Extraction de caractéristiques non linéaire et discriminante : application à la classification de phonèmes

Non linear and discriminant feature extraction applied to phonemes recognition

Bruno Gas, Mohamed Chetouani, Jean Luc Zarader
Université Pierre et Marie Curie-Paris 6, Groupe Perception et Réseaux Connexionnistes, EA 2385, Ivry sur Seine, F-94200 France

Manuscrit reçu le 1er octobre 2005

Résumé et mots clés

Nous proposons dans cet article une nouvelle méthode d'extraction de caractéristiques appliquée à la reconnaissance de phonèmes. Le modèle proposé : le codage neuronal prédictif (NPC pour Neural Predictive Coding) et ses deux déclinaisons NPC-2 et DFE-NPC (Discriminant Feature Extraction - NPC), est un modèle connexionniste de type perceptron multicouches (PMC) basé sur la prédiction non linéaire du signal de parole. Nous montrons qu'il est possible d'améliorer les capacités discriminantes d'un tel codeur en exploitant des informations de classe d'appartenance phonétique des signaux dès l'étape d'analyse. À ce titre, il entre dans la catégorie des extracteurs DFE déjà proposés dans la littérature. Dans cette étude, nous présentons une validation théorique du modèle dans l'hypothèse de signaux respectivement non bruités et bruités (bruit additif gaussien). Les performances de l'extracteur NPC pour la classification de phonèmes sont comparées avec celles obtenues par les méthodes traditionnellement utilisées en extraction de caractéristiques sur des signaux des bases Darpa Timit et Ntimit. Les simulations présentées montrent que les taux de reconnaissance sont nettement améliorés, en particulier dans le cas de phonèmes de la langue anglaise fréquents mais réputés délicats à catégoriser. Enfin, une application en reconnaissance de mots isolés et petit vocabulaire est présentée dans le but de montrer comment l'on peut insérer les paramètres NPC dans une application de reconnaissance à l'aide d'un système mixte ANN-HMM (Artificial Neural Networks – Hidden Markov Models).

Extraction de caractéristiques, réseaux de neurones prédictifs, traitement non linéaire du signal, reconnaissance de phonèmes.

Abstract and key words

In this article, we propose to study a speech coding method applied to the recognition of phonemes. The proposed model (the Neural Predictive Coding, NPC) and its two declinations (NPC-2 and DFE-NPC) is a connectionist model (multilayer perceptron) based on the non linear prediction of the speech signal. We show that it is possible to improve the discriminant capacities of such an encoder with the introduction of signal membership class information as from the coding stage. As such, it fits in with the category of DFE encoders (Discriminant Features Extraction) already proposed in literature. In this study we present a theoretical validation of the model in the hypothesis of unnoised signals and gaussian noised signals. NPC performances are compared to that obtained with traditional methods used to process speech on the Darpa Timit an Ntimit speech bases. Simulations presented here show that the classification rates are clearly improved compared to usual methods, in particular regarding phonemes considered difficult to process. A small vocabulary word recognition experiment is provided to show how NPC features can be used in a more conventional speech ANN-HMM based system approach.

Speech feature extraction, prédictive neural networks, nonlinear signal processing, phonemes recognition.

1. Introduction

Les systèmes de reconnaissance automatique de la parole (RAP) comportent pour la plupart trois principales étapes de traitement. La première concerne l'extraction de caractéristiques : l'estimation de vecteurs de paramètres sur des trames de parole d'environ 10 à 20 millisecondes comme, par exemple, les paramètres LPC (Linear Predictive Coding), MFCC (Mel Frequency Cepstrum Coding) ou PLP (Perceptual Linear Predictive). La deuxième réalise l'association de ces vecteurs acoustiques avec les modèles acoustiques, les phonèmes, les plus vraisemblables : les méthodes GMM (Modèles de Mixtures de Gaussiennes) ou connexionnistes sont souvent utilisées pour cela. Enfin, la troisième étape réunit ces hypothèses locales pour effectuer une recherche de la séquence de mots la plus probable à l'aide d'algorithmes HMM (Chaînes de Markov Cachées).
Les systèmes de reconnaissance les plus aboutis permettent d'obtenir aujourd'hui de bons résultats lorsque le signal de parole est « propre », c'est-à-dire non bruité ou correctement débruité. En revanche, dans des conditions dégradées, ce qui est malheureusement presque toujours le cas en situation réelle, leurs performances chutent fortement [27]. Cette baisse des performances trouve son origine dans l'importante *variabilité* du signal [35], une variabilité essentiellement due au message linguistique (celle que l'on exploite en reconnaissance de la parole et que l'on décrit au travers des phonèmes), due au locuteur (celle que l'on exploite en reconnaissance du locuteur) et due au canal (acoustique de l'environnement, bruits d'acquisition de transmission et de traitement).
Réduire la variabilité du signal de parole selon l'un au moins des trois axes précédents permet d'améliorer sensiblement les performances des systèmes de RAP. On distingue par exemple des applications à grand vocabulaire mais dépendantes du locuteur comme la dictée vocale, ou, *à contrario*, des applications indépendantes du locuteur mais à vocabulaire limité comme les répondeurs vocaux. Au demeurant, et comme le suggèrent plusieurs auteurs comme Bourlard [11] et Hermansky [30], réaliser des systèmes de RAP se rapprochant des performances humaines, au moins en ce qui concerne leur extraordinaire robustesse, nécessite de revisiter certaines des étapes de traitement. L'extraction de caractéristiques en est une et constitue le cœur de cet article dans lequel nous proposons d'aborder deux aspects : l'extraction de caractéristique *non linéaire* d'une part et *discriminante* d'autre part.

Plan de l'article

L'article s'articule autour de cinq sections incluant cette introduction. En section 2, nous donnons un aperçu de l'état de l'art des méthodes d'extraction de caractéristiques des signaux de parole utilisées dans les applications de reconnaissance. L'accent est plus particulièrement porté sur les méthodes d'analyse non linéaire d'une part, et explicitement discriminantes d'autre part. En section 3 nous définissons la version de base de l'extracteur NPC (Neural Predictive Coding). Nous montrons ainsi qu'il est possible d'extraire du signal de parole des informations non linéaires tout en générant des vecteurs de caractéristiques de dimension arbitraire. En section 4 nous introduisons les deux versions NPC-2 et DFE-NPC de l'extracteur ou son adaptation à la tâche de classification. Une validation expérimentale sur les bases de données TIMIT et NTIMIT est présentée. Enfin, nous terminons par une discussion du modèle où nous présentons des éléments de validation théorique en section 5 puis une application des paramètres NPC en reconnaissance de mots isolés à l'aide d'un système ANN-HMM.

2. L'état de l'art en extraction de caractéristiques pour la classification

Les méthodes d'extraction de caractéristiques héritent historiquement des méthodes de codage de la parole avec les paramètres LPC (Linear Predictive Coding) [33], [3]. Les besoins des systèmes de RAP diffèrent pourtant de ceux du codage de la parole. En général, les informations liées au locuteur et à l'environnement acoustique doivent être préservées en codage pour assurer la synthèse la plus fidèle possible. En RAP, il n'y a pas cette nécessité de reconstruire le signal de parole, l'objectif étant de retrouver le contenu linguistique ou l'identité du locuteur.

2.1. L'extraction de caractéristiques dans les systèmes de RAP

Au début des années 70, les résultats obtenus en reconnaissance de mots isolés avec les paramètres LPC ne sont pas jugés probants. Itakura propose donc une distance [32] permettant d'établir une règle de décision plus performante. Les paramètres NPC (Neural Predictive Coding) que nous décrivons dans cet article sont une extension au domaine non linéaire des paramètres LPC et la distance NPC que nous proposons est calquée précisément sur la distance d'Itakura.
Depuis les années 80, la question de l'extraction de caractéristiques est considérée généralement comme résolue, la raison principale étant que les paramètres les plus utilisés, les paramètres MFCC (Mel Frequency Cepstrum Coding), sont robustes et simples à mettre en oeuvre [17]. De plus, leur distribution statistique présente l'intérêt d'être localement gaussienne avec des coefficients faiblement corrélés, propriété largement

exploitée par les méthodes de modélisation/classification utilisées dans les étapes supérieures de traitement [35]. Basés sur une représentation non paramétrique du spectre court-terme, les coefficients MFCC tirent leur avantage du principe de déconvolution mis en œuvre et permettant de séparer les contributions de la source et du conduit vocal. Sont également prises en compte certaines caractéristiques auditives de l'oreille comme l'échelle non linéaire de MEL [61], [17] ou les effets de masquage [2]. Au début des années 90, [29] continue dans cette voie en proposant les paramètres PLP (Perceptual Linear Prediction) combinant plusieurs caractéristiques de la perception auditive : l'échelle de BARK notamment [73], mais également le filtrage RASTA qui est une généralisation du filtrage CMS (Cepstral Mean Substraction) pouvant s'appliquer dans l'espace des coefficients cepstraux, MFCC ou PLP [31].

L'ensemble de ces méthodes d'extraction réalisent donc, de différentes manières, une modélisation du spectre sur de courtes fenêtres de signal (mis à part le filtrage RASTA), exploitant la propriété de quasi-stationarité du signal de parole sur ces intervalles de temps. Par ailleurs, l'étape d'extraction de caractéristiques est toujours conçue indépendamment des étages supérieurs de traitement et fonctionne de façon parfaitement autonome. D'autres approches ont cependant été proposées, consistant à prendre en compte explicitement certains aspects non linéaires du processus de production de la parole, mais également à ne plus considérer l'analyse du signal indépendamment du reste du système de reconnaissance.

2.2. La modélisation non linéaire

Si l'on considère l'ensemble du canal de transmission dans la communication parlée humaine, il convient de considérer l'organe vocal, le milieu de transmission (l'air), l'environnement acoustique (le lieu), les éventuels systèmes électroniques de transmission utilisés (téléphone par exemple) et enfin l'organe auditif. Toutes ces étapes de la transmission sont autant de systèmes potentiellement non linéaires. Nous nous intéressons dans cet article à la modélisation non linéaire du système de production pour laquelle deux approches sont possibles selon que l'on connaît ou non ce système : l'étude du système lui-même ou l'étude du signal produit pas le système.

Loin du modèle source filtre souvent utilisé, l'organe de production vocal est un système complexe dans lequel les phénomènes non linéaires tiennent une place essentielle [43], [63]. Par exemple, avec la vibration des cordes vocales, la forme d'onde change avec l'amplitude du signal [62] [59] [60]. Des phénomènes de bifurcation existent au point que l'on caractérise les cordes vocales par leur diagramme de bifurcations [48]. Des *sous harmoniques* sont présentes, entraînant des phénomènes de quasi-périodicité car les deux cordes vocales ne vibrent pas exactement à la même fréquence. Des phénomènes de turbulence sont également à la source de bruits dans la génération de certains sons [51] comme les fricatifs par exemple. Ils surviennent dès qu'il y a une forte interaction entre la source d'air et les différents obstacles présents dans le conduit vocal : palais, lèvres, etc. Une synthèse sur ce thème a été publiée par Faundez *et al.* [21] dans le cadre de l'action européenne COST'277 sur le traitement non linéaire de la parole.

D'autres études, du signal vocal cette fois, montrent la présence de non linéarités dans le processus générateur. Thyssen *et al.* [65] ont expérimentalement montré la présence de telles non linéarités en estimant les paramètres d'un filtre linéaire tous pôles (analyse LPC) sur des trames de signal de 25ms. Une étude sur l'ordre du filtre doit alors être menée [25] afin d'estimer l'ordre optimal permettant d'obtenir la meilleure prédiction. Afin de s'assurer qu'il ne restait plus aucune information de type linéaire dans le signal d'erreur, Thyssen a proposé d'effectuer plusieurs analyses LPC en cascades. Il a ainsi montré que pour un signal voisé, le gain de prédiction du $5^{\text{ème}}$ filtre LPC arrive à $0\,dB$. Les paramètres d'un filtre non linéaire (typiquement un réseau de neurones prédicteur ou un filtre de Volterra d'ordre de prédiction identique) sont alors estimés sur la dernière erreur de prédiction. Le gain de prédiction non nul résultant confirme bien la présence de non linéarités, en tous cas d'informations « non captées » par les filtres linéaires. Cet argument donné par Thyssen a été repris à plusieurs reprises dans la littérature puisqu'on le retrouve cité dans de nombreuses publications [43], [21], [8], [19], [50], [49]. D'autres auteurs comme Townshend [66] avaient déjà montré quelques années auparavant, et à l'aide de méthodes similaires, qu'un prédicteur non linéaire peut extraire des informations complémentaires (de l'ordre de 2,8 dB à 3 dB de plus sur les mesures effectuées), y compris à partir de séries chaotiques [40].

Une étude reportée par Kubin dans [43], basée en particulier sur la méthode des *données synthétiques de remplacement* ou *surrogate data* [64], a montré que l'apport des prédicteurs non linéaires concerne surtout les sons voisés. Le modèle source-filtre, où la source est un bruit blanc gaussien et le filtre un modèle AR, s'avère remarquablement bien adapté à la description des sons non voisés. Ces derniers peuvent donc, de ce fait, être considérés comme gaussiens et linéaires. En revanche, ce n'est pas le cas des sons voisés qui sont non gaussiens et ne peuvent être correctement décrits par un système prédictif linéaire AR. Les sons voisés constituant environ 60 % du signal, le remplacement des filtres linéaires par des opérateurs non linéaires devrait permettre d'obtenir une amélioration significative des performances.

Les filtres de Volterra constituent un premier type de filtre non linéaire déjà étudié dans la littérature [16]. Leur principal avantage tient au fait que l'équation des moindres carrés pour le calcul des coefficients admet une solution analytique. En revanche, dans une application d'extraction de caractéristiques, ils présentent l'inconvénient majeur de voir le nombre de paramètres libres augmenter très rapidement avec la taille de la fenêtre de prédiction [65]. Un filtre de Volterra requiert 65 coefficients pour un horizon de prédiction de 10 échantillons là où un filtre AR n'en requiert que 10 puisque le nombre de coefficients y est égal à l'ordre du prédicteur.

Les réseaux de neurones peuvent également être considérés comme des filtres adaptatifs non linéaires [20]. Ainsi que l'ont montré Lapedes et Farber [46], un réseau multicouches (PMC), entraîné avec l'algorithme de rétropropagation, fournit de remarquables résultats en prédiction. Mais, appliqué à l'extraction de caractéristiques, cette solution conduit au même problème de démultiplication des paramètres puisque la présence d'au moins une *couche cachée* est obligatoire.
Nous montrons dans cet article qu'il existe une solution à ce problème : une structure PMC, adéquatement modifiée, permet de réaliser un extracteur de caractéristiques non linéaire dont le nombre de paramètres, quelconque, peut être choisi indépendamment de l'horizon de prédiction.

2.3. Extraction discriminante de caractéristiques

Un certain nombre d'idées ont déjà été proposées pour améliorer la qualité de l'extraction de caractéristiques, comme indiqué au paragraphe 2.1. Même si elle augmente considérablement le nombre de paramètres, la méthode consistant à incorporer des caractéristiques dynamiques (coefficients Δ et $\Delta\Delta$) [22] permet d'améliorer sensiblement les scores en reconnaissance. La dimension élevée du vecteur généré conduit cependant à utiliser des méthodes de réduction de dimension comme l'analyse en composantes principales (ACP), en composantes indépendantes (ACI) [47], ou encore l'ACP non linéaire à l'aide de réseaux de neurones associatifs [13]. Ces analyses, que l'on peut considérer comme une étape supplémentaire de traitement, gagnent à être également discriminantes. On utilise alors avantageusement l'analyse discriminante (AD) [39, 38]. Zahorian *et al.* [72] ont amélioré les capacités discriminantes des HMMs en utilisant cette technique. Les auteurs ont montré qu'à partir d'un ensemble de 30 mots d'une syllabe, on pouvait obtenir une amélioration de près de 25 % des taux de reconnaissance. Des extensions intéressantes ont également été proposées [58] dans le cas de classes inégalement réparties. Cependant, l'analyse AD a montré des gains importants pour des petits vocabulaires mais des résultats plus mitigés pour des vocabulaires moyens. Notons qu'elle peut être étendue au domaine non linéaire en utilisant un perceptron multi-couches (Non Linear Discriminant Analysis, NLDA) [57]. Cette étape de réduction *discriminante* de la dimension des données s'insère entre la phase d'extraction de caractéristiques proprement dite et la phase de classification qui suit. L'ensemble constitué par l'extraction et la réduction de dimension est en quelque sorte adapté à la tâche de classification, mais ces deux étapes restent traitées séparément. Plus récemment, Povey *et al.* [54] ont publié une méthode d'extraction-compensation de caractéristiques très remarquée du fait des progrès significatifs qu'elle permet d'obtenir en reconnaissance de la parole conversationnelle. Il s'agit des paramètres fMPE (feature-space minimum phone error) qui sont définis par :

$$\mathbf{a}'_t = \mathbf{a}_t + \mathbf{M}\mathbf{h}_t \qquad (1)$$

où $\mathbf{a}_t$ représente le vecteur acoustique original au temps t (paramètres PLP), $\mathbf{h}_t$ un vecteur de caractéristiques intermédiaire de très grande dimension dont les éléments sont calculés à partir des paramètres $\mathbf{a}_t$ (vraisemblances calculées à partir d'un très grand ensemble de lois gaussiennes) et $\mathbf{M}$ une matrice de projection dont l'objet est de projeter le vecteur $\mathbf{h}$ dans le sous espace initial de faible dimension des vecteurs acoustiques. $\mathbf{M}$ est déterminée par minimisation du critère connu sous le nom de *critère MPE* (Minimum Phone Error) [55]. La forme additive $\mathbf{a}_t + \mathbf{M}\mathbf{h}_t$ est motivée par la nécessité de démarrer l'apprentissage avec une solution initiale correcte (celle donnée par les vecteurs caractéristiques PLP). Des améliorations de près de 2 points ont été reportées sur les données de l'évaluation NIST RT-04 (Rich Transcription 2004).
Le vocable de *Discriminant Feature Extraction* (DFE) fait son apparition au début des années 90 avec les travaux de Juang et Katagiri [34]. Les méthodes DFE sont des méthodes d'extraction de caractéristiques, en général adaptatives, exploitant les informations de catégorisation des signaux en classes (les classes phonétiques par exemple). Même si l'extraction et la classification peuvent toujours être traitées séparément [18], la théorie MCE/GPD (Minimum Classification Error/Generalized Probabilistic Descent) proposée par Juang et Katagiri [34, 36] donne un cadre théorique tel que l'espace de représentation des caractéristiques (l'espace des paramètres acoustiques) est déterminé en prenant en compte l'ensemble de la problématique paramétrisation et classification. Biem et Katagiri [6, 7, 5] ont utilisé ce cadre pour proposer l'adaptation d'un bancs de filtres (distribution des fréquences et des largeurs des bancs de filtre) ainsi que la conception d'un filtre cepstral. Bachianni *et al.* [4] ont également proposé d'utiliser les techniques DFE pour l'optimisation de filtres de masquage temps-fréquence.
L'extracteur de caractéristiques NPC que nous proposons [23] permet également de prendre en compte les informations de classe d'appartenance des signaux [24]. Nous montrons sur une application de reconnaissance de phonèmes que cette adaptation de l'extracteur au problème de classification permet d'améliorer les scores en reconnaissance. Le formalisme utilisé ici n'est cependant pas celui proposé par la théorie MCE/GPD mais suit le principe de la maximisation du critère de l'Information Mutuelle Maximale (critère MMI).

3. Codage prédictif neuronal : le modèle NPC

Comme l'indique la figure 1, l'extraction de caractéristiques s'effectue sur des trames entrelacées d'une durée de 10 à 20ms. Le signal est alors supposé stationnaire et une estimation, paramétrique ou non, du spectre court-terme peut être effectuée. On obtient un vecteur caractéristique de la trame dont les composantes, ou paramètres, sont le plus souvent au nombre de 12 ou 16.

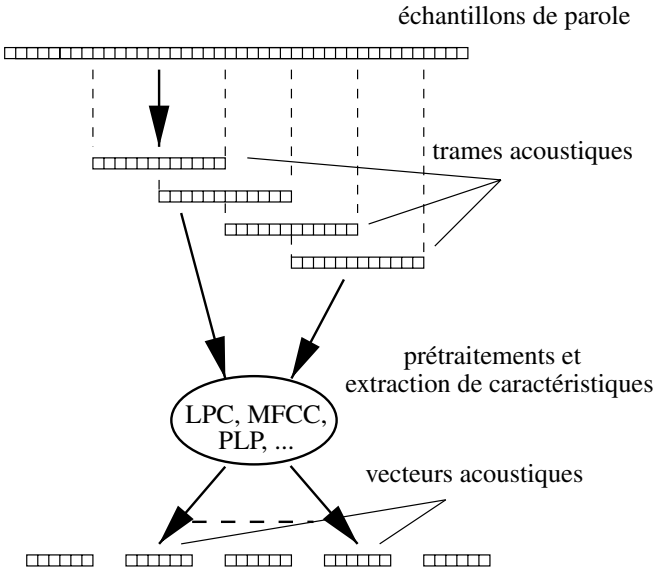

Figure 1. Extraction de caractéristiques dans les systèmes de RAP.

Le modèle LPC consiste à considérer le signal présent dans la trame comme un processus stochastique dont le modèle serait du type source-filtre, la source étant un bruit blanc centré, gaussien :

$$y_k = \sum_{i=1}^{\lambda} a_i\, y_{k-i} + \varepsilon_k. \tag{2}$$

3.1. Principe

Le modèle NPC (Neural Predictive Coding) est une extension du codage linéaire prédictif LPC au domaine non linéaire. Le modèle est toujours basé sur la représentation source-filtre du conduit vocal à ceci près que le filtre est remplacé par un réseau de neurones PMC (Perceptron Multicouches). Cette approche du filtrage non linéaire à déjà été étudiée par Lapedes et Farber [46] ainsi que par Dreyfus *et al.* [20]. Ainsi, le signal prédit $\hat{y}_k$ à l'instant k s'écrit :

$$\hat{y}_k = \sum_{i=1}^{n} a_i \left[\sigma\left(\sum_{j=1}^{\lambda} \omega_{ij}\, y_{k-j} \right) \right] \tag{3}$$

où les ω_{ij} désignent les poids de la première couche (couche cachée) et les a_i les coefficients synaptiques de la cellule de sortie. Le réseau de neurones étant dédié à une tâche de prédiction à une dimension, il ne comporte qu'une seule cellule de sortie et la fonction de transition de cette cellule est linéaire (quoique cela n'est pas obligatoire). Le nombre n de cellules de la couche cachée est un paramètre libre du modèle mais le nombre des entrées de ces cellules correspond à l'horizon de prédiction λ.

L'extraction de caractéristiques sur une trame de signal à l'aide d'un tel modèle consiste à estimer les poids du réseau minimisant l'erreur de prédiction $\epsilon_k = \hat{y}_k - y_k$ où $\hat{y}_k = F_{\Omega,\mathbf{a}}(\mathbf{y}_k)$ représente la sortie calculée par le réseau à l'instant k lorsque l'on présente en entrée le vecteur $\mathbf{y}_k = [y_{k-1}, \ldots, y_{k-\lambda}]$ des λ échantillons précédents. Ω représente l'ensemble des poids de la couche cachée et $\mathbf{a}$ le vecteur des poids de la cellule de sortie. La fonction de coût est l'erreur quadratique moyenne calculée sur les $D - \lambda$ derniers échantillons, où D est le nombre d'échantillons de la trame :

$$Q(\Omega,\mathbf{a}) = \frac{1}{2} \sum_{k=\lambda}^{D-1} \epsilon_k^2 \tag{4}$$

Enfin, l'algorithme d'apprentissage utilisé est par exemple l'algorithme de rétropropagation du gradient.

Après apprentissage, lorsque la fonction de coût Q a atteint un minimum, l'ensemble des poids du réseaux $\{\Omega^\star, \mathbf{a}^\star\}$ caractérisent la trame et constituent un ensemble de paramètres représentatifs. Ainsi que Thyssen le souligne [65], le nombre de poids augmente rapidement avec la taille de la fenêtre de prédiction λ et rend cette solution impraticable. Par exemple, une structure de réseau $16 \times 8 \times 1$ conduit à 145 paramètres (sont considérés l'ensemble des poids et seuils du réseaux) contre 16 paramètres pour un filtre LPC d'ordre 16. Thyssen a proposé une structure de réseau dite *à poids partagés*, reprenant le principe du partage des poids des réseaux TDNN (Time Delay Neural Network) de Waibel et Lang [68, 45] et permettant de réduire substantiellement le nombre de paramètres libres du système.

La solution que nous proposons s'appuie sur un résultat obtenu par Kolmogorov en 1951 [42] concernant le principe de superposition de fonctions et mis en œuvre dans le cadre des réseaux de neurones PMC par Hecht-Nielsen [28] en 1987 : tous les poids du réseau ne « codent » pas la même information, en particulier seuls les poids de la couche de sortie codent les informations caractéristiques de la trame (voir le paragraphe 5 pour une discussion à ce sujet). Avec le modèle NPC, seuls les poids de la couche de sortie, les coefficients a_i dans l'équation (3), sont les paramètres libres de l'algorithme d'apprentissage et deviennent donc, après convergence, les paramètres caractéristiques. L'horizon de prédiction λ, c'est-à-dire le nombre d'entrées du réseau, ne conditionne plus la dimension du vecteur des paramètres, laquelle devient seulement dépendante du nombre n de cellules cachées. Les poids de la première couche Ω sont déterminés lors d'une phase de calcul appelée *phase de réglage*, préalable nécessaire à l'extraction de caractéristiques. Le réglage de l'extracteur est réalisé une fois pour toute et peut faire intervenir des informations de classe d'appartenance des signaux (méthode DFE). Il a alors lieu le plus souvent durant la phase d'apprentissage du classifieur amont. Nous présentons trois algorithmes de réglage dans cette article appelés respectivement NPC, NPC-2 et DFE-NPC.

3.2. Algorithme NPC

La phase de réglage NPC est indépendante du problème de classification. Sur une base d'apprentissage représentative des signaux de parole traités, on considère un réseau PMC par trame de signal. La structure est du type *poids partagés* pour les poids Ω de la couche cachée et poids indépendants $\mathbf{a}_l$ pour chacune des trames l de la base. La figure 2 représente la structure globale de l'extracteur en phase de réglage.

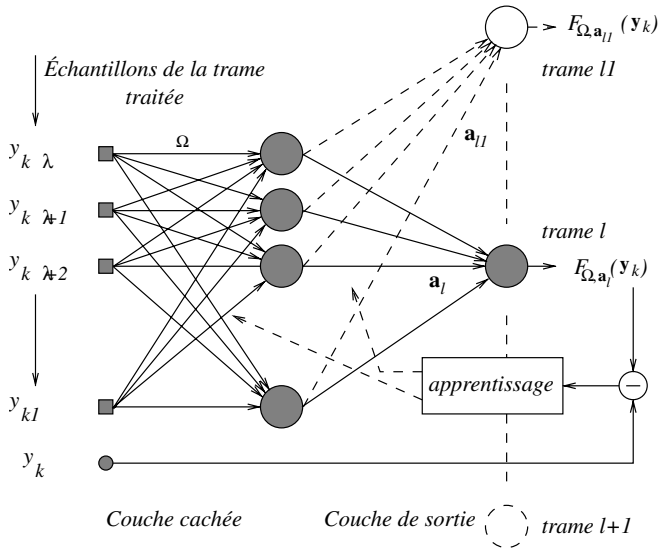

Figure 2. Structure de l'extracteur NPC.

La fonction de coût globale à minimiser, pour un ensemble de L trames d'apprentissage, s'écrit :

$$Q(\Omega,\mathbf{a}_{1,\ldots,L}) = \frac{1}{2}\sum_{l=1}^{L}\sum_{k}\sum_{i=1}^{L}(\epsilon_{l,k}^{i})^2\delta_{i-l} \tag{5}$$

où $\epsilon_{l,k}^{i} = F_{\Omega,\mathbf{a}_i}(\mathbf{y}_{l,k}) - y_{l,k}$ représente l'erreur de prédiction i du réseau lorsque l'on présente le vecteur des échantillons passés $\mathbf{y}_k$ de la trame l. δ est le symbole de Kronecker permettant de sélectionner la seconde couche $i = l$ lorsque la trame l est traitée. La règle d'apprentissage est du type règle Madaline III modifiée (MRIII) [70]. Compte tenu du principe du partage des poids de la première couche, les vitesses d'apprentissage des première et seconde couches sont pondérées de sorte à équilibrer les temps de convergence :

$$\begin{cases} \omega_{ij} & \longleftarrow & \omega_{ij} - \dfrac{\theta_0}{L}\dfrac{\partial Q}{\partial \omega_{ij}} \\ a_i & \longleftarrow & a_i - \theta_0\dfrac{\partial Q}{\partial a_i} \end{cases} \tag{6}$$

Maintenir ainsi une couche de poids Ω commune à l'ensemble des trames entraîne un temps de convergence plus long, coûteux en temps de calcul. Lorsque cette phase de réglage est terminée, seuls les poids Ω sont conservés et la phase d'extraction de caractéristiques peut intervenir, beaucoup plus rapide. Précisons que lorsque les fonctions de transition des cellules de sortie sont linéaires, l'extraction de caractéristiques peut s'effectuer à l'aide d'une méthode d'optimisation linéaire, rendant le calcul équivalent à celui de l'extraction des paramètres LPC.

3.3. Gains de prédiction

Nous avons effectué des mesures du gain de prédiction moyen d'un codeur NPC après réglage de la couche cachée sur une base comportant 6 classes de phonèmes : /*s*/, /*z*/, /*ah*/, /*ih*/, /*aa*/ et /*iy*/ à raison de 1 000 trames de 256 échantillons extraits de la base NTIMIT (*cf.* paragraphe 3.4.). La comparaison avec un filtre LPC à 12 coefficients montre que le codeur NPC a un gain de prédiction supérieur en moyenne sur l'ensemble des phonèmes testés, ainsi que l'indique la table 1. Ces gains de prédiction apparaissent moins élevés que ceux trouvés dans la littérature : Thyssen dans [65] reporte des gains allant de $14.1dB$ pour un prédicteur LPC jusqu'à $19.3dB$ pour un prédicteur non linéaire PMC, Diaz [19] reporte des gains passant de $13.6\ dB$ à $15.1\ dB$ et Townshend dans [66] une augmentation moyenne de $2.8\ dB$. La cause principale en est qu'ils ont été calculés sur des signaux de parole bruités et à bande limitée puisqu'il s'agit de signaux téléphoniques.

Tableau 1. Gains de prédiction moyens calculés pour les 6 classes de phonèmes extraits de la base NTIMIT.

Phonèmes	LPC (12)	NPC (12)	Amélioration
/*s*/	3.90	4.04	+3.59 %
/*z*/	3.68	3.88	+5.43 %
/*ah*/	8.38	8.84	+5.49 %
/*ih*/	5.97	6.22	+4.18 %
/*aa*/	10.64	11.44	+7.52 %
/*iy*/	5.93	6.26	+5.56 %
app.	6.45	6.79	+5.27 %
test	6.39	6.77	+5.90 %

Il convient de noter que ces mesures ne suffisent pas à conclure que les paramètres NPC modélisent des caractéristiques spécifiquement non linéaires. En effet, à ordre de prédiction égal (12 dans le cas qui nous intéresse), le nombre de paramètres libres n'est pas identique d'un codeur à l'autre : 12 également dans le cas du filtre LPC mais $12 \times 12 + 12 = 156$ dans le cas du filtre NPC. L'amélioration du gain de prédiction apportée par le codeur NPC peut être tout autant due à sa complexité, nettement supérieure à celle du LPC (156 paramètres libres contre 12). Un

avantage certain de la structure NPC est précisément de pouvoir augmenter l'ordre de prédiction du filtre, sans pour autant modifier la dimension du vecteur des paramètres NPC. À titre d'exemple, la table 2 indique les résultats obtenus pour différentes tailles de la fenêtre de prédiction mais un même nombre de coefficients du vecteur caractéristique égal à 12. On observe bien une sensible augmentation du gain de prédiction avec le nombre de paramètres libres du filtre NPC. En revanche, l'analyse faite par Kubin dans [43] selon laquelle les phénomènes non linéaires sont plus à chercher du côté des signaux voisés que non voisés est corroborée par le tableau 1 : l'augmentation du gain de prédiction apportée par le modèle NPC est plus faible pour le seul phonème non voisé $/s/$ que pour l'ensemble des autres phonèmes voisés. Des mesures portant sur d'autres phonèmes non voisés ($/sh/$ et $/f/$ par exemple avec 2.85 % et 1.9 % d'augmentation du gain de prédiction respectivement) montrent des résultats similaires. Ce fait devrait cependant être confirmé par une étude exhaustive portant sur l'ensemble des phonèmes voisés et non voisés.

Tableau 2. Gains de prédiction moyens calculés sur les six phonèmes extraits de la base NTIMIT pour différentes valeurs de la fenêtre de prédiction.

Mémoire de prédiction	16	20	25	40
Paramètres libres	204	252	312	492
Gain de prédiction (dB)	6.63	6.78	6.82	6.92

3.4. Conditions expérimentales

Dans le but d'évaluer les performances des divers extracteurs NPC présentés dans cet article, nous avons construit plusieurs bases de phonèmes, toutes extraites des bases de données parole Darpa-TIMIT et Darpa-NTIMIT [1].

Les signaux de parole utilisés. La base TIMIT a été développée par le Massachusetts Institute of Technology (MIT), le Standford Research Institute (SRI) et Texas Instrument (TI). Il s'agit d'une base de parole continue multilocuteurs composée de 10 phrases prononcées par 630 locuteurs (192 hommes et 438 femmes) de 8 régions dialectales des États-Unis. Les enregistrements ont été effectués à l'aide de microphones de qualité et à une fréquence d'échantillonnage de 16kHz (16bits). La base NTIMIT a été obtenue par le passage de la base TIMIT à travers un réseau téléphonique (appels locaux et longue distance). La bande passante est donc réduite entre 330Hz et 3400Hz. Les performances des systèmes de reconnaissance sont très dégradées sur cette base [52], [71]. Elle constitue donc un outil intéressant d'évaluation des codeurs NPC et de leur robustesse.

Nous avons nommé B1 à B4 les quatre bases que nous exploitons ici. La première, B1, regroupe quatre classes de phonèmes voisés (voyelles) très couramment utilisées : $/aa/$, $/ae/$, $/ey/$ et $/ow/$. Elles ont été extraites de la base TIMIT. Les bases B2 et B3 regroupent deux séries de phonèmes extraits de la base TIMIT également : $/b/$, $/d/$, $/g/$ (plosives voisées) et $/p/$, $/t/$, $/k/$ (plosives non voisées). Ces deux bases sont d'un intérêt particulier pour l'évaluation des méthodes de paramétrisaton et de classification pour le traitement de la parole : elles sont fréquemment utilisées et simultanément difficiles à traiter. Elles ont été utilisées par Waibel et Lang [68], [45] pour valider leur modèle TDNN (Time Delay Neural Network). B4, la quatrième base, comporte six classes de phonèmes $/s/$, $/z/$, $/aa/$, $/ah/$, $/iy/$, $/ih/$ extraits de la base téléphonée NTIMIT. Nous l'avons utilisé pour tester l'extracteur NPC sur des signaux à bande passante limitée. Étant donné qu'elle est composée de trois sous-classes de phonèmes, cette base permet de mettre en évidence les capacités discriminantes d'un extracteur : les trois sous-classes sont bien séparées et les phonèmes appartenant à une même sous-classe sont accoustiquement très proches. Nous avons subdivisé chacune des bases B1 à B4 en deux sous-bases : l'une pour le réglage de l'extracteur et l'apprentissage du classifieur (Bi, $i = 1 \ldots 4$) et l'autre pour les tests en généralisation (BiT, $i = 1 \ldots 4$). La table 3 résume les caractéristiques des quatre bases ainsi constituées.

Tableau 3. Principales caractéristiques des bases utilisées : nom des bases, base d'origine, fréquence d'échantillonnage, nombre de trames et longueur des trames.

B1	B2	B3	B4
Timit	Timit	Timit	Ntimit
16KHz	16KHz	16KHz	8KHz
B1L/B1T	B2L/B2T	B3L/B3T	B4L/B4T
500 / 500	500 / 500	500 / 500	500 / 500
256 / 256	256 / 256	256 / 256	128 / 128

La méthode d'évaluation. Le problème posé est de comparer l'efficacité des paramètres NPC relativement à d'autres paramètres de la littérature. Nous avons procédé à une évaluation « locale » des scores obtenus, en reconnaissance de phonème, à l'aide d'un classifieur. Les paramètres testés (LPC, MFCC et NPC) l'ont été dans des conditions expérimentales rigoureusement identiques : 12 coefficients caractéristiques, classifieur PMC comportant 12 entrées, 10 cellules cachées et autant de sorties que de classes de phonèmes (apprentissage du type descente de gradient par minimisation de l'erreur quadratique moyenne avec algorithme de rétropropagation).

Résultats expérimentaux. Les résultats obtenus avec le codeur NPC montrent une amélioration des scores en classification de phonèmes comparativement aux principales méthodes d'analyse utilisées. La table 4 montre les scores obtenus avec la base de généralisation B4T comparés avec les codages LPC et MFCC.

Tableau 4. Taux de reconnaissance obtenus en utilisant les trois paramètres LPC, MFCC et NPC.

Paramètres	LPC	MFCC	NPC
Scores (%)	55.5	58	**61.2**

Comme dans tout système à apprentissage, il est nécessaire d'évaluer les performances du système en *généralisation*, c'est-à-dire lorsque l'on présente des données qui n'ont pas servi lors de l'apprentissage. La phase de réglage ne présente pas de phénomène de sur-apprentissage : augmenter le nombre d'itérations d'apprentissage conduit à améliorer les scores sur la base de test B4T. À titre indicatif, les scores reportés table 4 ont été obtenus après 5 000 itérations d'apprentissage.

Tableau 5. Performances du NPC (en %) pour plusieurs nombres d'itérations (1ère ligne) du processus d'extraction.

1	5	10	20	40	100	250	500
60.1	**61.2**	**61.2**	60	59.6	59.8	59.7	59.7

Les mêmes expérimentations ont cependant révélé l'existence d'un problème de sur-apprentissage durant la phase d'extraction de caractéristiques. Nous avons extrait les codes générés à plusieurs étapes successives de cet apprentissage pour établir l'évolution des scores en fonction du nombre d'itérations pratiquées : la courbe obtenue présente un maximum comme le montre le tableau 5. Par ailleurs, les meilleurs résultats s'obtiennent avant les 20 premières itérations. Ce nombre très faible s'explique car l'apprentissage est du type *gradient stochastique*. Une itération représente donc l'apprentissage d'une trame, soit la présentation de $D - \lambda - 1$ échantillons. Nous retrouverons ce problème lors de l'étude des méthodes de réglage NPC-2 et DFE-NPC abordées dans les paragraphes suivants et nous en proposerons une explication au paragraphe 5.3.

4. Extraction discriminante de caractéristiques

Ainsi que l'expliquent Zahorian *et al.* dans [72], l'extraction de caractéristiques doit être telle que les vecteurs acoustiques appartenant à la même classe soient proches au sens d'une certaine distance définie dans leur espace de représentation, tandis que des vecteurs de classes différentes soient éloignés dans ce même espace. Si l'on admet que les vecteurs caractéristiques suivent une distribution gaussienne, l'extraction de caractéristiques *discriminante* (DFE) consistera à minimiser la covariance intra-classe et à maximiser simultanément la covariance inter-classes. Cette opération pourra s'effectuer explicitement sous la forme de l'ajout d'un terme de discrimination à la fonction de coût par exemple, lors du réglage de l'extracteur.
Nous avons donc proposé dans [24] le modèle NPC-C (paramètres NPC-*Contraints*) dont la fonction de coût s'écrit :

$$Q^{NPC-C} = \alpha Q^M + (1-\alpha) Q^D \tag{7}$$

où Q^M et Q^D désignent respectivement les coûts de *modélisation*, c'est-à-dire la minimisation de l'erreur de prédiction, et de *discrimination* : minimisation des variances intra-classe et maximisation des variances inter-classes dans le cas simplifié où les distributions sont gaussiennes sphériques :

$$Q^D = \beta \sum_{c=1}^{N_C} \sigma_c^2 + \gamma \frac{1}{\Sigma^2} \tag{8}$$

σ_c^2 désigne la variance intra-classe estimée sur l'ensemble des trames d'apprentissage réparties en N_C classes et Σ^2 la variance inter-classes estimée également sur l'ensemble des signaux d'apprentissage. La minimisation de Q^D s'apparente à celle du logarithme de l'inverse du F-ratio, (le F-ratio est défini comme le rapport de la variance inter-classe à la variance intra-classe et mesure la discrimination entre les classes qu'un paramètre est apte à réaliser) : $\log 1/F = \log \sigma^2/\Sigma^2 = \log \sigma^2 + \log 1/\Sigma^2$. Les résultats obtenus ne permettent pas, en général, d'obtenir une amélioration significative des scores en reconnaissance. On observe que la première couche n'est que très peu influencée par la contrainte discriminante. La raison principale est à chercher dans la nature approximative des hypothèses faites comme celle de la densité supposée gaussienne et sphérique. La prise en compte d'hypothèses moins restrictives permet d'établir des contraintes plus optimales mais au prix d'une complexité accrue de l'étape de réglage. Dans les méthodes DFE, les contraintes optimales restent celles définies par le classifieur lui-même. La mise en œuvre de tels systèmes nécessite un cadre formel particulier, comme celui donné par la théorie MCE/GPD [5, 36], et dans lequel il devient possible de faire coopérer le classifieur avec l'extracteur de caractéristiques lors de son réglage. Nous proposons une telle version de l'extracteur dans un article à venir. La méthode de réglage que nous présentons ci-dessous est indépendante du classifieur utilisé et minimise toujours l'erreur quadratique de prédiction. En revanche la structure du codeur (le nombre de cellules de sortie durant la phase de réglage) se trouve être modifiée de sorte à prendre en considération l'appartenance des signaux à des catégories phonétiques.

4.1. Paramètres NPC-2

Le modèle NPC-2 est une extension du modèle NPC qui permet la prise en compte des classes d'appartenance des signaux pendant la phase de réglage, mais sans imposer de contraintes

explicites lors de l'apprentissage. La simplicité de mise en oeuvre du réglage NPC-2 contraste avec celle des paramètres NPC-C vus plus haut : il se résume en une simple modification de la structure du prédicteur. La notion de partage des poids de la première couche est étendue à celle de partage des poids de la deuxième couche entre les signaux appartenant à une même classe. Ainsi, le prédicteur ne comporte plus qu'une cellule de sortie par classe au lieu d'une cellule de sortie par trame.

Définition du modèle. Supposant que c_l est la classe d'appartenance de la trame l parmi un ensemble de N_C classes possibles, la fonction de coût devient :

$$Q^{NPC2}(\Omega, \mathbf{a}_{1,\ldots,N_C}) = \frac{1}{2}\sum_{l=1}^{L}\sum_{k}\sum_{i=1}^{N_C}(\epsilon_{l,k}^{i})^2\delta_{i-C(l)} \qquad (9)$$

où $\epsilon_{l,k}^{i}$ est l'erreur de prédiction de la sortie i du réseau lorsque l'on présente le vecteur des échantillons passés $\mathbf{y}_k$ de la trame l (voir eq. 5). $\delta_{i-C(l)}$ est le symbole de Kronecker permettant de sélectionner la seconde couche $i = C(l)$ lorsque la classe de la trame traitée l est $C(l)$.

Résultats expérimentaux. Sur une simulation de reconnaissance de phonèmes (bases B1, B2 et B3), nous avons comparé quatre extracteurs de caractéristiques dans les mêmes conditions expérimentales : LPC, MFCC, NPC et NPC-2. Les taux de reconnaissance ont été obtenus après 30000 itérations d'apprentissage du classifieur. Nous avons reporté les résultats table 6 (voyelles, plosives voisées et plosives non voisées).

Tableau 6. Taux de reconnaissance obtenus (en généralisation) par un classifieur PMC sur les bases B1, B2 et B3 pour plusieurs jeux de paramètres : LPC, MFCC, NPC et NPC-2.

bases	LPC	MFCC	NPC	NPC-2
B1	55.5 %	58 %	61.2 %	**63 %**
B2	57.5 %	62.3 %	65 %	**70.2 %**
B3	61 %	**68 %**	61.5 %	65 %

On peut noter que les résultats sur les phonèmes /p/,/t/,/k/ (base B3) sont relativement proches pour toutes les méthodes de codage utilisées (LPC, MFCC, NPC, NPC-2). Par ailleurs, la méthode NPC-2 donne de meilleurs scores pour les phonèmes /aa/, /ae/, /ey/, /ow/ (base B1) et /b/, /d/, /g/ (base B2) : plus de 60 à 65 % de reconnaissance contre 58 à 62 % avec les paramètres MFCC. Ces phonèmes sont voisés tandis que les phonèmes /p/, /t/, /k/ ne le sont pas. Ainsi que nous l'avons souligné au paragraphe 2.2, les phonèmes non voisés sont correctement représentés par le modèle linéaire source-filtre. L'extracteur non linéaire n'apporte donc pas d'informations supplémentaires significatives. C'est ensuite sur l'aspect discriminant que nous nous sommes focalisés avec le modèle DFE-NPC présenté ci-dessous pour améliorer l'extraction de caractéristiques des phonèmes non voisés.

4.2. Les paramètres DFE-NPC

Les méthodes DFE (Discriminant Feature Extraction) introduites par Juang et Katagiri [34, 36], caractérisent les dispositifs d'extraction de caractéristiques explicitement guidés par la tâche de classification. Contrairement aux auteurs, nous n'utilisons pas ici la théorie MCE-GPD : les paramètres DFE-NPC sont en effet générés sur la base d'une nouvelle fonction de coût permettant de s'affranchir de tout classifieur pendant la phase de réglage.

Définition du modèle. La fonction de coût que nous proposons d'utiliser, appelée le *rapport d'erreur de modélisation* ou MER (*Modelization Error Ratio*), généralise la fonction de coût NPC-2 en maximisant l'erreur de prédiction d'une trame sur les cellules de sortie ne représentant pas la même classe. La minimisation de l'erreur de prédiction sur la cellule représentant la bonne classe est bien entendu conservée :

$$\Gamma(\Omega, \mathbf{a}_{1,\ldots N_C}) = \frac{1}{N_C - 1}\frac{\sum_{l,k}\sum_{i\neq C(l)}(\varepsilon_{l,k}^{i})^2}{\sum_{l,k}\sum_{i}(\varepsilon_{l,k}^{i})^2\delta_{i-C(l)}} \qquad (10)$$

Au numérateur, on trouve la somme des erreurs de prédiction obtenues lorsque l'on présente les trames de paroles l de la base d'apprentissage mais que l'on utilise les poids-codes $\mathbf{a}_i$ qui ne correspondent pas à leur classe d'appartenance : $i \neq C(l)$. Le dénominateur représente la somme des erreurs de prédiction obtenues lorsque l'on présente les trames de paroles l et que l'on utilise les poids-codes $\mathbf{a}_{i,i=C(l)}$ qui correspondent à leur classe d'appartenance $C(l)$. Le rapport $1/(N_C - 1)$ joue le rôle de facteur de normalisation permettant d'accorder une importance équivalente à l'erreur de modélisation (numérateur) et à l'erreur de discrimination (dénominateur). Plus l'amplitude du MER est élevée, plus l'extracteur s'avère discriminant au sens d'une classification qui aurait lieu par minimisation de l'erreur de prédiction, ainsi que nous le montrons au paragraphe 5.3. *A contrario*, un rapport proche de 1 témoigne d'un extracteur non discriminant au sens où aucune association n'est faite entre l'erreur de prédiction calculée sur les sorties du réseau et la classe d'appartenance des trames. Afin d'obtenir une fonction de coût à minimiser, on considère le plus souvent l'inverse du logarithme du MER :

$$Q^{DFE-NPC}(\Omega, \mathbf{a}_{1,\ldots N_C}) = \frac{1}{\log\Gamma} \qquad (11)$$

Règles d'adaptation DFE-NPC. La loi de modification des poids ω du réseau est issue de la minimisation de la fonction de coût $Q^{DFE-NPC}$ par la méthode de descente du gradient en utilisant l'algorithme de rétropropagation. Pour tous les poids ω du réseau (couche cachée et couche de sortie), nous avons à l'itération d'apprentissage q :

$$\Delta\omega(q) = -\mu\frac{\partial}{\partial\omega}\left(\frac{1}{\log\Gamma}\right) \qquad (12)$$

où μ est le pas d'apprentissage, éventuellement adaptatif. Il vient :

$$\Delta\omega(q) = \mu \frac{\partial}{\partial \omega} \left(\log \Gamma \right) = \mu \frac{1}{\Gamma} \frac{\partial \Gamma}{\partial \omega} \tag{13}$$

En détaillant le MER comme le rapport de l'erreur de modélisation sur l'erreur de discrimination $\Gamma = Q^D / Q^M$, nous obtenons :

$$\Delta\omega(q) = \mu \left(\frac{1}{Q^D} \frac{\partial Q^D}{\partial \omega} - \frac{1}{Q^M} \frac{\partial Q^M}{\partial \omega} \right) \tag{14}$$

Le deuxième terme n'est autre que la modification consécutive à la minimisation de l'erreur de prédiction, c'est-à-dire la règle NPC-2. Le premier terme correspond à la maximisation de l'erreur de discrimination. Ces deux termes sont pondérés respectivement par l'amplitude de l'erreur de prédiction et de discrimination. Afin de conserver une influence sur la pondération prédiction/discrimination, nous avons finalement retenu :

$$\Delta\omega(q) = \mu \left((1-\alpha) \frac{\partial Q^D}{\partial \omega} - \alpha \frac{\partial Q^M}{\partial \omega} \right) \tag{15}$$

La détermination exacte des règles de modification des poids du réseau ne pose pas de problème particulier.

Validation expérimentale. Nous avons testé l'évolution du MER lors du réglage de l'extracteur sur la base B1 composée des quatre phonèmes voisés */aa/*, */ae/*, */ey/* et */ow/* pour deux valeurs de α : $\alpha = 1$ (pas de contrainte discriminante, l'extracteur est équivalent au modèle NPC-2) et $\alpha = 0.5$ avec laquelle nous obtenons un extracteur DFE-NPC et une contribution égale entre l'erreur de prédiction et l'erreur de discrimination. La figure 3 illustre cette évolution du MER et nous pouvons observer qu'il se stabilise à une valeur plus élevée lorsque la contrainte discriminante est appliquée ($\alpha = 0.5$). Les paramètres DFE-NPC se révèlent donc plus discriminants que les paramètres NPC-2.

Nous avons souhaité observer si ces résultats se retrouvaient dans la distribution des codes générés. Pour cela, nous avons effectué une estimation des variances inter-classes à chaque itération d'apprentissage, en supposant que les distributions sont gaussiennes sphériques. La figure 4 reporte les résultats obtenus et nous pouvons voir qu'effectivement, dans le cas du codeur DFE-NPC, la variance inter-classes converge vers une valeur plus élevée que dans le cas du codeur NPC-2. Nous montrons ainsi, expérimentalement, que la maximisation du MER conduit à maximiser la variance inter-classes des données et donc à une meilleure représentation des codes dans leur espace de représentation en vue de leur classification. Nous avons réitéré les expérimentations précédentes pour la classification des pho-

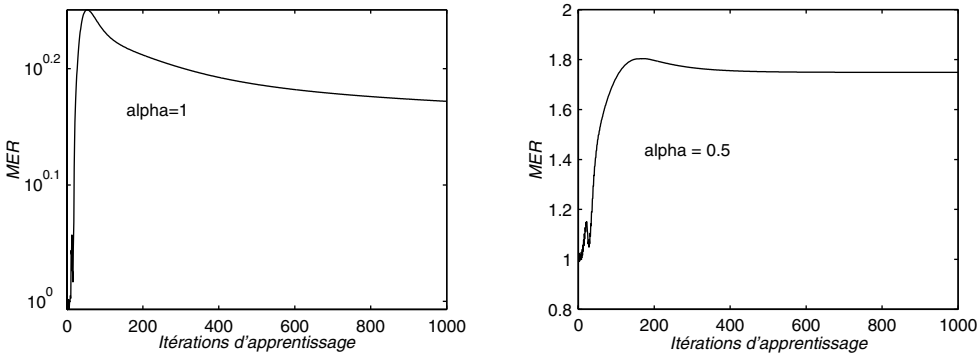

Figure 3. Évolution du MER pendant la phase de réglage. En haut : $\alpha = 1$ (modèle NPC-2) et en bas $\alpha = 0.5$ (modèle DFE-NPC)

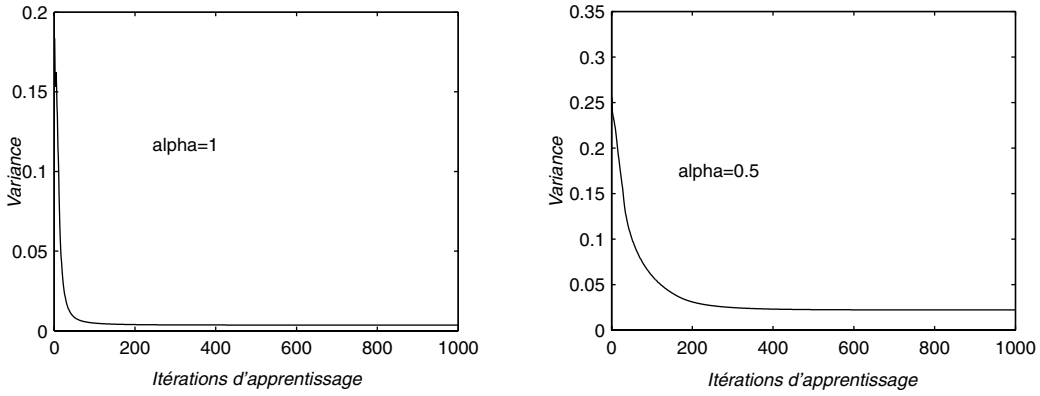

Figure 4. Évolution de la variance inter-classes pendant la phase de réglage. En haut : $\alpha = 1$ (paramètres NPC-2). En bas, $\alpha = 0.5$ (paramètres DFE-NPC).

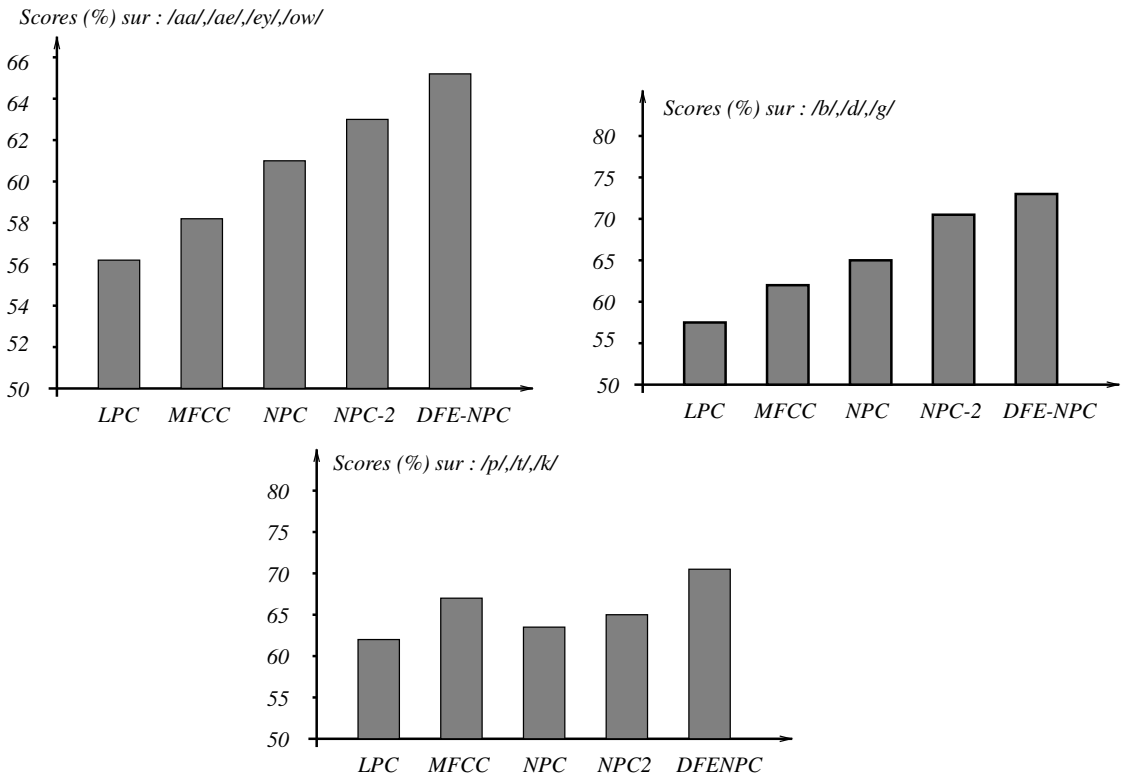

Figure 5. Taux de reconnaissance comparatifs obtenus en généralisation par un classifieur PMC pour les phonèmes /aa/, /ae/, /ey/, /ow/, /b/, /d/, /g/ et /p/, /t/, /k/ et plusieurs jeux de paramètres.

nèmes des bases B1, B2 et B3 et avons obtenu les scores présentés figures 5. Les résultats obtenus sur les bases B1 et B2 (phonèmes voisés) sont meilleurs que ceux obtenus à l'aide des extracteurs précédents (NPC et NPC-2). L'amélioration apportée par l'extracteur DFE-NPC sur la base B3 (phonèmes /*p*/, /*t*/, /*k*/) est remarquable en ce qu'elle est spécifique à cet extracteur. Rappelons en effet que les paramètres NPC-2 ne donnent pas de bonnes performances sur cette même base (voir section 4.1 de cet article).

La valeur optimale de la pondération α n'est pas facile à déterminer car elle varie selon la classe du phonème étudié. À titre d'exemple, le tableau 7 présente les valeurs optimales de α obtenues pour les trois catégories de phonèmes considérées. Dans le contexte de ces simulations, nous avons donc choisi un facteur de pondération adaptatif. La loi d'adaptation en question obéit à la règle selon laquelle la minimisation de l'erreur de prédiction (valeur de α proche de 1) prévaut sur la maximisation de l'erreur de discrimination (α proche de 0) en début de simulation. Ensuite, elle est dirigée par l'évolution du MER. Lorsque ce dernier augmente, signe d'une amélioration des caractéristiques de l'extracteur, α est diminué afin de privilégier progressivement l'aspect discriminant. Lorsqu'au contraire le MER décroît, la priorité est à nouveau donnée à la modélisation en augmentant la valeur de α. On a la règle d'adaptation suivante :

$$\begin{cases} \alpha_0 = 0.7 \\ \alpha_{q,q>0} = \begin{cases} 0.9\alpha_{q-1} \text{ si } \Gamma_q - \Gamma_{q-1} > 0 \\ 1.1\alpha_{q-1} \text{ sinon} \end{cases} \end{cases} \quad (16)$$

La figure 6 donne un exemple d'évolution du paramètre α adaptatif durant la phase de réglage de l'extracteur sur la base B1. Nous voyons que, partant d'une valeur initiale égale à 0.7, α

Tableau 7. Valeurs optimales du paramètre α obtenues pour les trois catégories de phonèmes étudiées.

phonèmes	/voyelles/	/*b*/, /*d*/, /*g*/	/*p*/, /*t*/, /*k*/
α optimal	0.5	0.7	0.5

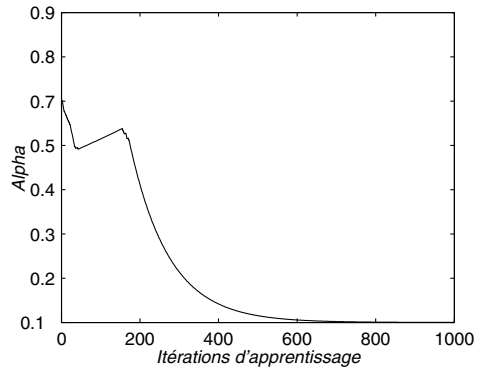

Figure 6. Évolution du paramètre α adaptatif durant la phase de réglage DFE-NPC (phonèmes /aa/, /ae/, /ey/ et /ow/).

converge ensuite vers 0, c'est-à-dire une valeur pour laquelle seule l'erreur de discrimination continue d'être minimisée. Le tableau 8 présente des résultats de classification obtenus dans les deux cas de figure d'une valeur constante de α d'une part et d'une valeur estimée par notre algorithme d'autre part.

Tableau 8. Taux de classification obtenus selon la détermination de la contrainte α.

phonèmes	α constant	α adaptatif
/aa/, /ae/, /ey/, /ow/	63.5	65.25
/b/, /d/, /g/	67.66	70.1
/p/, /t/, /k/	66.33	69.67

5. Analyse des modèles

Dans cette section, nous proposons une discussion des différents algorithmes proposés dans cet article. Nous présentons des éléments de validation expérimentale et théorique. Nous revenons en particulier sur le problème du sur-apprentissage vu au paragraphe 3, puis proposons une discussion plus générale sur la problématique de l'extraction de caractéristiques. Enfin, nous concluons en présentant une expérimentation des paramètres NPC sur une application de reconnaissance de mots isolés. L'intégration des paramètres NPC dans un système de RAP de type ANN-HMM est discutée.

5.1. Le principe de superposition

Ainsi que nous l'avons défini au paragraphe 3 de cet article, avec les modèles NPC, NPC-2 et DFE-NPC, seuls les poids de la couche de sortie sont les paramètres libres de l'algorithme d'apprentissage lors de l'extraction de caractéristiques. Cela signifie que nous considérons d'emblée que l'information propre aux signaux modélisés est essentiellement portée par ces poids, et non par les poids de la couche cachée. Des résultats obtenus par Kolmogorov [42] permettent de donner une justification théorique de ce fait.

En effet, en 1957, Kolmogorov prouve avec son théorème de superposition (réfutation du 13ème problème de Hilbert) que toute fonction continue f de $\Re^\lambda$ dans $\Re$ peut être représentée par une somme de fonctions continues de $\Re$ dans $\Re$:

$$f(x_1,\ldots,x_n) = \sum_{q=1}^{2\lambda+1} \phi_q \left(\sum_{p=1}^{\lambda} \psi_{pq}(x_p) \right) \qquad (17)$$

Ce résultat signifie que l'on peut représenter toute fonction continue de plusieurs variables par une superposition d'un petit nombre de fonctions à une variable. Si l'on raisonne en terme de réseaux de neurones de type PMC, comme l'a proposé Hecht-Nielsen[28] en 1987, cela signifie que toute fonction continue de plusieurs variables peut être calculée à l'aide d'un réseau à deux couches cachées. D'autres auteurs ont pu montrer qu'un tel résultat pouvait être obtenu par un réseau PMC à une seule couche cachée. Poggio et Girosi [26] ont cependant souligné les limites de cette approches. Des résultats obtenus par Vitushkin [67] ont en effet montré que les fonctions du théorème de Kolmogorov (les fonctions ψ_{pq}) pouvaient être très fortement irrégulières bien que continues. Or les fonctions mises en oeuvre dans les réseaux PMC (les fonctions sigmoïdes) sont régulières, paramétriques, et ce point est important pour l'obtention des propriétés de généralisation et de robustesse que l'on en attend. Mais ce qui nous intéresse plus particulièrement dans les travaux de Hecht-Nielsen, c'est que dans sa démonstration, les couches cachées sont estimées indépendamment de la fonction f approximée, de sorte que cette partie du réseau de neurones est apprise une fois pour toute pour une valeur donnée de λ. Kurkova [44], Sprecher et Katsuura [37] et d'autres encore, ont ainsi montré qu'il existait des couches cachées *universelles*, indépendantes même de λ. Le modèle d'extraction des paramètres NPC est construit selon ce principe : seuls les poids de la couche de sortie dépendent de la fonction f approximée. Des auteurs [53] ont déjà tiré parti du théorème de Kolmogorov pour réaliser des filtres non-linéaires. Par ailleurs, d'autres arguments que nous présentons ci-dessous permettent d'obtenir une conclusion identique.

5.2. Des paramètres NPC aux paramètres DFE-NPC

Selon le formalisme présenté au paragraphe précédent, un prédicteur NPC modélise un ensemble de fonctions : autant de fonctions de $\Re^\lambda$ dans $\Re$ qu'il y a de cellules de sorties, donc de trames. Cette formalisation repose ainsi sur le principe qu'une trame de signal quelconque l est produite par un processus du type :

$$y_k = f_l(\mathbf{y}_k) + \varepsilon_k. \qquad (18)$$

Dans le cas idéal, ε_k désigne un bruit blanc gaussien. Mais nous avons vu dans l'introduction de cet article qu'une telle superposition n'est en général pas réaliste avec le signal de parole. Lorsque la trame est non voisée, il existe f_l linéaire satisfaisant (18). En revanche, lorsque la trame est voisée, nous avons vu que f_l non linéaire est une meilleure approximation. Nous ne pouvons considérer pour autant que ε_k est blanche. Il reste en effet une erreur résiduelle quasi-périodique, souvent idéalisée par un peigne de Dirac à la fréquence du pitch. Ce signal résiduel correspond à la source glottique dans le modèle source-filtre du conduit vocal. Sa présence est due au fait que (18) est une modélisation prédictive *court-terme* du signal de parole. Dans la démonstration qui suit, nous négligeons cette composante en assimilant ε_k à un bruit blanc. Nous considérons ainsi que toute l'information concernant le processus générateur est

présente dans f_l, ce qui n'est en fait qu'une approximation de la réalité. On y trouve en particulier des informations concernant le locuteur [56], même si elles sont, la plupart du temps il est vrai, ignorées par les analyseurs (MFCC notamment).

Signaux non bruités. Nous nous plaçons dans le cadre théorique de l'approximation de fonctions et considérons dans un premier temps des données d'apprentissage non bruitées. La modélisation du processus générateur d'un signal $y_k = f(y_{k-1}, \dots y_{k-\lambda}) + \varepsilon_k$ est vue comme un problème d'approximation d'une fonction f de $[-1,+1]^\lambda$ dans $[-1,+1]$ qui à $\mathbf{y_k}$ associe $y_k = f(\mathbf{y}_k)$ et pour lequel on suppose dans un premier temps $\varepsilon_k = 0$.

Soient deux phonèmes de classes différentes i et j, $j \neq i$ et un ensemble d'apprentissage $\mathcal{D} = \mathcal{D}_i \cup \mathcal{D}_j$ constitué de trames de classe i et j. On a pour chacun des couples d'apprentissage $(y_k, \mathbf{y}_k)$:

$$\begin{cases} (\mathbf{y}_k, y_k) \in \mathcal{D}_i \Longrightarrow y_k = f_i(\mathbf{y}_k) \\ (\mathbf{y}_k, y_k) \in \mathcal{D}_j \Longrightarrow y_k = f_j(\mathbf{y}_k). \end{cases} \tag{19}$$

Construire un analyseur NPC, $F_{\Omega,\mathbf{a}_i,\mathbf{a}_j}$, minimisant le coût quadratique calculé sur $\mathcal{D}$ conduit à obtenir deux opérateurs $F_{\Omega,\mathbf{a}_i}$ et $F_{\Omega,\mathbf{a}_j}$, vérifiant sur les données d'apprentissage [10] :

$$\begin{cases} (\mathbf{y}_k, y_k) \in \mathcal{D}_i \Longrightarrow y_k = F_{\Omega,\mathbf{a}_i}(\mathbf{y}_k) \\ (\mathbf{y}_k, y_k) \in \mathcal{D}_j \Longrightarrow y_k = F_{\Omega,\mathbf{a}_j}(\mathbf{y}_k). \end{cases} \tag{20}$$

Sur les couples de $\mathcal{D}_i$ et de $\mathcal{D}_j$ uniquement, on obtient l'égalité des prédicteurs avec les processus :

$$\begin{cases} H_{\mathbf{a}_i} \circ G_\Omega = f_i \\ H_{\mathbf{a}_j} \circ G_\Omega = f_j \end{cases} \tag{21}$$

où $H_{\mathbf{a}_i}$ et $H_{\mathbf{a}_j}$ représentent l'opérateur réalisé par les deux couches de sortie du réseau et G_Ω l'opérateur réalisé par la couche cachée. L'opérateur global F réalisé par le réseau est donné par la composition des fonctions : $F_{\Omega,\mathbf{a}_i} = H_{\mathbf{a}_i} \circ G_\Omega$ et $F_{\Omega,\mathbf{a}_j} = H_{\mathbf{a}_j} \circ G_\Omega$. Nous cherchons à montrer dans ce qui suit que $i \neq j \Longrightarrow H_{\mathbf{a}_i} \neq H_{\mathbf{a}_j} \Longrightarrow \mathbf{a}_i \neq \mathbf{a_j}$.

Soient $\mathcal{D}_i^{\mathbf{y}}$ et $\mathcal{D}_j^{\mathbf{y}}$ les deux ensembles de définition sur lesquels sont appris $F_{\Omega,\mathbf{a}_i}$ et $F_{\Omega,\mathbf{a}_j}$. S'il existe des échantillons $\mathbf{y}$ de $\mathcal{D}_i^{\mathbf{y}} \cap \mathcal{D}_j^{\mathbf{y}}$ tels que $f_i(\mathbf{y}) \neq f_j(\mathbf{y})$, alors il n'existe pas $F_{\Omega,\mathbf{a}_i}$ et $F_{\Omega,\mathbf{a}_j}$ telles que $H_{\mathbf{a}_i} = H_{\mathbf{a}_j}$ et qui soient des fonctions. D'où, dans ce cas précis, $\mathbf{a}_i \neq \mathbf{a}_j$. Nous caractérisons cette situation par $\mathcal{D}_i \cap \mathcal{D}_j = \emptyset$ et $\mathcal{D}_i^{\mathbf{y}} \cap \mathcal{D}_j^{\mathbf{y}} \neq \emptyset$. Elle est malheureusement très restrictive et il est plus raisonnable de considérer que si les ensembles $\mathcal{D}_i$ et $\mathcal{D}_j$ sont disjoints, les ensembles de définition le sont aussi.

Pour $\mathcal{D}_i^{\mathbf{y}} \cap \mathcal{D}_j^{\mathbf{y}} = \emptyset$ (deuxième situation), nous ne pouvons prouver que $f_i \neq f_j \Longrightarrow \mathbf{a}_i \neq \mathbf{a}_j$. Il est en effet possible de construire une fonction f tel que :

$\forall \mathbf{y} \in \mathcal{D} = \mathcal{D}_i^{\mathbf{y}} \cup \mathcal{D}_j^{\mathbf{y}}, \mathbf{y} \in \mathcal{D}_i^{\mathbf{y}} \Longrightarrow f(\mathbf{y}) = f_i(\mathbf{y})$

et $\mathbf{y} \in \mathcal{D}_j^{\mathbf{y}} \Longrightarrow f(\mathbf{y}) = f_j(\mathbf{y})$. De part la propriété d'approximation universelle des réseaux PMC à une couche cachée, il existe un réseau de neurone F tel que $F = H \circ G$ et F est une approximation de f sur $\mathcal{D}$, d'où $\mathbf{a}_i = \mathbf{a}_j$ devient une solution possible.

Cette difficulté peux être contournée en modifiant judicieusement les ensembles d'apprentissage respectifs $\mathcal{D}_i$ et $\mathcal{D}_j$. En effet, si nous pouvons trouver deux nouveaux ensembles d'apprentissage vérifiant $\mathcal{D}_i' \cap \mathcal{D}_j' = \emptyset$ et $\mathcal{D}_i^{\mathbf{y}'} \cap \mathcal{D}_j^{\mathbf{y}'} \neq \emptyset$, alors nous sommes ramenés à la première situation pour laquelle nous avons montré que $i \neq j \Longrightarrow \mathbf{a}_i \neq \mathbf{a}_j$. Pour cela, étendons les deux ensembles de définition à leur réunion : $\mathcal{D}_i^{\mathbf{y}'} = \mathcal{D}_j^{\mathbf{y}'} = \mathcal{D}_i^{\mathbf{y}} \cup \mathcal{D}_j^{\mathbf{y}} = \mathcal{D}^{\mathbf{y}}$. La condition $\mathcal{D}_j^{\mathbf{y}'} \cap \mathcal{D}_j^{\mathbf{y}'} \neq \emptyset$ est alors vérifiée par construction puisque $\mathcal{D}_i^{\mathbf{y}'} \cap \mathcal{D}_j^{\mathbf{y}'} = \mathcal{D}^{\mathbf{y}}$. Pour s'assurer que la condition $\mathcal{D}_i' \cap \mathcal{D}_j' = \emptyset$ est également vérifiée, il nous faut compléter les couples d'apprentissage de telle sorte qu'à partir de tout élément de $\mathbf{y}$ de $\mathcal{D}_i^{\mathbf{y}}$, on construit un couple $(\mathbf{y}, y')$ de $\mathcal{D}_j'$ tel que $y' \neq f_i(\mathbf{y})$. Réciproquement, à partir des éléments $\mathbf{y}$ de $\mathcal{D}_j^{\mathbf{y}}$, on construit les couples $(\mathbf{y}, y') \in \mathcal{D}_i'$ tels que $y' \neq f_j(\mathbf{y})$.

On montre simplement qu'une telle construction est obtenue en maximisant le MER, c'est-à-dire en *pénalisant* explicitement les données d'apprentissage de $F_{\Omega,\mathbf{a}_i}$ sur $F_{\Omega,\mathbf{a}_j}$ (maximisation de l'erreur de prédiction) et inversement en *pénalisant* celles de $F_{\Omega,\mathbf{a}_j}$ sur $F_{\Omega,\mathbf{a}_i}$. Ainsi, sous condition de convergence de l'algorithme d'apprentissage, et dans le cadre des hypothèses initiales, l'algorithme de réglage DFE-NPC permet de garantir la différenciation des poids de la deuxième couche.

Rien ne garantit que le résultat précédent sera vérifié lors de la phase d'extraction puisque l'on ne peut qu'utiliser l'algorithme d'apprentissage initial NPC pour lequel seule l'erreur de prédiction est minimisée. Néanmoins, l'on peut montrer que la règle de décision consistant à choisir le modèle minimisant l'erreur de prédiction est une approximation de la règle de décision maximisant le MER. Soit $\hat{c}(l)$ l'estimation de la classe d'appartenance d'une trame l quelconque obtenue par minimisation de l'inverse du MER sur un extracteur NPC-2 :

$$\hat{c}(l) = \arg\min_m \frac{Q_l(\mathbf{a}_m)}{\sum_{i,i\neq m}^{N_c} Q_l(\mathbf{a}_i)} \tag{22}$$

où $Q_l(\mathbf{a}_m)$ désigne l'erreur de prédiction estimée sur les poids $\mathbf{a}_m$ de la cellule de sortie m de l'extracteur lorsque l'on présente les échantillons de la trame l. Par une approximation au dénominateur, on obtient donc la nouvelle règle de décision minimisant l'erreur de prédiction :

$$\hat{c}(l) \simeq \arg\min_m \frac{Q_l(\mathbf{a}_m)}{\sum_i^{N_c} Q_l(\mathbf{a}_i)} = \arg\min_m Q_l(\mathbf{a}_m). \tag{23}$$

Afin d'achever cette démonstration, il conviendrait de montrer que : $i = j \Longrightarrow \mathbf{a}_i = \mathbf{a}_j$. Cette condition n'est pas vérifiée pour deux raisons au moins. La première est la grande variabilité du signal de parole : même lorsque l'on suppose le bruit négligeable, deux trames de même classe phonétique peuvent obéir à des modèles différents $f_i \neq f_j$ correspondant par exemple à deux locuteurs différents.

La deuxième raison tient aux réseaux de neurones qui pour l'approximation d'une fonction donnée f n'offrent pas une solution

unique des poids. Ce problème peut être éliminé lors de l'extraction de caractéristiques où seule la couche de sortie est adaptée. Si la fonction de transition est linéaire, alors la méthode d'extraction sera une méthode d'estimation linéaire garantissant l'unicité de la solution. Si la fonction de transition est non linéaire (fonction sigmoïde), on pourra avantageusement démarrer la phase d'apprentissage en utilisant des poids initiaux calculés après approximation linéaire du réseau comme proposé dans [14].

Bruit gaussien. Les résultats obtenus précédemment peuvent être généralisés au cas d'échantillons $(\mathbf{y}_k, y_k)$ bruités additivement par un bruit blanc gaussien centré. Il existe en effet un résultat important concernant l'interprétation des sorties d'un réseau entraîné par minimisation d'une fonction d'erreur quadratique : la sortie $F(\mathbf{y})$ approxime l'espérance conditionnelle de la donnée cible y : $F(\mathbf{y}) =< y|\mathbf{y} >$ [9] [10].
En supposant que les données cibles sont générées à partir d'une fonction déterministe $f(\mathbf{y})$ avec un bruit additif gaussien centré ε, alors les données cibles sont données par $y = f(\mathbf{y}) + \varepsilon$. La sortie du réseau, lorsque le minimum de la fonction de coût est atteint, devient :
$F(\mathbf{y}) =< y|\mathbf{y} >=< f(\mathbf{y}) + \varepsilon|\mathbf{y} >= f(\mathbf{y}) + < \varepsilon >= f(\mathbf{y})$
puisque ε est centré. Ce résultat est généralisable au cas des données d'entrées $\mathbf{y}_k$ également perturbées par un bruit gaussien centré. Webb [69] a en effet montré que dans ce cas, la solution optimale (minimum de la fonction de coût) est à nouveau donnée par l'espérance conditionnelle de la donnée cible $< y_k|\mathbf{y}_k >$.
Enfin, il y a des résultats expérimentaux qui mettent clairement en évidence les propriétés discriminantes des poids de sortie du codeur NPC et dont nous discutons au paragraphe 5.3.

Équivalence du MER avec le critère MMI. De même qu'Itakura dans son étude des coefficients LPC [32] montre que la minimisation du logarithme du rapport des erreurs de prédiction est équivalent, du point de vue statistique, à la maximisation du logarithme du rapport de vraisemblance, Chetouani *et al.* dans [15] ont récemment montré que minimiser le LMER équivaut à maximiser le critère MMI (Maximisation de l'information mutuelle).
La minimisation de $Q^{DFE-NPC}$ conduit ainsi à l'élaboration de l'extracteur permettant la meilleure classification lorsqu'on l'utilise comme classifieur, c'est-à-dire lorsque la règle de décision pour une trame l inconnue consiste à choisir la classe minimisant l'erreur de prédiction : $c(l) = \arg\min_i Q^{NPC}(\Omega^\star, \mathbf{a}_i^\star)$.
La démonstration de l'équivalence du MER avec le critère MMI a été obtenue sous l'hypothèse de modélisation du conduit vocal par un processus statistique du type $y_k = f(\mathbf{y}_k) + \varepsilon_k$ où ε_k est un bruit blanc gaussien centré, les poids solution correspondant à un minimum global de la fonction de coût.

5.3. Validations expérimentales

Pour commencer, le modèle NPC apporte une première validation expérimentale de ce que les informations propres aux fonctions implémentées sont situées sur la couche de sortie et non sur la couche cachée. En témoignent en effet les scores obtenus à partir des paramètres NPC en reconnaissance de phonème et présentés au paragraphe 3.4 dans la table 4 : ils restent comparables avec les scores obtenus par les paramètres LPC et MFCC.

Mesure du MER durant le réglage NPC-2. Il était intéressant de mesurer les valeurs prises par le MER lors de la phase de réglage de l'extracteur NPC-2, c'est-à-dire sans minimiser explicitement $Q^{DFE-NPC}$, afin d'observer l'évolution des propriétés discriminantes de l'extracteur.
Nous avons réalisé des expérimentations sur les trois bases de phonèmes B1, B2 et B3 dont les résultats sont reportés sur les figures 7, 8 et 9. Nous avons calculé le rapport MER pour chaque itération d'apprentissage lors du réglage de l'extracteur. On peut voir sur ces figures que les valeurs initiales sont proches de 1, ce qui indique que l'état initial de l'extracteur NPC-2 (les poids initiaux sont choisis aléatoirement) corres-

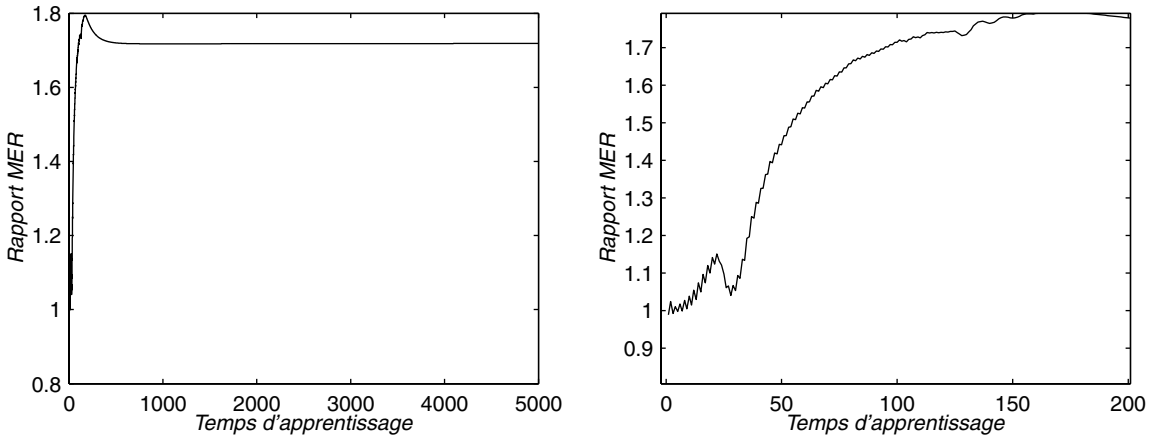

Figure 7. Évolution du MER lors de la phase de réglage NPC-2 sur la base B1. À gauche, 5000 itérations d'apprentissage, à droite le détail des 200 premières itérations.

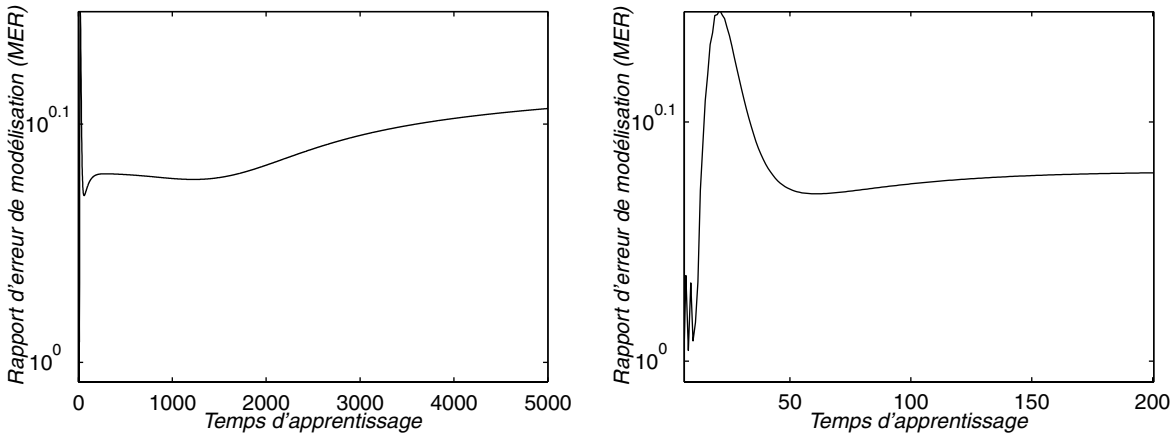

Figure 8. Évolution du MER lors de la phase de réglage NPC-2 sur la base B2. À gauche, 5000 itérations d'apprentissage, à droite le détail des 200 premières itérations.

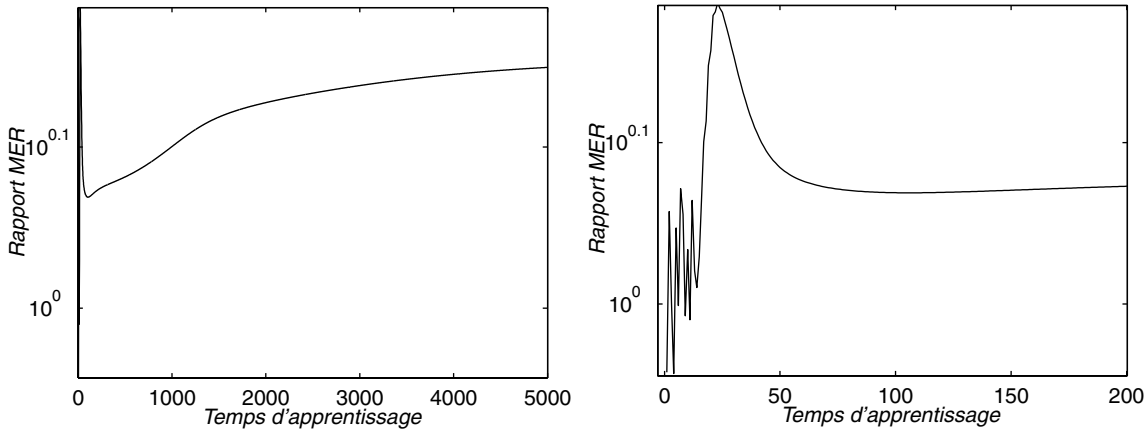

Figure 9. Évolution du MER lors de la phase de réglage NPC-2 sur la base B3. À gauche, 5000 itérations d'apprentissage, à droite le détail des 200 premières itérations.

pond à celui d'un extracteur dépourvu de toute capacité discriminante. Les courbes obtenues montrent ainsi l'acquisition progressive de capacités discriminantes par l'extracteur NPC-2. Elles présentent cependant un maximum, placé généralement avant les 200 premières itérations, et semblant montrer la possibilité de sur-apprentissage, ainsi que nous l'avons déjà mis en évidence lors de la phase d'extraction de caractéristiques NPC (*cf.* paragraphe 3.4). Les paramètres NPC-2 présentent les mêmes caractéristiques comme le montre la table 9 où l'on observe que les scores obtenus en reconnaissance présentent un maximum avant les 30 premières itérations de la phase d'extraction. Contrairement à ce qu'indique l'évolution du MER en cours de réglage, aucun des algorithmes de réglage (NPC, NPC-2 et DFE-NPC) ne présente de sur-apprentissage. En général, plus le nombre d'itérations est important en phase de réglage, meilleurs sont les scores en reconnaissance. La principale explication tient au fait que le réglage vise la détermination des poids de la couche cachée, dont on sait (*cf.* § 5.1 et § 5.2) qu'ils ne codent pas les informations discriminantes exploitées en reconnaissance. Le maximum présent sur les courbes de réglage, figures 7, 8 et 9, tient au fait que le MER représente les capacités discriminantes de l'extracteur lorsqu'on l'utilise en classifieur (voir la fin du paragraphe 5.2 à ce sujet).

Tableau 9. Performances des paramètres NPC-2 obtenues à différentes étapes du processus d'extraction des caractéristiques (phonèmes /aa/,/ae/,/ey/,/ow/, ligne du milieu et /b/, /d/, /g/, dernière ligne).

5	10	20	30	100	150	200	500
59.2	62	**62.6**	61.8	59.8	59.5	58.5	58
70	64	56	55	49.6	48	48.1	48

L'explication de ce phénomène tient au faible nombre d'échantillons d'apprentissage disponibles pour réaliser une extraction. Une trame de signal dont il faut extraire les coefficients NPC présente $D - \lambda$ échantillons d'apprentissage. Il est difficile d'augmenter la taille des trames (D) du fait de l'hypothèse de

stationnarité qui doit être respectée. λ ne peut être diminué non plus puisqu'il s'agit de l'horizon de prédiction. Ce dilemme fait que la solution apportée au problème de sur-apprentissage n'est pour le moment qu'empirique : toute extraction de caractéristiques se voit imposer le nombre de 20 itérations d'apprentissage.

5.4. Application des paramètres NPC à la reconnaissance de mots isolés

Le classifieur utilisé dans cet article pour établir les validations expérimentales, en l'occurrence un réseau PMC, était relativement simple : le problème posé dans cet article n'est pas celui de la classification mais bien celui de l'extraction de caractéristiques. Le classifieur PMC nous a donc permis d'établir des comparaisons entre différents modèles, dans des conditions expérimentales rigoureusement identiques. Par ailleurs, nous avons estimé des scores en reconnaissance de phonèmes (et donc effectué une décision) là où les systèmes actuels de RAP n'attendent qu'une estimation des probabilités *a posteriori* ou bien encore des vraisemblances dites « locales ». Les scores présentés sont donc à interpréter, non pas dans l'absolu, mais relativement à chaque jeu de paramètres, leur objet étant d'établir des comparaisons quantitatives. Les conditions expérimentales ont été rendues volontairement difficiles : la décision étant effectuée au niveau de la trame phonétique plutôt qu'au niveau du phonème entier. Les phénomènes de co-articulation n'ont donc pas été pris en compte. Enfin, la reconnaissance était indépendante du contexte.

Il nous a semblé important de montrer qu'il est possible d'intégrer les paramètres NPC dans un système de reconnaissance plus complet, en l'occurrence un système de reconnaissance de mots isolés à base de chaînes de Markov cachées (HMM). Nous avons donc réalisé une application de reconnaissance de mots isolés à vocabulaire limité, multilocuteurs, que nous avons testée sur dix mots des bases TIMIT et NTIMIT.

Le système proposé. Les caractéristiques de l'expérimentation se résument ainsi :

- modélisation des mots par un système de type HMM : un modèle par mot, comportant autant d'états que de phonèmes constitutifs ;
- calcul des paramètres NPC : un unique codeur NPC réglé pour l'ensemble des modèles ;
- 16 coefficients ainsi que 16 coefficients Δ et 16 $\Delta\Delta$ constituent un vecteur acoustique de 48 paramètres. 3 vecteurs consécutifs sont présentés en entrée soit un total de 144 paramètres ;
- estimation des densités de probabilité par un réseau de type perceptron multicouches (PMC) à une couche cachée (10 cellules) et associé à chacun des modèles HMM. Utilisation de fonctions *softmax* en sortie, les réseaux comportant autant de sorties que d'états HMM, afin d'établir les vraisemblances locales ;
- algorithme de Viterbi pour l'apprentissage de chacun des modèles HMM et la reconnaissance (segmentation phonétique).

La figure 10 représente la structure globale du système.

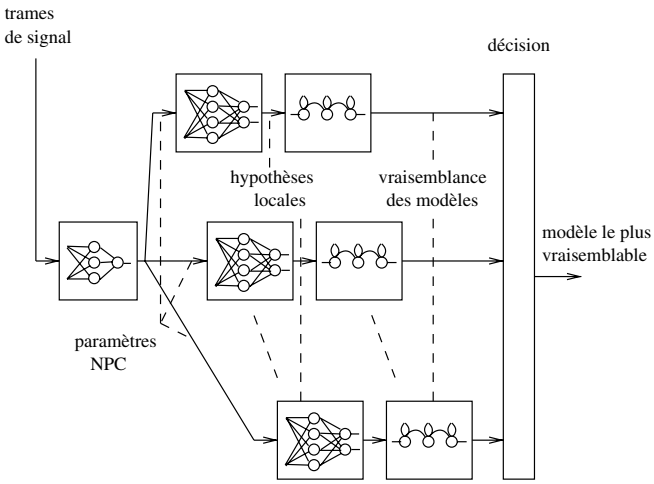

Figure 10. Structure globale du système de reconnaissance ANN-HMM utilisé.

L'on voit que l'association d'un réseau PMC pour l'estimation des vraisemblances locales avec un modèle HMM pour modéliser les séquences de paramètres nous donne un modèle global de type ANN-HMM tels que ceux décrits dans la littérature [12]. Les modèles ANN-HMM sont fondés sur le principe d'estimer les densités de probabilités nécessaires au fonctionnement d'un système HMM à l'aide d'un perceptron multicouches. Un certain nombre d'auteurs [10] ont en effet montré que les sorties d'un perceptron utilisé en classifieur peuvent être interprétées, sous certaines conditions, comme des estimateurs des probabilités a posteriori d'appartenance du vecteur d'entrée aux différentes classes ou états HMM. L'application de la loi de Bayes permet ensuite d'obtenir une estimation des vraisemblances locales. La figure 11 représente le détail d'un modèle de mot ANN-HMM obéissant à ce principe.

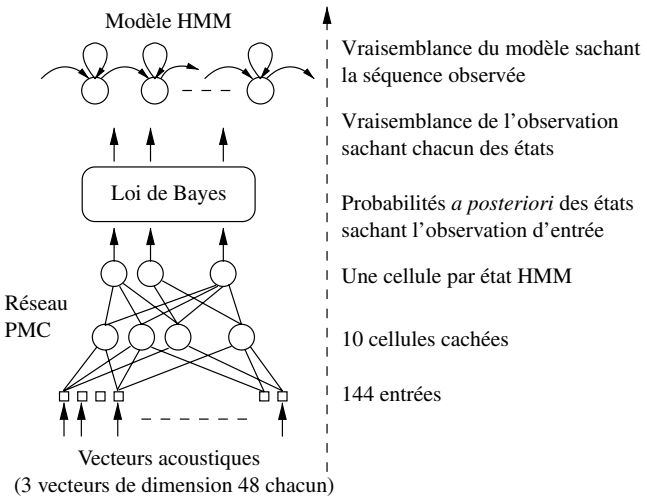

Figure 11. Détail d'un modèle de mot ANN-HMM.

Résultats expérimentaux. Les mots appris sont les 10 mots de la phrase *she had your dark suit in greasy wash water year*, prononcée par l'ensemble des locuteurs de la base TIMIT.
Nous avons effectué deux séries de tests, l'un portant sur les signaux non bruités de la base TIMIT (dialectes *DR1* et *DR2*), l'autre portant sur les signaux téléphoniques de la base NTIMIT (dialectes *DR1* et *DR2* également). Ces bases sont constituées de 74 locuteurs pour l'apprentissage et de 19 locuteurs pour le test. La ventilation retenue des locuteurs entre les deux bases reprend exactement celle proposée dans la distribution TIMIT entre les ensembles *train* et *test*. L'extracteur NPC a été réglé sur l'ensemble des signaux de la base d'apprentissage, à raison de 1000 itérations d'apprentissage.
Nous avons reporté sur la figure 12 les scores obtenus pour une dizaine d'itérations d'apprentissage du système entier. Précisons qu'une itération représente ici un cycle *Estimation-Maximisation* (algorithme EM) incorporant 1500 itérations d'apprentissage pour chacun des réseaux PMC et nécessitant environ 6 heures de calcul sur une machine de type Pentium cadencé à 2.4GHz.

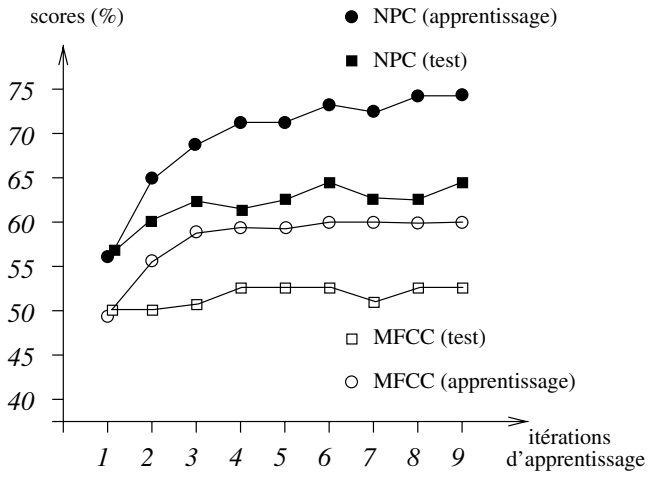

Figure 12. Scores obtenus en apprentissage et en test en fonction du nombre d'itérations d'apprentissage sur la base NTIMIT.

Le tableau 10 donne les meilleurs scores obtenus en test sur chacune des deux bases Timit et Ntimit. Ces résultats montrent que les paramètres MFCC permettent d'obtenir de meilleures performances sur la première des deux bases, tandis que les paramètres NPC donnent au contraire de meilleurs performances sur la base téléphonique. Ces mêmes expérimentations ont également montré que les paramètres NPC permettent d'obtenir en apprentissage des scores systématiquement plus élevés que les paramètres MFCC. (de près de 90 % pour 10 mots appris et prononcés par les 74 locuteurs à 100 % pour 10 mots prononcés par un unique locuteur). Ils conduiraient, de ce fait à un système global plus facilement sujet au sur-apprentissage. Il serait intéressant, à ce point du travail, d'étudier l'impacte du

Tableau 10. Scores obtenus en reconnaissance de mots isolés sur 10 mots des bases TIMIT et NTIMIT.

	TIMIT	NTIMIT
apprentissage MFCC	80 %	60 %
test MFCC	**68** %	52 %
apprentissage NPC-1	82 %	74 %
test NPC-1	63 %	**63** %

nombre d'itérations d'apprentissage lors de l'étape d'extraction de caractéristiques NPC (10 itérations dans le cas présent) sur les taux de reconnaissance obtenus en généralisation. Retrouverait-on ici le problème de sur-apprentissage reporté au paragraphe § 3.4 de cet article ?

Insertion des paramètres NPC dans un système de l'état de l'art. Le classifieur que nous avons utilisé est donc un système de l'état de l'art dont la vocation ici était de montrer un exemple d'intégration des paramètres NPC au sein d'un système de RAP. Ces premiers résultats peuvent être bien entendu améliorés car nous sommes loin des conditions optimales d'utilisation de ce type de système. En particulier :
- l'emploi d'un perceptron par modèle HMM de mot, même s'il s'avère plus simple à mettre en œuvre, conduit à réaliser un système à vocabulaire obligatoirement limité ;
- l'emploi, au contraire, d'un unique perceptron partagé par l'ensemble des modèles (un réseau qui serait alors un estimateur des émissions de probabilité de l'ensemble des phonèmes de la langue) permettrait de disposer d'un corpus d'apprentissage plus important et d'améliorer en conséquence les performance en test [41]. C'est d'ailleurs ce type de système qui est habituellement implémenté dans la littérature.
- les modèles ANN-HMM de l'état de l'art donnent leurs meilleurs performances lorsque que les vecteurs acoustiques sont constitués de 9 vecteurs caractéristiques, soient plus de 200 entrées, et comportent entre 500 et 4000 unités cachées ;
- enfin, nous avons initialisé les modèles HMM à partir d'une segmentation initiale sans information *a priori* concernant la longueur des phonèmes constitutifs. Une initialisation basée sur la segmentation phonétique proposée dans le corpus TIMIT augmenterait les chances de convergence des modèles vers des solutions plus optimales.
D'autres arguments existent bien entendu pour étayer notre propos, que l'on peut trouver facilement dans la littérature traitant des modèles ANN-HMM. Il en existe un qui concerne plus spécifiquement l'utilisation des paramètres NPC. En effet, un système plus conséquent, et que nous étudions actuellement, vise à mettre en œuvre les paramètres DFE-NPC dont l'étude présentée dans cet article a montré les propriétés discriminantes intéressantes. Le réglage DFE-NPC exige la connaissance de la classe d'appartenance phonétique des trames de parole. Cette connaissance était supposée connue jusqu'à présent lors de la phase de réglage. Or on ne dispose pas toujours d'une base seg-

mentée phonétiquement, y compris lors de l'apprentissage des modèles. Cette expérimentation nous montre que l'insertion des paramètres NPC dans un système HMM peut malgré tout se faire même lorsque l'on ne dispose pas de segmentation phonétique. Des algorithmes comme l'algorithme de Viterbi permettent en effet d'obtenir une segmentation phonétique à chaque étape de l'apprentissage. Le réglage de l'extracteur s'effectuerait alors simultanément avec l'apprentissage des modèles HMM.

Enfin, plus que d'obtenir des scores dans l'absolu, une étude intéressante consisterait à caractériser les distributions statistiques des paramètres NPC. Nous avons en effet choisi d'utiliser un classifieur (que ce soit le modèle PMC initial ou le modèle ANN-HMM utilisé ici) permettant de s'abstenir de l'hypothèse de distribution multi-gaussienne diagonale des paramètres. Il n'y pas de raison en soit de penser que cela désavantage les paramètres MFCC. En revanche, cela pourrait montrer que, dans le cas où les paramètres NPC respecteraient mal cette hypothèse, ils ne pourraient être que moins performants dans un système de RAP plus traditionnel de type GMM-HMM (estimation des densités de probabilités à l'aide de mixtures de gaussiennes).

6. Conclusion

Nous avons présenté dans cet article un nouveau modèle d'analyse adaptative non linéaire du signal de parole, appliqué à l'extraction de caractéristiques pour le reconnaissance de phonèmes.

Nous avons proposé trois algorithmes d'adaptation et montré que la prise en compte, d'une part des caractéristiques non linéaires du signal (modèle NPC), et d'autre part d'informations de classe d'appartenance des signaux dès l'étape d'analyse (modèles NPC-2 et DFE-NPC) était pertinente pour les applications en reconnaissance automatique de la parole. Nous avons présenté un ensemble de résultats expérimentaux comparatifs montrant une amélioration significative des scores en reconnaissance de phonèmes sur des signaux de parole extraits des bases internationales TIMIT et NTIMIT. Fondée sur la notion de partage des poids d'un modèle connexionniste de type PMC et sur la minimisation des fonctions de coûts adéquates, nous avons démontré, tant d'un point de vue théorique qu'expérimental, le bien fondé de l'approche NPC proposée.

Enfin, nous avons présenté une expérimentation des paramètres NPC dans le cadre d'une application de reconnaissance de mots isolés à l'aide d'un système de type ANN-HMM.

Nous travaillons actuellement à l'incorporation des paramètres NPC dans un système de reconnaissance de locuteur plus complet en vue de leur validation dans le cadre de campagnes d'évaluation nationales et internationales en segmentation et regroupement de locuteurs.

Références

[1] University of Pennsylvania Linguistic Data Consortium. The nist darpa-timit acoustic-phonetic continuous speech corpus: a multi speakers data base, 1990.

[2] K. AIKAWA, H. SINGER, H. KAWAHARA, and Y. TOHKURA, A dynamic cepstrum incorporating time-frequency masking and its application to continuous speech recognition. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 2, pp. 668-671, 1993.

[3] B.S. ATAL and S. L. HANAUER, Speech analysis and synthesis by linear prediction of the speech wave. *Journal of the Acoustical Society of America*, 50:637-655, 1971.

[4] M. BACCHIANI and K. AIKAWA, Optimization of time-frequency masking filters using the minimum classification error criterion. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 2, pp. 197-200, 1994.

[5] A. BIEM, *Neural models for extracting speaker characteristics in speech modelization system*. PhD thesis, Paris VI, 1997.

[6] A. BIEM and S. KATAGIRI, Feature extraction based on minimum classification error/generalized probabilistic descent method. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 2, pp. 275-278, 1993.

[7] A. BIEM and S. KATAGIRI, Filter bank design based on discriminative feature extraction. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 485-488, 1994.

[8] M. BIRGMEIER, Nonlinear prediction of speech signals using radial basis function networks. In *Proceedings of EUSIPCO96*, pp. 459-462, September 1996.

[9] C. M. BISHOP, Novelty detection and neural network validation. In *IEE proceedings: Vision, Image and Signal Processing. Special Issue on applications of neural networks*, volume 141, pp. 217-222, 1994.

[10] C. M. BISHOP, *Neural Networks for Pattern Recognition*. Clarendon Press - Oxford, 1995.

[11] H. BOURLARD, H. HERMANSKY, and N. MORGAN, Towards increasing speech recognition errors. *Speech Communication*, 18:205-231, 1996.

[12] H. BOURLARD and N. MORGAN, Hybrid hmm/ann systems for speech recognition: Overview and new research directions. *Lecture Notes In Computer Science*, 1387:389-417, 1997.

[13] H. BOURLARD and Y. KAMP, Auto-association by multilayer perceptron and singular value decomposition. *Biological Cybernetics*, 59:291-294, 1988.

[14] T. BURROWS, *Speech Processing with Linear and Neural Network Models*. PhD thesis, Cambridge University, 1996.

[15] M. CHETOUANI, B. GAS, and J. L. ZARADER, Maximisation of the modelisation error ratio for neural predictive coding. In *NOLISP'03 (ISCA Tutorial and Research Workshop on NOn-Linear Speech Processing)*, pp. 77-80, 2003.

[16] P. CHEVALIER, P. DUVAUT, and B. PICINBONO, Le filtrage de volterra transverse réel et complexe en traitement du signal. *Traitement du Signal*, 7(5):451-476, 1990.

[17] S. B. DAVIS and P. MELMERSTEIN, Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 28(4):357-376, 1980.

[18] A. DE LA TORRE, A. M. PEINADO, A. J. RUBIO, V. E. SÁNCHEZ, and J. E. DÍAZ, An application of minimum classification error to feature space transformations for speech recognition. *Speech Communication*, 20:273-290, 1996.

[19] F. DÍAZ -DE-MARÍA and A. R. FIGUEIRAS-VIDAL, Nonlinear prediction for speech coding using radial basis functions. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 788-791, 1995.

[20] G. DREYFUS, O. MACCHI, S. MARCOS, O. NERRAND, L. PERSONNAZ, ROUSSEL-RAGOT, D. URBANI, and C. VIGNAT, Adaptive training of feedback neural networks for non linear filtering. *Neural Networks for signal processing*, 2:550-559, 1992.

[21] M. FAUNDEZ-ZANUY and A. ESPOSITO, Nonlinear speech processing applied to speaker recognition. In *Conf. on the Advent of Biometrics on the Internet*, 2002.

[22] S. FURUI, Speaker-independant isolated word recognition using dynamic features of speech spectrum. *The Journal of the Acoustical Society of America*, pp. 1738-1752, 1986.

[23] B. GAS, J. L. ZARADER, and C. CHAVY, A new approach to speech coding : The neural predictive coding. *Journal of Advanced Computational Intelligence*, 4(1):120-127, 2000.

[24] B. GAS, J. L. ZARADER, C. CHAVY, and M. CHETOUANI, Discriminant neural predictive coding applied to phoneme recognition. *Neurocomputing*, 56:141-166, 2004.

[25] T. GAUTAMA, D.P. MANDIC, and M.M VAN HULLE, On the characterisation of the deterministic/stochastic and linear/nonlinear nature of time series. Technical Report DPM-04-5, Imperial College London, 2004.

[26] F. GIROSI and T. POGGIO, Representation properties of networks: Kolmogorov's theorem is irrelevant. *Neural Computation*, 1(4):465-469, 1989.

[27] Y. GONG, Speech recognition in noisy environments: A survey. *Speech Communication*, 16:261-291, 1995.

[28] R. HECHT-NIELSEN, Kolmogorov's mapping neural network existence theorem. In *Proceedings of the International Conference on Neural Networks*, pp. 11-13, 1987.

[29] H. HERMANSKY, Perceptual linear predictive (plp) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738-1752, 1990.

[30] H. HERMANSKY, Should recognizers have ears ? *Speech Communication*, 25:3-27, 1998.

[31] H. HERMANSKY and N. MORGAN, Rasta processing of speech. *IEEE Transactions on Speech and Audio Processing*, 2:587-589, 1994.

[32] F. ITAKURA, Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 23:67-72, 1975.

[33] F. ITAKURA and S. SAITO, Analysis synthesis telephony based on the maximum likelihood method. In *Proceedings of the 6th International Congress on Acoustic*, pp. 17-20, 1968.

[34] B. H. JUANG and S. KATAGIRI, Discriminative learning for minimum error classification. *IEEE Transactions on Signal Processing*, 40(12):3043-3054, december 1992.

[35] S. KARAJEKAR, *Analysis of variability in Speech with Applications to Speech and Speaker Recognition*. PhD thesis, OGI, Portland, USA, 2002.

[36] S. KATAGIRI, *Handbook of Neural Networks for Speech Processing*. Artech House, 2000.

[37] H. KATSUURA and D. A. SPRECHER, Computational aspects of kolmogorov's superposition theorem. *Neural Networks*, 7(3):455-461, 1994.

[38] T. KAWAHARA and S. DOSHITA, Phoneme recognition by combining discriminant analysis and hmm. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 557-560, 1991.

[39] T. KAWAHARA, T. OGAWA, S. KITAZAWA, and S. DOSHITA, Phoneme recognition by combining bayesian linear discriminations of selected pairs of classes. In *International Conference on Speech and Signal Processing (ICASSP)*, p. 78, 1990.

[40] G. I. KECHRIOTIS and E. S. MANOLAKOS, Using neural networks for nonlinear and chaotic signal processing. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 465-468, 1993.

[41] D. J. KERSHAW, *Phonetic Context-Dependancy In a Hybrid ANN/HMM Speech Recognition System*. PhD thesis, St. John's College, University of Cambridge, 1997.

[42] A. KOLMOGOROV, On the representation of continuous functions of many variables by superpositions of continuous functions of one variable and addition. *Doklady Akademii Nauk USSR*, 114(5):953-956, 1957.

[43] G. KUBIN, *Speech coding and synthesis*, chapter Nonlinear processing of Speech, pp. 557-609. W.B. Kleijn and K.K. Paliwal Editors, Elsevier Science, 1995.

[44] V. KURKOVA, Kolmogorov's theorem and multilayer neural networks. *Neural Networks*, 5:501-506, 1992.

[45] K. J. LANG, A. H. WAIBEL, and G.E. HINTON, A time-delay neural network architecture for isolated word recognition. *Neural Networks*, 3:23-43, 1990.

[46] A. LAPEDES and R. FARBER, Nonlinear signal processing using neural networks: Prediction and system modelling. *Internal Report, Los Alamos National Laboratory*, july 1987.

[47] J. H. LEE, H. Y. JUNG, T. W. LEE, and S. Y. LEE, Speech feature extraction using independent component analysis. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 3, pp. 1631-1634, 2000.

[48] J. C. LUCERO, A theoretical study of the hysteresis phenomenon at vocal fold oscillation onset-offset. *Journal of Acoustic Society of America*, 1:423-431, 1999.

[49] N. MA, T. NISHI, and G. WEI, On a code-excited nonlinear predictive speech coding (cenlp) by means of recurrent neural networks. *IEICE Transactions fundamentals, spec. issue on digital signal processing*, E81-A(8):1628-1634, 1998.

[50] N. MA and G. WEI, Speech coding with nonlinear local prediction model. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 2, pp. 1101-1104, 1998.

[51] P. M. MARAGOS and A. POTAMIANOS, Fractal dimensions of speech sounds: computation and application to automatic speech recognition. *Journal of Acoustic Society of America*, 3:1925-1933, 1999.

[52] P. J. MORENO and R. N. STERN, Sources of degradation of speech recognition in the telephone network. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 109-112, 1994.

[53] A. PAGÈS-ZAMORA, M. A. LAGUNAS, M. NÁJAR, and A.I. PÉREZ-NEIRA, The k-filter: A new architecture to model and design non-linear systems from kolmogorov's theorem. *Signal Processing*, 44:249-267, 1995.

[54] D. POVEY, B. KINGSBURY, L. MANGU, G. SAON, H. SOLTAU, and G. ZWEIG, fmpe: Discriminatively trained features for speech recognition. In *Proc. of DARPA EARS RT-04 Workshop*, 2004.

[55] D. POVEY and P.C. WOODLAND, Minimum phone error and i-smoothing for improved discriminative tranning. In *International Conference on Speech and Signal Processing (ICASSP)*, 2002.

[56] V. C. RAYCAR, B. YEGNANARAYANA, and S. R. DURAISWAMY, Speaker localization using excitation source information in speech. *IEEE Transactions on Speech and AUdio Processing*, 2004.

[57] W. REICHL, S. HARENGEL, F. WOLFERSTETTER, and G. RUSKE, Neural networks for nonlinear discriminant analysis in continuous speech recognition. In *Eurospeech*, pp. 537-540, 1995.

[58] G. SAON, M. PADMANABHAN, R. GOPINATH, and S. CHEN, Maximum likelihood discriminant feature spaces. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 2, pp. 1229-1132, 2000.

[59] J. SCHOENTGEN, Non-linear signal representation and its application to the modelling of the glottal waveform. *Speech communication*, 9:189-201, 1990.

[60] J. SCHOENTGEN, On the bandwidth of a shaping function model of the phonatory excitation signal. In *No Linear Speech Processing Workshop (NOLISP'03)*, 2003.

[61] S.S. STEVENS and J. WOLKMANN, The relation of pitch of frequency: A revised scale. *American Journal of Psychology.*, 53:329-353, 1940.

[62] H. STRIK, Automatic parametrization of differentied glottal flow: comparing methods by means of synthetic flow pulses. *Journal of Acoustic American society*, 5:2659-2669, may 1998.

[63] H. M. TEAGER and S. M. TEAGER, Evidence for non linear sound production mechanisms in the vocal tract. *Speech Production and Speech Modeling*, 55:241-261, July 1989.

[64] J. THEILER, S. EUBANK, A. LONGTIN, B. GALDRIKIAN, and J. FARMER, Testing for nonlinearity in time series: the method of surrogate data. *Physica D*, 58:77-94, 1992.

[65] J. THYSSEN, H. NIELSEN, and S. D. HANSEN, Non-linear short-term prediction in speech coding. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 185-188, 1994.

[66] B. TOWNSHEND, Non linear prediction of speech. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 425-428, 1991.

[67] A. G. VITUSHKIN, On hilbert's thirteenth problem. *Dokl. Akad. Nauk. SSSR*, 95:701-704, 1954.

[68] A. H. WAIBEL, T. HANAZAWA, G. E. HINTON, K. SHIKANO, and K. J. LANG, Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustic, Speech, and Signal processing*, 37(3):328-339, march 1989.

[69] A. R. WEBB, Functional approximation by feed-forward networks: a least square approach to generalisation. *IEEE Transactions on Neural Networks*, 5(3):363-371, 1994.

[70] B. WIDROW, 30 years of adaptative neural networks: Perceptron, madaline and backpropagation. *Proc. of the IEEE*, 78:1415-1442, 1990.

[71] D. YUK and J. FLANAGAN, Telephone speech recognition using neural networks and hidden markov models. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 157-160, 1999.

[72] S. A. ZAHORIAN, D. QIAN, and A.J. JAGHARGHI, Acoustic-phonetic transformations for improved speaker-independent isolated word recognition. In *International Conference on Speech and Signal Processing (ICASSP)*, volume 1, pp. 561-564, 1991.

[73] E. ZWICKER and E. TERHARDT, Analytical expressions for critical band rate and critical bandwidth as a function of frequency. *The journal of The Acoustical Society of America*, 68:1523-1525, 1980.

Bruno **Gas**

Bruno Gas est Docteur de l'Université d'Orsay (1994). Maître de Conférences à l'Université Pierre et Marie Curie depuis 1995 et habilité à diriger des recherches depuis 2005, ses domaines de recherche concernent l'étude des modèles connexionnistes pour le traitement non linéaire des signaux (LIDAR et parole).

Mohamed **Chetouani**

Mohamed Chetouani est titulaire du DEA de Robotique et des Systèmes Intelligents et Docteur de l'Université Pierre et Marie Curie (2004). Il a été chercheur invité à l'Université de Stirling (Ecosse) ainsi qu'à l'Ecole Polytechnique de Mataro (Barcelone). Il est Maître de Conférences à l'Université Pierre et Marie Curie depuis 2005. Ses domaines de recherche concernent le traitement non-linéaire du signal.

Jean-Luc **Zarader**

Jean-Luc Zarader est Docteur de l'Université Pierre et Marie Curie, Professeur à l'Université Pierre et Marie Curie depuis 2002. Ses domaines de recherche couvrent le traitement adaptatif du signal (signaux LIDAR et ultrasonores) et l'extraction de caractéristiques des signaux de parole à l'aide de méthodes neuronales.