-90-196, 29 pp, Université Carnegie Mellon, Pittsburgh (PA, États-Unis), 1990.

[pearson94] J. K. Pearson et D. L. Bisset. Clifford networks: an introduction. Rapport technique, 34 pp, Université du Kent, Canterbury (Angleterre), 1994.

[peckles73] J. P. Peckles et M. Rossi. Le test diagnostic par paires minimales. Revue d'acoustique, vol. 27, pp 245-262, 1973.

[phillips87] M. S. Phillips. Speaker independent classification of vowels and diphthongs in continous speech. Proceedings of the 11th International Congress of Phonetic Sciences, pp 240-243, 1987.

[pican95] N. Pican. Approches statique et dynamique de la modulation des efficacités synaptiques dans les réseaux de neurones. Thèse de doctorat mention informatique, 162 pp, Université Henri Poincaré - Nancy 1, Nancy (France), 1995.

[piche94] S. W. Piche. Steepest descent algorithms for neural network controllers and filters. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 198-212, 1994.

[pickett56] J. M. Pickett. Effect of vocal force on the intelligibility at high noise levels. Journal of the Acoustical Society of America, vol. 28, pp 902–905, 1956.

[pierrel81] J.-M. Pierrel. Étude et mise en œuvre de contraintes linguistiques en compréhension automatique du discours continu - Application aux langages artificiels : le système myrtille I - Application aux langages pseudo naturels : le système myrtille II. Thèse de doctorat ès-sciences mathématiques, 446 pp, Université de Nancy 1, Nancy (France), 1981.

[pierrel87] J.-M. Pierrel. Dialogue oral homme-machine ; Connaissances linguistiques, stratégies et architectures des systèmes. 240 pp, Collection Langue-Raisonnement-Calcul, Hermès, 1987.

[pierrel91] J.-M. Pierrel et Y. Laprie. Dialogue oral homme-machine. Cours de DEA informatique, Université de Nancy 1, 1991.

[pineda87] F. J. Pineda. Generalization of backpropagation to recurrent neural networks. Physical Review Letters, vol. 59, no 19, pp 2229-2232, 1987.

[pisoni93] D. B. Pisoni. Long-term memory in speech perception: some new findings on talker variability, speaking rate and perceptual learning. Speech Communication, vol. 13, nos 1-2, pp 109–125, 1993.

[plate91] T. A. Plate. Holographic reduced representations. Rapport technique CRG-TR-91-1, 28 pp, Université de Toronto, Toronto (Canada), 1991.

[plate94] T. A. Plate. Distributed representations and nested compositional structure. Thèse de doctorat mention informatique, 202 pp, Université de Toronto, Toronto (Canada), 1994.

[postma96] E. O. Postma et H. J. van der Herik. Geometric phase shift in a neural oscillator. Rapport technique, 8 pp, Université de Maastricht, Maastricht (Pays-Bas), 1996.

[potage90c] J. Potage. Traitement du signal de parole, volume 10627/03/..a1990, tome 3 : la numérisation de la parole (2). Cours du service IAP, 86 pp, École Supérieure d'Électricité, établissement de Metz, 1991.

[prager86] R. W. Prager. Boltzmann machines for speech recognition. Computer speech and language, vol. 4, no 1, 1986.

[pratt91] L. Y. Pratt et C. A. Kamm. Improving a phoneme classification neural network through problem decomposition. Proceedings of the International Joint Conference on Neural Networks, 6 pp, 1991.

[priel93] A. Priel, M. Blatt, T. Grossman, E. Domany et I. Kanter. Computational capabilities of restricted two layered perceptrons. Rapport technique, 41 pp, Institut Weizmann (Israël), 1993.

[principe92] J. C. Principe, A. de Vries et P. G. de Oliveira. Generalized feedforward structures: a new class of adaptive filters. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 4, pp 244-248, 1992.

[principe93a] J. C. Principe, A. de Vries et P. G. de Oliveira. The gamma filter - a new class of adaptive IIR filters with restricted feedback. IEEE Transactions on Signal Processing, vol. 41, no 2, pp 649–656, 1993.

[principe93b] J. C. Principe, A. de Vries, J.-M. Kuo et P. G. de Oliveira. Modeling applications with the focused gamma net. Advances in Neural Information Processing Systems, pp 143–150, 1993.

[principe93c] J. C. Principe, A. Rathie et J.-M. Kuo. Prediction of chaotic time series with neural networks and the issue of dynamic modelling. World Scientific, vol 2, no 4, pp 989-996, 1993.

[principe93d] J. C. Principe et J.-M. Kuo. Backpropagation through time with fixed memory size constraints. Proceedings of the IEEE Neural Network for Signal Processing Workshop, pp 207-215, 1993.

[principe93e] J. C. Principe et J.-M. Kuo. Noise reduction in state space using the focused gamma neural network. Proceedings of the SPIE, vol. 2038, pp 326-331, 1993.

[principe94a] J. C. Principe, J. M. Kuo et S. Celebi. An analysis of the gamma memory in dynamic neural networks. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 331-337, 1994.

[principe94b] J. C. Principe et M. Motter. System identification with dynamic neural networks. Proceedings of the World Conference on Neural Network, vol. 2, pp 284-289, 1994.

[principe94c] J. C. Principe et L. Turner. Word spotting with the gamma neural model. Proceedings of the World Conference on Neural Network, vol. 4, pp 502-505, 1994.

[principe94d] Principe J., Hsu H., and Kuo J. Analysis of short term neural memory structures for nonlinear prediction. In: Advances in Neural Information Processing Systems, pp 1011-1018, 1994.

[principe95] J. C. Principe, S. Celebi, C. Wang, C. W. Lefebvre, J. G. Harris et N. R. Euliano. Time lagged recurrent networks. Rapport technique, 37 pp, Computational Neuro Engineering Laboratory, Université de Floride à Gainesville, Gainesville (FL, États-Unis), 1995.

[puskorius94] G. V. Puskorius et L. A. Feldkamp. Neurocontrol of nonlinear dynamical systems with Kalman filter-trained recurrent networks. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 279-297, 1994.

[quastler58] H. Quastler. A primer on information theory. Symposium on information theory in biology, pp 3-49, Pergamon Press, 1958.

[rabiner77] L. R. Rabiner, S. E. Levinson, A. E. Rosenberg et J. G. Wilpon. Speaker independent recognition for isolated words using clustering techniques. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 27, no 4, pp 336-349, 1977.

[rabiner89] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, vol. 77, no 2, pp 257-286, 1989.

[rahim92] M. G. Rahim. A neural tree network for phoneme classification with experiments on the TIMIT database. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 345–348, 1992.

[rajasekaran86] P. Rajasekaran, G. Doddington et J. Picone. Recognition of speech under stress and in noise. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 733–736, 1986.

[rall69] W. Rall. Time constants and electrotonic length of membrane cylinders and neurons. Biophysical Journal, vol. 9, pp 1483-1508, 1969. Cité dans [kim92].

[ramesh91] P. Ramesh, J. G. Wilpon, M. A. McGee, D. B. Roe, C. H. Lee et L. R. Rabiner. Spoken independent' recognition of spontaneously spoken connected digits. Proceedings of the European Conference on Speech Communication and Technology, pp 17-20, 1991.

[rander92] P. W. Rander et K. P. Unnikrishnan. Learning the time-delay characteristics in a neural network. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, pp 285–288, 1992.

[ray93] T. S. Ray. An evolutionary approach to synthetic biology - Zen and the art of creating life. Artificial Life, vol. 1, no 1, 39 pp, 1993.

[renals89] S. Renals et R. Rohwer. Learning phoneme recognition using neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 413-416, 1989.

[renals91] S. Renals, N. Morgan, H. Bourlard, M. Cohen, H. Franco, C. Wooters et P. Kohn. Connectionist speech recognition: status and prospects. Rapport technique TR-91-070, 24 pp, International Computer Science Institute, Université de Berkeley, Berkeley (CA, États-Unis), 1991.

[renals94a] S. Renals, M. Hochberg et A. J. Robinson. Learning temporal dependencies in large-scale connectionnist speech recognition. Advances in Neural Information Processing Systems, pp 1051–1058, 1994.

[renals94b] S. Renals et M. Hochberg. Using gamma filters to model temporal dependencies in speech. Proceedings of the International Conference on Spoken Language Processing, pp 1491–1494, 1994.

[richards92] E. L. Richards. A multi-task neural network approach to speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 413-416, 1992.

[rigoll92] G. Rigoll. Unsupervised information theory-based training algorithms for multilayer neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 393-396, 1992.

[rinzel89] J. Rinzel et G. B. Ermebtrout. Analysis of neural excitability and oscillations. Methods in neural modeling - From synapses to networks, pp 135-169, 1989.

[ritter89] H. J. Ritter, T. M. Martinetz et K. J. Schulten. Topology-conserving maps for learning visuo-motor coordination. Neural Networks, vol. 2, pp 159-168, 1989.

[robert-ribes95] J. Robert-Ribes, J.-L. Schwartz et P. Escudier. Auditory, visual and audiovisual vowel representations: experiments and modelling. Proceedings of the 13th International Congress of Phonetic Sciences, vol. 3, pp 114-121, 1995.

[robinson89] A. J. Robinson. Dynamic error propagation networks. Thèse de doctorat mention informatique, 30 pp, Université de Cambrige, Cambridge (Angleterre), 1989.

[robinson90a] A. J. Robinson et F. Fallside. Phoneme recognition from the TIMIT database using recurrent error propagation networks. Rapport technique CUED/F-INFENG/TR42, 12 pp, département d'ingénierie, Université de Cambridge, Cambridge (Angleterre), 1990.

[robinson90b] A. J. Robinson, J. Holdsworth, R. Patterson et F. Fallside. A comparison of preprocessors for the Cambridge recurrent error propagation network speech recognition system. Proceedings of the International Conference on Spoken Language Processing, 4 pp, 1990.

[robinson91] A. J. Robinson. Several improvments to a recurrent error propagation network phone recognition system. Rapport technique CUED/F-INFENG/TR82, 12 pp, département d'ingénierie, Université de Cambridge, Cambridge (Angleterre), 1991.

[robinson92] A. J. Robinson. A real-time recurrent error propagation network word recognition system. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 617–620, 1992.

[robinson94] A. J. Robinson. An application of recurrent nets to phone probability estimation. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 298-305, 1994.

[robinson96] A. J. Robinson. Speech analysis. Manuel de cours, Université de Cambridge, 53 pp, Cambridge (Angleterre), 1996.

[roche94] E. Roché et G. Vaucher. Perceptrons multicouches étendu au corps des complexes. Journées NSI-ACTH, 19 pp, 1994.

[rock91] I. Rock et S. Palmer. L'héritage du gestaltisme. Pour la science, no 160, pp 64-70, 1991.

[romary89] L. Romary. Vers la définition d'un modèle cognitif autour de la représentation du temps dans un système de dialogue homme-machine. Thèse de doctorat mention informatique, 208 pp, Université de Nancy 1, Nancy (France), 1989.

[rosenblatt59] F. Rosenblatt. Principles of neurodynamics. Spartan Books, 1959.

[sadri87] F. Sadri. Three recent approaches to temporal reasoning. In: Temporal logic and their applications, Academic Press, 1987.

[saha82] D. C. Saha et G. P. Rao. A general algorithm for parameter identification in lumped continuous systems: the poisson moment functional approach. IEEE Transactions on Automatic Control, vol. 27, no 1, pp 223–225, 1982.

[sakoe71] H. Sakoe et S. Chiba. A dynamic programming approach to continuous speech recognition. Proceedings of the 7th International Conference on Acoustics, article 20C-13, 6 pp, 1971.

[sakoe78] H. Sakoe et S. Chiba. Dynamic programming algorithms optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 26, no 1, pp 43-49, 1978.

[sakoe79] H. Sakoe. Two level DP-matching - A dynamic programming based pattern matching algorithm for connected word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 27, pp 588-595, 1979.

[sankar91a] A. Sankar et R. J. Mammone. Optimal pruning of neural tree networks for improved generalization. Proceedings of the IEEE International Joint Conference on Neural Networks, 6 pp, 1991.

[sankar91b] A. Sankar et R. J. Mammone. Speaker independent vowel recognition using neural tree networks. Proceedings of the IEEE International Joint Conference on Neural Networks, 6 pp, 1991.

[sankar95] A. Sankar et C. H. Lee. Robust speech recognition based on stochastic matching. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 121-124, 1995.

[sastry94] P. S. Sastry, G. Santharam et K. P. Unnikrishnan. Memory neuron networks for identification and control of dynamical systems. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 306-319, 1994.

[sauer91] T. Sauer, J. A. Yorke et M. Casdagli. Embedology. Journal of Statistical Physics, vol. 65, nos 3/4, pp 579–616, 1991.

[saul95a] L. K. Saul et M. I. Jordan. Learning in Boltzmann trees. Rapport technique, 11 pp, Massachusetts Institute of Technology, Cambridge (MA, États-Unis), 24 janvier 1995.

[saul95b] L. K. Saul et M. I. Jordan. Boltzmann chains and hidden Markov models. Rapport technique, 10 pp, Massachusetts Institute of Technology, Cambridge (MA, États-Unis), 1995.

[sawai91a] H. Sawai. Frequency-time-shift-invariant time-delay neural networks for robust continuous speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 45–48, 1991.

[sawai91b] H. Sawai et S. Nakamura. Time-delay neural network architectures for high performance speaker-independent recognition. Proceedings of the European Conference on Speech Communication and Technology, pp 1011-1014, 1991.

[sawai91c] H. Sawai. TDNN-LR continuous speech recognition system using adaptive incremental TDNN training. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 53-56, 1991.

[schiffmann92] W. Schiffmann, M. Joost et R. Werner. Optimization of the backpropagation algorithm for training multilayer perceptrons. Rapport technique, 36 pp, Institut de Physique, Université de Coblence, Coblence (Allemagne), 1992.

[schmidhuber96] J. Schmidhuber et S. Hochreiter. Guessing can outperform many long time lag algorithms. Note technique IDSIA-19-96, 3 pp, 1996.

[scourias95] J. Scourias. Overview of the global system for mobile communications. Rapport technique, 25 pp, Université de Waterloo, Waterloo (ON, Canada), 1995.

[seneff88] S. Seneff. A joint synchrony/mean-rate model of auditory speech processing. Journal of Phonetics, vol. 16, pp 55-76, 1988.

[servan91] D. Servan-Schreiber, A. Cleeremans et J. L. McClelland. Graded state machines: the representation of temporal contingencies in simple recurrent networks. Machine Learning, vol. 7, pp 161-193, 1991.

[sejnowski87] T. J. Sejnowski et C. Rosenberg. Parallel network that learn to pronounce english text. Complex systems, vol. 1, pp 145-168, 1987.

[shirai91] K. Shirai, E. Kitagawa et T. Endo. Optimal construction of context sensitive quantizier for phoneme recognition in continuous speech. Proceedings of the European Conference on Speech Communication and Technology, pp 405-408, 1991.

[shynks89] J. Shynks. Adaptive IIR filtering. IEEE ASSP Magazine, pp 4-21, 1989.

[siegler95] M. A. Siegler et R. M. Stern. On the effects of speech rate in large vocabulary speech recognition systems. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 612–615, 1995.

[siegelmann95] H. T. Siegelmann, B. G. Horne et C. L. Giles. Computational capabilities of recurrent NARX neural networks. Rapport technique UMIACS-TR-95-12, 12 pp, Institute for Advanced Computer Studies, Université du Maryland, College Park (MD, États-unis), 1995.

[silva92] T. Silva, P. G. de Oliveira, J. C. Principe et A. de Vries. Generalized feedforward filters with complex poles. Proceedings of the IEEE Workshop Neural Networks for Signal Processing, pp 503-510, 1992.

[silva95] T. Silva. Optimality conditions for truncated second order Kautz networks with two complex conjugate poles. IEEE Transactions on Automatic Control, 1995.

[simon94] Sous la direction de M. Simon. La peau de l'âme - intelligence artificielle, neurosciences, philosophie, théologie. 443 pp, Les Éditions du Cerf, 1994.

[siohan95] O. Siohan. Reconnaissance automatique de la parole continue en environnement bruité : application à des modèles stochastiques de trajectoire. Thèse de doctorat mention informatique, 213 pp, Université Henri Poincaré - Nancy 1, Nancy (France), 1995.

[sjoberg95] J. Sjöberg. Non-linear systems identification with neural networks. Thèse de doctorat mention informatique, 223 pp, Université de Linköping, Linköping (Suède), 1995.

[slaney88] M. Slaney. Lyon's cochlear model. Rapport technique 13, Apple computer, 1988.

[slaney93a] M. Slaney et R. F. Lyon. On the importance of time - A temporal representation of sound. In: [cooke93].

[slaney93b] M. Slaney. An efficient implementation of the Patterson-Holdsworth auditory filter bank. Rapport technique 35, Apple computer, 1993.

[slaney95] M. Slaney. MATLAB auditory toolbox. Rapport technique 45, 41 pp, Apple computer, 1995.

[sorensen91a] H. B. D. Sorensen. A cepstral noise reduction multi-layer neural network. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 933-936, 1991.

[sorensen91b] H. B. D. Sorensen et U. Hartmann. A self-structuring neural noise reduction model. Proceedings of the European Conference on Speech Communication and Technology, pp 567-570, 1991.

[souvay92] G. Souvay. Diapason : un environnement de développement pour l'intégration d'une entrée vocale dans des applications de type commande de machine. Thèse de doctorat mention informatique, 135 pp, Université de Nancy 1, Nancy (France), 1992.

[spender93] Lady N. Spender. Psychologie de la musique. In: Le cerveau, un inconnu, dictionnaire encyclopédique, collection Bouquins, Robert Laffont, pp 1094-1104, 1993.

[spieth56] W. Spieth. Annoyance threshold judgments of bands of noise. Journal of the Acoustical Society of America, vol. 28, pp 872–877, 1956.

[srinivasan94] B. Srinivasan, J. R. Radsad et N. J. Rao. Back propagation through adjoints for the identification of nonlinear dynamic systems using recurrent neural models. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 213-228, 1994.

[steenek80] H. J. M. Steeneken et T. Houtgast. A physical method for measuring speech transmission quality. Journal of the Acoustical Society of America, vol. 67, no 1, pp 318–326, 1980.

[steenek86] H. J. M. Steeneken. Diagnostic information of subjective intelligibility test. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 4 pp, 1986.

[steenek92b] H. J. M. Steeneken. Subjective and objective intelligibility measures. ESCA Technical Research Workshop: Speech processing in adverse conditions, pp 1–10, 1992.

[steenek95] H. J. M. Steeneken. Speech communication quality. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 13.9, pp 504-506, 1995.

[stern94] R. M. Stern, F.-H. Liu, P. J. Moreno et A. Acero. Signal processing for robust speech recognition. Proceedings of the International Conference on Spoken Language Processing, vol. 3, pp 1027-1030, 1994.

[stern95] R. M. Stern. Robust speech recognition. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 1.4, pp 17-23, 1995.

[sternberg69] S. Sternberg. The discovery of processing stages: extensions of Donder's method. Acta Psychologica, vol. 30, pp 276-315, 1969.

[stromberg91] J.-E. Strömberg, J. Zrida et A. Isaksson. Neural trees - Using neural nets in a tree classifier structure. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 137-140, 1991.

[summers88] W. Summers, D. Pisoni, R. Bernacki, R Pedlow et M. Stokes. Effects of noise on speech production: acoustic and perceptual analyses. Journal of the Acoustical Society of America, vol. 84, no 3, pp 917–928, 1988.

[sun91] G.-Z. Sun, H.-H. Chen, Y.-C. Lee. Green's function method for fast on-line learning algorithm of recurrent neural network. Advances in Neural Information Processing System, pp 333-340, 1991.

[sun95] G.-Z. Sun, C. L. Giles, H.-H. Chen et Y.-C. Lee. The neural network pushdown automaton: model, stack and learning simulations. Rapport technique UMIACS-TR-93-77, 2ème édition, 36 pp, Institute for Advanced Computer Studies, Université du Maryland, College Park (MD, États-Unis), 1995.

[sutherland] N. S. Sutherland. The biological causes of irrationality. [damasio96a], pp 145-156.

[sutton84] R. R. Sutton. Temporal credit assignment in reinforcement learning. Thèse de doctorat mention informatique, Université du Massachussets, Amherst (MA, États-Unis), 1984.

[szentagothai73] J. Szentagothai. Synaptology of the visual cortex. In: Handbook of sensory physiology, Springer-Verlag, 1973.

[takami91] J.-I. Takami et S. Sagayama. A pairwise discriminant approach to robust phoneme recognition by time-delay neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 89-92, 1991.

[takens81] F. Takens. Detecting strange attractors in turbulence. Dynamical systems and turbulence, Lectures Notes in Mathematics, vol. 898, pp 366–381, Springer-Verlag, 1981.

[tamura90] S.-I. Tamura et M. Nakamura. Improvements to the noise reduction neural network. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 825-828, 1990.

[tank87a] D. W. Tank et J. J. Hopfield. Concentrating information in time: analog neural networks with applications to speech recognition problems. Proceedings of the IEEE International Conference on Neural Networks, 1987.

[tank87b] D. W. Tank et J. J. Hopfield. Neural computation by concentrating information in time. Proceedings of the National Academy of Science of USA, vol 84, pp 1896-1900, 1987.

[tebelskis90] J. Tebelskis et A. H. Waibel. Large vocabulary recognition using linked predictive neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 437-440, 1990.

[tebelskis91] J. Tebelskis, A. H. Waibel, B. Petek et O. Schmidbauer. Continuous speech recognition using linked predictive neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 61-64, 1991.

[tewfik91] A. H. Tewfik et P. E. Jorgensen. On the choice of a wavelet for signal coding and processing. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 2025-2028, 1991.

[thrun91] S. B. Thrun et K. Möller. On plannig and exploration in non-discrete environments. Rapport technique, 31 pp, Forschungszentrum Informationstechnik GmbH, Berlin (Allemagne), 1991.

[timitphon90] National Institute of Standards and Technology, États-Unis. TIMIT Phonetic Documentation, 1990. Fichier /timit/doc/phoncode.doc.

[timitdic91] National Institute of Standards and Technology, États-Unis. TIMIT Lexicon Documentation, 1991. Fichier /timit/doc/timitdic.doc.

[tino95] P. Tino, B. G. Horne et C. L. Giles. Finite state machines and recurrent neural networks - Automata and dynamical systems approaches. Rapport technique UMIACS-TR-95-1, 47 pp, Institute for Advanced Computer Studies, Université du Maryland, College Park (MD, États-Unis), 1995.

[tom95] M. D. Tom et M. F. Tenorio. A neural computation model with short-term memory. IEEE Transactions on Neural Networks, vol. 6, no 2, pp 387-397, 1995.

[torkkola91] K. Torkkola et M. Kokkonen. Using the topology-preserving properties of SOFMs in speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 261-264, 1991.

[torkkola91b] K. Torkkola, M. Kokkonen, M. Kurimo et P. Utela. Improving short-time speech frame recognition results by using context. Proceedings of the European Conference on Speech Communication and Technology, pp 793-796, 1991.

[tracey94] J. Tracey et J. C. Principe. Isolated word speech recognition using the gamma model. Journal of Artificial Neural Networks, vol. 14, no 1, pp 481-489, 1994.

[trompf92] M. Trompf. Experiments with noise reduction neural networks for robust speech recognition. Rapport technique TR-92-035, 16 pp, International Computer Science Institute, Berkeley (CA, États-Unis), 1992.

[tsoi94] A. C. Tsoi et A. D. Back. Locally recurrent globally feedforward networks: a critical review of architectures. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 229-239, 1994.

[tsung90] F.-S. Tsung. Learning in recurrent finite difference networks. Proceedings of the 1990 Summer School on Connectionist Models, pp 124-130, 1990.

[tsung93] F.-S. Tsung et G. W. Cottrell. Phase-space learning for recurrent networks. Rapport technique, 18 pp, Université de Californie à San Diego, San Diego (CA, États-Unis), 1993.

[turing36] A. M. Turing. On computable numbers, with an application to the Endscheidungsproblem. Proceedings of the London Mathematical Society: 2nd Series, no 42, pp 230-265, 1936 ; Addition, no 43, pp 544-546, 1936.

[unnikrishnan91] K. P. Unnikrishnan, J. J. Hopfield et D. W. Tank. Connected-digit speaker-dependent speech recognition using a neural network with time-delayed connections. IEEE Transactions on signal processing, vol. 39, no 3, pp 698-713, 1991.

[usher95] M. Usher et J. L. McClelland. On the time course of perceptual choice: a model based on principles of neural computation. Rapport technique PDP-CNS-95-5, 52 pp, Université Carnegie Mellon, Pittsburgh (PA, États-Unis), 1995.

[varga90] A. P. Varga et R. K. Moore. Hidden Markov model decomposition of speech and noise. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 845-848, 1990.

[varga92] A. P. Varga, H. J. M. Steeneken, M. Tomlinson et D. Jones. The NOISEX-92 study on the effect of additive noise on automatic speech recognition. NOISEX92 CDROM, 1992.

[vaucher96] Gilles Vaucher. À la recherche d'une algèbre neuronale spatio-temporelle. Rapport technique, École supérieure d'Électricité, Supélec campus de Rennes, 1996.

[viterbi67] A. J. Viterbi. Error bounds for convolutional codes and asymptotically optimal decoding algorithm. IEEE Transactions on Information Theory, vol. 13, pp 260-296, 1967.

[vries90] A. de Vries et J. C. Principe. The gamma model - a new neural model for temporal processing. Proceedings of the IEEE Conference on Engineering in Medecine and Biology, pp 1439-1440, 1990.

[vries91a] A. de Vries et J. C. Principe. A theory of neural networks with time delays. Advances in Neural Information Processing Systems, pp 162–168, 1991.

[vries91b] A. de Vries, J. C. Principe et P. G. de Oliveira. Adaline with adaptive recursive memory. Proceedings of the IEEE Neural Networks for Signal Processing Workshop, pp 101-110, 1991.

[vries92] A. de Vries et J. C. Principe. The gamma model - a new neural model for temporal processing. Neural Networks, vol. 5, pp 565-576, 1992.

[wahlberg94] B. Wahlberg. System identification with the Kautz models. IEEE Transactions on Automatic Control, vol. 39, no 6, pp 1276-1282, 1992.

[waibel88] A. H. Waibel, T. Hanazawa, G. E. Hinton, K. Shikano et K. J. Lang. Phoneme recognition using time-delay neural networks. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 107-110, 1988.

[waibel89] A. H. Waibel, T. Hanazawa, G. E. Hinton, K. Shikano et K. J. Lang. Phoneme recognition using time-delay neural networks. IEEE Transaction on Acoustics, Speech and Signal Processing, vol. 37, no 3, pp 328–339, 1989.

[waibel91] A. H. Waibel. Neural network approaches for speech recognition. In: Advances in speech signal processing, pp 555-595, Marcel Delker, 1991.

[wan93] E. A. Wan. Time series prediction by using a connectionist network with internal delay lines. In: Time series prediction: forecasting the future and understanding the past, SFI Studies in the sciences of complexity, Addison-Wesley, pp 195-217, 1993.

[wang90] D. Wang et M. A. Arbib. Complex temporal sequence learning based on short-term memory. Proceedings of the IEEE, vol. 78, no 9, pp 1536-1543, 1990.

[wang91] H. Wang et F. Itakura. An approach of deverberation using multi-microphone sub-band envelope estimation. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 953–956, 1991.

[wang95a] C. Wang et J. C. Principe. When does supervised learning degenerate to unsupervised learning? Rapport technique ARPA/ONR N00014-94-1-0858, 19 pp, Computational Neuro Engineering Laboratory, Université de Floride à Gainesville, Gainesville (FL, États-Unis), 1995.

[wang95b] C. Wang et J. C. Principe. A relation between hebbian and MSE learning. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 5, pp 3363-3366, 1995.

[wang95c] C. Wang et J. C. Principe. A relation between hebbian and MSE learning in nonlinear neural networks. Proceedings of the World Congress on Neural Networks, vol. 1, pp 540-543, 1995.

[wang96] L. Wang. Local dynamic modeling with self-organizing feature map. Thèse de doctorat mention informatique, 126 pp, Université de Floride à Gainesville, Gainesville (FL, États-Unis), 1996.

[webber94] C. J. S. Webber. Self-organisation of transformatio-invariant detectors for constituents of perceptual patterns. Rapport technique, 26 pp, DRA, Farnborough (Angleterre), 1994.

[werbos89] P. Werbos. Backpropagation and neural control: a review and prospectus. Proceedings of the International Joint Conference on Neural Networks, vol. 1, pp 209-216, 1989.

[werbos90] P. J. Werbos. Backpropagation through time: what it does and how to do it. Proceedings of the IEEE, vol. 78, no 10, pp 1550-1560, 1990.

[whitney36] H. Whitney. Differentiable manifolds. Annals of Mathematics, vol. 37, pp 645, 1936. Cité dans [principe95].

[widrow85] B. Widrow et S. Stearns. Adaptive signal processing. Prentice Hall, 1985.

[williams89a] R. J. Williams et D. Zipser. A learning algorithm for continually running fully recurrent neural networks. Neural Computation, vol. 1, no 4, pp 270-280, 1989.

[williams89b] R. J. Williams et D. Zipser. Experimental analysis of the real-time recurrent learning algorithm. Connection Science, vol. 1, no 1, pp 87-111, 1989.

[wurtz94] R. P. Würtz. Multilayer dynamic link networks for establishing image point correspondences and visual object recognition. Thèse de doctorat mention informatique, 155 pp, Université de la Ruhr, Bochum (Allemagne), 1994.

[yamaguchi90] K. Yamaguchi, K. Sakamoto, T. Akabane et Y. Fujimoto. A neural network for speaker-independent isolated word recognition. Proceedings of the International Conference on Spoken Language Processing, pp 1077-1080, 1990

[zak87] M. Zak. Creative dynamics approach to neural intelligence. Biological Cybernetics, vol. 64, pp 15-23, 1990.

[zbikowski94] R. W. Zbikowski. Recurrent neural networks: some control aspects. Thèse de doctorat mention informatique, 103 pp, Faculté d'Ingénierie, Université de Glasgow, Glasgow (Angleterre), 1994.

[zeng92] H. Zeng et T. Yu. Parallel sequential running neural network and its application to automatic speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp 429-432, 1992.

[zeng94] Z. Zeng, R. M. Goodman et P. Smyth. Discrete recurrent neural networks for grammatical inference. IEEE Transactions on Neural Networks, vol. 5, no 2, pp 320-330, 1994.

[zerling79] J.-P. Zerling. Articulation et coarticulation dans les groupes occlusive-voyelle en français. Thèse de doctorat de 3ème cycle, Université de Nancy 2, Nancy (France), 1979.

[zhang92] Q. Zhang et A. Benveniste. Wavelet networks. IEEE Transactions on Neural Networks, vol. 3, no 6, pp 889-898, 1992.

[zhang93] Q. Zhang. Regressor selection and wavelet network construction. Rapport de recherche 1967, 22 pp, Institut National de Recherche en Informatique et Automatique, 1993.

[zhang94] K. Zhang. Temporal association by hebbian connections: the method of characteristic systems. Rapport technique 9402, 21 pp, Département des sciences cognitives, Université de Californie à San Diego, San Diego (CA, États-Unis), 1994.

[zhu90] M. Zhu et K. Fellbaum. A connectionnist model for speaker-independent isolated word recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp 529–532, 1990.

[zipser91] D. Zipser. Recurrent network model of the neural mechanism of short-term active memory. Neural computation, vol. 3, pp 179–193, 1991.

[zue95] V. Zue, R. Cole et W. Ward. Speech recognition. Survey of the state of the art in human language technology, NSF, CE-DG13E & OGI-CSLU éds, art. 1.2, pp 4-10, 1995.

[zwicker81] E. Zwicker et R. Feldtkeller. Psychoacoustique - L'oreille récepteur d'information. 234 pp, Collection technique et scientifique des télécommunications - CNET - ENST, Masson, 1981.

INDEX DE MOT-CLÉS

Nous espérons que cet index permettra au lecteur de retrouver facilement les informations contenues dans cette thèse et pour lesquelles il aura un intérêt particulier. C'est, après tout, le rôle de tout bon index... La constitution de cet index ne résulte cependant pas d'une recherche de mot-clés mais d'un marquage des termes cités lors d'une relecture. Ainsi, toutes les références des mots de cet index ne correspondent pas aux seules occurences des mots qu'il sera possible de trouver dans le texte. Nous avons, en effet, uniquement indexé les références nous semblant être les plus significatives et nous espérons que les choix réalisés auront été les même que ceux du lecteur.

Mot-clés pour la lettre A

Mot-clés pour la lettre B

Mot-clés pour la lettre C

Mot-clés pour la lettre D

Mot-clés pour la lettre E

Mot-clés pour la lettre F

Mot-clés pour la lettre G

Mot-clés pour la lettre H

Mot-clés pour la lettre L

Mot-clés pour la lettre M

Mot-clés pour la lettre N

Mot-clés pour la lettre P

Mot-clés pour la lettre Q

Mot-clés pour la lettre R

Mot-clés pour la lettre S

Mot-clés pour la lettre T

Mot-clés pour la lettre V

Mot-clés pour la lettre W

Mot-clés pour la lettre Z

Université HENRI POINCARÉ NANCY 1

Nom : **BUNIET**

Prénom : **Laurent**

DOCTORAT de l'UNIVERSITE HENRI POINCARE, NANCY-I

en INFORMATIQUE

VU, APPROUVÉ ET PERMIS D'IMPRIMER

Nancy, le 8 JUIL 1997 UHP 020/97

Le Président de l'Université

UNIVERSITÉ HENRI POINCARÉ NANCY 1 LE PRÉSIDENT J.P. FINANCE

3C-F9

UNIVERSITÉ HENRI POINCARÉ · NANCY 1
24-30, rue Lionnois B.P.3069 54013 Nancy cedex
Tél : 83 32 81 81 Fax : 83 32 95 90
Adresse électronique : _____ @presn1.u-nancy.fr

RÉSUMÉ

Les recherches effectuées dans le domaine de la reconnaissance automatique de la parole (RAP) permettent d'envisager un éventail toujours plus large d'applications industrielles ou grand public. Cependant, la compréhension des mécanismes de production et de reconnaissance de la parole par l'Homme ne suffit pas en elle-même pour élaborer effectivement les dites applications. Les conditions de laboratoire qui ont prévalues lors de l'enregistrement des premiers corpus de parole utilisés à des fins de recherches sont en effet très différentes des conditions réelles que l'on rencontre généralement dans les lieux de travail ou de vie. Ayant le plus souvent été enregistrés en chambre anéchoïde, ces corpus ne permettaient pas plus d'appréhender les dégradations que le milieu peut engendrer sur le signal de parole que de constater quelles pouvaient être les modifications provoquées sur ce signal par un locuteur essayant de s'adapter à son milieu. Certaines des recherches actuelles en RAP essaient donc d'améliorer les capacités de résistance au bruit des systèmes existants. Pour ce faire, il est possible d'utiliser un système d'abord défini pour la reconnaissance de la parole non bruitée en lui ajoutant un mécanisme lui permettant de s'adapter à certaines conditions de bruit. Il est également possible de définir un système ab-nihilo qui soit tout aussi bien adapté aux conditions non bruitées qu'aux conditions bruitées.

Le sujet de cette thèse porte sur la reconnaissance de petits vocabulaires, tels que les lettres ou les chiffres, prononcés de manière continue en milieu bruité. Pour mener à bien cette étude, différentes architectures connexionnistes ont été étudiées. L'utilisation de modèles connexionnistes nous a permis de mettre au point, grâce au mécanisme d'apprentissage, des systèmes qui sont immédiatement adaptés à différentes conditions de bruit. Un premier système a été mis en place qui permet, en trois étapes, de reconnaître les mots du vocabulaire étudié. Une première étape identifie des points d'ancrage dans le signal, ces points d'ancrage correspondant à une segmentation des parties vocaliques du signal. Une deuxième étape permet de reconnaître les voyelles contenues dans les segments retenus alors qu'une troisième étape permet de distinguer les différents mots du vocabulaire qui possèdent les mêmes voyelles. Cette architecture, basée sur des perceptrons multicouches, a prouvé être de bonne qualité mais l'étape de segmentation s'est révélée être de moindre qualité à des rapports signal sur bruit faible c'est à dire de l'ordre de 6 décibels ou moins. Ceci nous a poussé à étudier des modèles connexionnistes dynamiques, à l'opposé des perceptrons multicouches qui sont des modèles statiques. Les modèles dynamiques ont la particularité de mettre en place des mécanismes de récurrence qui permettent de mieux appréhender les phénomènes temporels tel que peut l'être un problème de segmentation de la parole. Le modèle gamma, un modèle connexionniste à récurrence locale, a ainsi été choisi tout autant pour ses capacités à modéliser les évènements temporels que pour la facilité avec laquelle il peut être analysé. Il a été appliqué à des problèmes de reconnaissance de séquences, ce qui a permis d'explorer ses capacités, ainsi qu'à des tâches de segmentation, pour tenter de résoudre les problèmes posés par les perceptrons multicouches lors de l'utilisation de notre premier système.

MOTS-CLÉ

reconnaissance automatique de la parole (RAP) ; mots isolés, mots enchaînés et parole continue ; environnements bruités ; réseaux de neurones artificiels ; dynamique, temps et mémoire dans les réseaux de neurones artificiels récurrents.