

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA  
RECHERCHE SCIENTIFIQUE  
UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE  
HOUARI BOUMEDIENNE  
FACULTÉ DE PHYSIQUE**



**MEMOIRE**

Présenté pour l'obtention du diplôme de MAGISTER  
En: PHYSIQUE  
Spécialité: Physique Théorique des Basses et Moyennes Energies

Par :Mme Hassiba TAIBI -IRATNI

Sujet :

**Adaptation des réseaux de neurones-ondelettes  
à l'approximation de tâches et à l'approximation des  
potentiels sextiques**

Soutenu publiquement, le 13 /04 /2010, devant le jury composé de :

Mme. F.Z. IGHEZOU	Professeur	U.S.T.H.B	Président
Mme. F. CHAFA-MEKIDECHE	Professeur	U.S.T.H.B	Directrice de mémoire
Mr . M. TRIBECHÉ	Professeur	U.S.T.H.B	Examinateur
Mme. A.H. HAMICI	Maître de conférences	U.S.T.H.B	Examinatrice
Mr. N. BENREKAA	Maître de conférences	U.S.T.H.B	Examinatrice

## Remerciements

*Ce travail a été réalisé au sein du Laboratoire de Physique Théorique de la Faculté des Sciences-Physiques de l'Université des Sciences et de la Technologie Houari Boumediene (USTHB) d'Alger.*

*Mes plus vifs remerciements sont adressés à ma directrice de thèse Madame F. MEKIDECHE-CHAFI, Professeur à l'U.S.T.H.B, pour son encadrement sans faille durant cette thèse, sa disponibilité, son suivi, ses conseils et son soutien. Son souci de valorisation des travaux accomplis, a beaucoup contribué à l'aboutissement de ce travail de thèse.*

*Je tiens aussi à exprimer ma profonde gratitude à Madame L. AIT GOUGAM-TRIBECHE, Docteur à l'U.S.T.H.B, pour son aide sans faille, ses conseils qu'elle m'a prodigués tout au long de ce travail et la pertinence de ses critiques et ses nombreuses connaissances dans le domaine des réseaux de neurones et des ondelettes.*

*Je souhaite aussi témoigner ma reconnaissance à Monsieur A. Chouchaoui, Professeur à l'U.S.T.H.B, pour m'avoir fait découvrir la méthode des potentiels quasi-exactement solubles.*

*Je tiens à témoigner toute ma reconnaissance à Madame F.Z. IGHEZOU, Professeur à l'U.S.T.H.B, qui nous a honorés en acceptant de présider le jury de la présente thèse.*

*Je tiens aussi à témoigner ma reconnaissance à Monsieur M. TRIBECHE, Professeur à l'U.S.T.H.B, pour ses remarques, tous les conseils qu'il m'a prodigués et d'avoir accepté de faire partie de ce jury.*

*Je souhaite aussi remercier Madame A.H. HAMICI et Monsieur N. BENREKAA, Maîtres de conférences à l'USTHB, pour avoir accepté de juger ce travail.*

*Enfin, je veux rendre hommage à tous ceux de mes proches qui m'ont soutenue durant ces années, en particulier mes parents, mon mari, mon frère et mes sœurs.*

# *Introduction principale*

La problématique d'approximation de tâches est un sujet d'actualité. Opter pour un approximateur parmi d'autres dépend essentiellement de son aptitude à approcher la fonction désirée avec la structure la plus simple possible et un nombre de paramètres le plus petit. Les réseaux de neurones artificiels dont l'une des propriétés fondamentales est l'approximation universelle parcimonieuse, c'est à dire que le modèle fait intervenir le plus petit nombre possible de paramètres ajustables, fait de ceux-ci une représentation mathématique très avantageuse pour la modélisation statique et dynamique des processus physiques. La conception des réseaux de neurones artificiels s'inspire du fonctionnement du neurone biologique. En effet, l'idée de s'inspirer du cerveau pour construire une machine intelligente, douée de propriétés cognitives et capable d'apprentissage est un vieux et ambitieux projet humain [1]. La science et la technique sont nées avec l'homme, par l'homme et pour l'homme. Tout a commencé avec la volonté de comprendre et d'agir pour vivre. Comprendre, c'est à partir d'informations multiples, raisonner, c'est établir entre différents phénomènes des liens, rechercher des causes et en déduire des effets. Comprendre, c'est vérifier par l'expérimentation, par la pratique ce que la pensée a construit, c'est associer étroitement les idées au réel, la pensée à l'action [2]. Puisque la pensée est produite par le cerveau ou en est une propriété, commencer à étudier comment celui-ci fonctionne, c'est incontestablement cette approche qui a conduit à l'invention des réseaux de neurones artificiels dont l'intention est d'imiter certaines des fonctions du cerveau humain. Cette formidable machine qu'est le cerveau est composée de plus de 10 milliards d'unités élémentaires, les neurones, reliés entre eux par une multitude de connexions, en moyenne 10 000 par neurone, qui travaillent de façon concertée à l'élaboration des tâches les plus complexes. Chaque neurone est relié à d'autres par des synapses, c'est l'interconnexion de ces neurones qui est appelée « réseau de neurones » [1]. L'histoire des réseaux de neurones est donc tissée à travers des découvertes conceptuelles et des développements technologiques survenus à diverses

époques [3]. Pour saisir la notion de cette nouvelle discipline qui est les réseaux de neurones, il est indispensable d'explorer ses origines [3].

La première vague d'intérêt aux réseaux de neurones a vu le jour en 1943 grâce à deux neurologues de l'université de Chicago, J.Mc Culloch et W. Pitts, qui menèrent les premiers travaux sur les réseaux de neurones et proposèrent le premier modèle mathématique et informatique du neurone biologique communément appelé le neurone formel [4]. J.Mc Culloch et W. Pitts sont les premiers à montrer que des réseaux de neurones formels simples peuvent réaliser des fonctions logiques ou arithmétiques [5]. Cependant, leurs travaux n'ont pas donné d'indications sur une méthode pour adapter les coefficients synaptiques. Cette question a connu un début de réponse grâce aux travaux du physiologiste américain D. Hebb. En 1949, D. Hebb, a donc proposé une théorie fondamentale pour l'apprentissage et une loi de modification des propriétés des connexions entre les neurones [5,6]. Les premiers succès de cette méthode furent présentés par F. Rosenblatt en 1957 qui présenta le perceptron, premier modèle d'un réseau de neurones artificiels. F.Rosenblatt l'a appliqué au domaine de la reconnaissance de formes en démontrant un réel processus d'apprentissage en mettant en œuvre les idées de D. Hebb, J.Mc Culloch et W. Pitts. La différence entre ce modèle et celui de J.Mc Culloch et W. Pitts est sa faculté d'apprentissage basée sur la règle de Hebb [6,7]. En 1960, B.Windrow, un automaticien, a développé le modèle Adaline (Adaptative Linear Element) qui ressemble au perceptron mais avec une amélioration de la loi d'apprentissage [5,6]. Celle-ci est à l'origine de l'algorithme de la rétropropagation de gradient très utilisé aujourd'hui avec les perceptrons multicouches [5]. En 1969, S. Papert et M. Minsky publièrent un ouvrage qui met en exergue les limites théoriques du perceptron, notamment, concernant l'impossibilité de traiter par ce modèle des problèmes non linéaires [5]. Il a fallu attendre 1982 pour voir, avec le travail de J.Hopfield, un physicien reconnu, le renouveau d'intérêt pour les réseaux de neurones artificiels. Au travers d'un article court, clair et bien écrit, il présenta une théorie du fonctionnement et des possibilités des réseaux de neurones [5].

J. Hopfield fixa préalablement le comportement à atteindre pour son modèle et construisit, à partir de là, la structure et la loi d'apprentissage correspondant au résultat escompté. Ce modèle est aujourd'hui encore très utilisé pour des problèmes d'optimisation [6]. La machine de Boltzmann, en 1983, est le premier modèle connu apte à traiter de manière satisfaisante les limitations recensées dans le cas du perceptron. Mais l'utilisation pratique s'avère difficile, la convergence de l'algorithme étant extrêmement longue [5]. En 1985, la rétropropagation du gradient apparaît. C'est un algorithme d'apprentissage adapté aux réseaux de neurones multicouches. Depuis cette découverte, nous avons la possibilité de réaliser une fonction non linéaire d'entrée/sortie sur un réseau en décomposant cette fonction en une suite d'étapes linéairement séparables. De nos jours, les réseaux multicouches et la rétropropagation du gradient restent le modèle le plus étudié et le plus adapté pour les applications [6]. Les réseaux neuronaux ont connu un grand essor suite aux nombreuses applications potentielles et la disponibilité de grands calculateurs. En fait, les réseaux de neurones sont utilisés dans des domaines divers à cause de leurs propriétés et, en particulier, leur capacité d'apprentissage et de généralisation [6]. L'une des propriétés fondamentales des réseaux de neurones, l'approximation universelle parcimonieuse, fait de ceux-ci une représentation mathématique très avantageuse pour la modélisation statique et dynamique des processus [8].

A l'instar des réseaux de neurones, les ondelettes, qui sont des familles de fonctions issues du traitement du signal et de l'image, possèdent la propriété d'approximateurs universels. Elles ont aussi été utilisées pour résoudre des problèmes d'approximation de fonctions [8]. En fait, de tout temps, de nombreux auteurs se sont intéressés à l'analyse de signaux qu'ils soient électriques, sismiques, optiques ou géophysiques. Il existe, à cet effet, différentes méthodes d'analyse. Nous citons, en particulier, la méthode de Fourier, la méthode de Gabor, ou encore appelée méthode de Fourier à fenêtre glissante et plus récemment, la méthode des ondelettes [9]. L'étymologie du mot ondelette n'est, en soi, pas très compliquée. Les ondelettes utilisées par Jean Morlet [10] avaient, en effet une allure de petites ondes, d'où cette appellation.

Vers 1975, ce géophysicien, qui travaillait à l'époque pour ELF Aquitaine, a créé de petites fonctions mathématiques ayant certaines propriétés intéressantes, et ceci afin de sonder les différentes couches géologiques et ainsi détecter des nappes de pétrole. J.Morlet est donc l'un des artisans majeurs des ondelettes [10]. En effet, l'analyse par ondelettes a été introduite au début des années 80 dans un contexte d'analyse de signal et d'exploration pétrolière par le géophysicien J. Morlet [11]. C'est en s'intéressant à l'étude des signaux sismiques que Morlet s'est aperçu que les décompositions spectrales classiques étaient inadaptées à l'analyse de ces signaux. En effet, il s'agissait à l'époque de donner une représentation des signaux permettant la mise en valeur simultanément des informations temporelles et fréquentielles (localisation temps- fréquence) [12], facilitant par là l'identification des caractéristiques physiques du signal. Les ondelettes n'ont, depuis lors, pas cessé de se développer et de trouver de nouveaux champs d'applications [13]. En 1984, P.Goupillaud, A.Grossman et J.Morlet [12], poussés par les exigences croissantes de la recherche d'hydrocarbures, proposèrent une méthode de reconstruction des signaux sismiques multidimensionnels permettant une restauration des hautes fréquences à l'aide d'une représentation temps-fréquence.

L'analyse par ondelettes découle tout naturellement de l'analyse de Fourier. Il est donc naturel de commencer par celle-ci afin de saisir les bases de cette nouvelle technique [10]. La transformation de Fourier, la plus ancienne méthode d'analyse, permet de translater une fonction du domaine temporel au domaine fréquentiel. Elle donne le maximum d'informations sur la répartition des fréquences mais perd totalement celles relatives au temps. La transformée de Fourier (TF) fournit une analyse spectrale globale et non locale. Pour remédier à ce problème et afin d'améliorer la localisation temporelle tout en préservant la localisation fréquentielle. Vers les années 40, D.Gabor [14], a développé le concept de la transformée de Fourier à fenêtre glissante qui n'est qu'une adaptation locale de l'analyse de Fourier. Le principe de cette méthode est de segmenter le signal étudié en parties et de calculer ensuite la transformée de Fourier. Cette méthode présente l'inconvénient majeur d'avoir une

largeur de fenêtre fixe, or pour étudier un signal qui a en général une allure irrégulière, il est intéressant de changer la forme de la fenêtre analysante en temps [15]. J. Morlet a constaté cette insuffisance, et il a proposé une transformée semblable à celle de la transformée de Fourier à fenêtre glissante dont la taille de la fenêtre est variable, en introduisant le paramètre d'échelle [16]. Cette transformée permet une représentation du signal simultanément dans le temps et dans l'espace et cela dans une combinaison linéaire des fonctions élémentaires appelées ondelettes, localisées en différents points de l'espace et ayant des tailles différentes. Ces fonctions élémentaires sont toutes construites à l'aide d'une fonction ondelette mère unique, par translation mais aussi par contraction ou dilatation de celle-ci [17].

En 1984, A. Grossmann et J. Morlet [12] ont établi un cadre rigoureux au concept de cette nouvelle décomposition espace-échelle et l'analyse par ondelettes a connu alors un développement considérable et a constitué un nouvel outil puissant d'analyse des signaux non stationnaires. Elle va connaître un essor important notamment par les travaux de I. Daubechies et Y. Meyer qui établirent l'aspect discret de la décomposition en ondelettes [18].

Le cadre de cette thèse s'inscrit dans l'adaptation de l'approche des réseaux de neurones-ondelettes pour l'approximation des fonctions. Ce mémoire s'organise de la manière suivante:

Le chapitre I est consacré à la présentation d'un bref rappel de l'analyse par transformée de Fourier et par celle de Fourier à fenêtre glissante ainsi que leurs limites. Nous rappelons aussi les propriétés fondamentales des ondelettes, nous donnons un aperçu succinct sur l'analyse par ondelettes puis nous présentons quelques exemples d'ondelettes continues.

Le chapitre II expose les différentes notions de base qui constituent les fondements théoriques des réseaux de neurones et de leurs apprentissages ainsi que leurs domaines d'applications.

Le chapitre III représente le socle de notre travail qui consiste à utiliser le modèle des réseaux de neurones-ondelettes pour l'approximation d'une tâche bien définie. Nous avons utilisé pour cela, une architecture bien définie d'un réseau en considérant différentes fonctions d'activation. Une étude comparative a été faite et a montré que l'ondelette « chapeau mexicain » est la mieux adaptée pour approcher la tâche choisie. Nous l'avons alors utilisée pour montrer l'influence du nombre de neurones ainsi que le nombre d'itérations dans l'amélioration de nos résultats.

Le chapitre IV examine une application des réseaux-ondelettes à un problème physique bien déterminé. Nous avons choisi de tester la fiabilité de notre approche en l'appliquant pour l'approximation des potentiels sextiques. Nous avons débuté par l'approximation d'un ensemble de potentiels sextiques dont la forme analytique est connue, pour chercher une forme approchée du potentiel sextique choisi, généralement utilisé pour interpréter les spectres vibrationnels des molécules, à l'aide de notre approche. Nous avons par la suite utilisé cette forme approchée pour le calcul des énergies de l'état fondamental et du premier état excité. Une comparaison de nos valeurs des énergies avec celles obtenues par une autre méthode, celle des potentiels quasi- exactement solubles (*Q.E.S*), a été faite.

L'adéquation entre ces différents résultats nous permet d'étendre l'utilisation de notre approche au cas où la forme analytique du potentiel ne serait pas connue.

**CHAPITRE I :**  
**LES ONDELETTES**

## **I.1 Introduction**

L'analyse par ondelettes a été introduite au début des années 80 dans un contexte d'analyse de signal et d'exploration pétrolière par le géophysicien J. Morlet [11]. C'est en s'intéressant à l'étude des signaux sismiques que J. Morlet s'est aperçu que les décompositions spectrales classiques étaient inadaptées à l'analyse de ces signaux. En effet, il s'agissait à l'époque de donner une représentation des signaux permettant la mise en valeur simultanément des informations temporelles et fréquentielles (localisation temps- fréquence), facilitant par là l'identification des caractéristiques physiques du signal. Depuis, les ondelettes n'ont cessé de se développer et de trouver de nouveaux champs d'application [13]. En 1984, P.Goupillaud, A.Grossman et J.Morlet [12], poussés par les exigences croissantes de la recherche d'hydrocarbures, proposèrent une méthode de reconstruction des signaux sismiques multidimensionnels permettant une restauration des hautes fréquences à l'aide d'une représentation temps-fréquence. C'est ainsi que le besoin des ondelettes, famille de fonctions déduites par opération de translation et dilatations d'une même fonction appelée ondelette mère, s'est fait ressentir, en remarquant que la transformée de Fourier, qui a dominé dès le début du 19<sup>ème</sup> siècle, perd le contrôle de la variable temporelle et reste incapable de décrire localement le comportement fréquentiel des signaux [15]. Donc, pour tirer du signal des informations pertinentes, il faut disposer d'une méthode qui utilise les deux paramètres temps et fréquence et qui mette aussi en évidence leur corrélation [12]. Plusieurs méthodes existent à cet effet : la méthode de Fourier, la méthode de Gabor et récemment, la méthode des ondelettes.

Afin de faciliter la compréhension de notre travail, nous allons présenter, dans ce qui suit et brièvement, des rappels de fondements mathématiques qui sous-tendent ces différentes méthodes.

## **I.2 La méthode de Fourier**

L'analyse de Fourier est une des bases majeures de la physique et des mathématiques. C'est la plus ancienne méthode d'analyse et elle consiste à présenter le signal comme une superposition d'ondes sinusoïdales de toutes les fréquences possibles avec des

amplitudes variables. Ces amplitudes forment alors une fonction de la fréquence appelée spectre continu des fréquences du signal : c'est la transformée de Fourier du signal (TF). Elle est calculée à l'aide de l'intégrale de Fourier. La (TF) est porteuse de précieuses informations sur le signal analysé. Nous savons par exemple que si la (TF) n'a que des faibles valeurs pour des valeurs élevées de la variable de fréquence alors ceci signifie que le signal varie lentement. Inversement, si la (TF) prend des valeurs importantes pour les hautes fréquences, le signal contient alors une quantité non négligeable de hautes fréquences, et donc varie rapidement, au moins dans certaines zones. C'est précisément là, que nous touchons du doigt l'une des limitations importantes de l'analyse de Fourier à savoir l'incapacité de la (TF) à localiser les portions du signal dans lesquelles les variations sont rapides, ni celles où elles sont lentes [13].

Si l'on considère une fonction  $f(t)$  appartenant à  $L^2(\mathbb{R}, dt)$ , ensemble des fonctions de  $\mathbb{R}$  dans  $\mathbb{R}$ , de carrés intégrables sur  $\mathbb{R}$ , la transformée de Fourier de  $f(t)$  est donnée par :

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} dt \quad (\text{I-1})$$

où  $\omega$  représente la fréquence.

Dans le cas d'une analyse en fréquences d'un signal, ce qui est le cas pour l'analyse par transformée de Fourier, les fonctions analysantes sont les fonctions trigonométriques  $e^{-i\omega t}$  de fréquence  $\omega$ , qui sont délocalisées dans le temps. Projeter le signal sur ces fonctions (analyse spectrale) renseigne sur son caractère fréquentiel. La (TF) peut être donc vue comme une représentation à base de sinusoides qui sont bien localisées en fréquences, mais pas en temps car leur support est infini. Les inconvénients de l'analyse par transformée de Fourier proviennent donc du fait que les fonctions analysantes sont à support de longueur infinie [15]. Les différentes fréquences contenues dans le signal sont décrites par une fonction  $\hat{f}(\omega)$  indépendante du temps donc la (TF) donne une analyse spectrale fréquentielle non localisée en temps avec disparition de tous les aspects temporels dans le spectre de  $\hat{f}(\omega)$ . La (TF) ne peut donc s'adapter à l'étude des signaux non stationnaires et une analyse en temps-

fréquence s'avère indispensable. La méthode de Fourier ne répondant pas au besoin des analystes, une nouvelle méthode est apparue : la représentation de Gabor ou analyse de Fourier à fenêtre glissante.

### I.3 Méthode de Gabor

Pour répondre au problème de localisation de la transformée de Fourier, D.Gabor a introduit vers les années 40, l'approche de la transformée de Fourier à fenêtre glissante. L'idée de base de cette méthode est d'introduire dans l'analyse de Fourier usuelle une notion de localité temporelle. Cela se réalise en découpant le signal en petits morceaux et de calculer ensuite leurs transformées de Fourier. Dans la pratique, il s'agit de remplacer la fonction analysée par un produit de celle-ci par une fenêtre convenablement choisie au préalable, possédant de bonnes propriétés de localisation, puis en calculant la transformée de Fourier du produit ainsi formé. Cette opération est ensuite renouvelée avec des copies translatées de la fenêtre ce qui conduit à une analyse locale en tout point [17]. Si nous choisissons de noter  $g(t)$  la fenêtre et  $f(t)$  la fonction analysée, le résultat est alors donné par [17]:

$$G(b, \omega) = \int_{-\infty}^{+\infty} f(t) g(t-b) e^{-i\omega t} dt \quad (\text{I-2})$$

$b$  est le paramètre temporel et  $\omega$  est la fréquence.

L'ensemble de ces transformées de Fourier ainsi localisées forme la transformée de Gabor du signal qui fournit donc une analyse fréquentielle locale. Si la technique de Gabor marquait déjà un grand pas dans le domaine de l'analyse en temps-fréquence, elle n'était cependant pas encore suffisante. En effet, elle présente un inconvénient majeur; la taille et la forme de la fenêtre qui restent constantes au cours de l'analyse et qui sont insensibles aux variations de la fréquence. Or, pour étudier un signal qui présente en général une allure irrégulière, il est toujours nécessaire de changer la forme de la fenêtre en temps [15]. D'où la nécessité d'avoir recours à une nouvelle méthode.

### I.4 Méthode des Ondelettes

C'est en s'intéressant à l'étude des signaux sismiques que Morlet [12], en 1948, s'est aperçu que les décompositions spectrales classiques étaient inadaptées à l'analyse de

ces signaux. En effet, il s'agissait à l'époque de donner une représentation des signaux permettant la mise en valeur simultanément des informations temporelles et fréquentielles (localisation temps-fréquence), facilitant par là l'identification des caractéristiques physiques du signal. Les ondelettes n'ont, depuis lors, pas cessé de se développer et de trouver de nouveaux champs d'applications [13].

Dans un souci de clarification et de compréhension, nous présentons dans ce qui suit un rappel des principales parties concernant le formalisme qui sous-tend la théorie des ondelettes en insistant principalement sur les parties intéressant notre travail.

#### I.4.1 La transformée en ondelettes

J. Morlet a proposé pour l'étude des signaux sismiques une transformée semblable à celle de la transformée de Fourier à fenêtre glissante dont la taille de la fenêtre est variable, ceci grâce à un paramètre d'échelle. C'est une méthode qui consiste à décomposer les signaux sur une famille d'ondelettes  $\Psi$ , dilatée ou compressée d'un facteur d'échelle  $a$  et translatée d'un paramètre de translation  $b$ , afin de couvrir tout le domaine de définition de la fonction décomposée. Une famille d'éléments temps-fréquences  $\Psi_{(a,b)}(t)$  associée à  $\Psi$ , s'obtient donc en dilatant d'un paramètre d'échelle  $a$ , ce dernier correspond à l'inverse de la fréquence, et en translatant d'un paramètre de translation  $b$ . Cette famille d'ondelettes est définie par la relation:

$$\Psi_{(a,b)}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right), a \in \mathbb{R}^+, b \in \mathbb{R} \quad (\text{I-3})$$

$\frac{1}{\sqrt{a}}$  est un facteur de normalisation.

#### I.4.2 Les fonctions ondelettes

Soit  $\Psi$  une fonction suffisamment régulière et bien localisée en temps et en fréquence. Pour pouvoir remplir le rôle d'ondelette, la fonction  $\Psi$  devra vérifier plusieurs conditions. Nous citons, dans ce qui suit, quelques propriétés nécessaires de la fonction  $\Psi$  [19]:

1.  $\Psi$  doit être de carré intégrable, elle doit vérifier l'inégalité [19] :

$$\int |\Psi_{(a,b)}(t)|^2 dt < \infty \quad (\text{I-4})$$

2. Elle doit vérifier la condition d'admissibilité [19]:

$$C_\Psi = \int_{-\infty}^{+\infty} \frac{|\hat{\Psi}(\omega)|^2}{\omega} d\omega < +\infty \quad (\text{I-5})$$

où  $\hat{\Psi}$  est la transformée de Fourier de  $\Psi$ .

Cette condition implique que  $\Psi$  est une fonction de moyenne nulle [19]:

$$\int_{-\infty}^{+\infty} \Psi(t) dt = 0 \quad (\text{I-6})$$

$\Psi$  est dite l'ondelette analysante.

### I.4.3 Transformée en ondelettes continue

Soit  $f(t)$  une fonction de  $L^2(\mathbb{R})$  et  $\Psi$  une ondelette analysante, la transformée en ondelettes continue de  $f(t)$  est définie par [20] :

$$C_f(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t) \bar{\Psi}_{(a,b)}(t) dt \quad (\text{I-7})$$

$\bar{\Psi}_{(a,b)}$  est le complexe conjugué de  $\Psi_{(a,b)}$ ,  $\frac{1}{\sqrt{a}}$  est un facteur de normalisation.

$a$  est le paramètre d'échelle et  $b$  représente le paramètre de translation.

Cette transformée est dite continue car les paramètres de dilatation et de translation  $(a,b)$  varient continûment dans  $(\mathbb{R}^{+*}, \mathbb{R})$  [17]. L'analyse en ondelettes offre la possibilité de suivre l'évolution d'un signal au cours du temps. De plus, cette transformation contracte et dilate l'axe des temps, agissant ainsi comme un microscope mathématique positionné en  $b$  et d'optique donnée par le choix de l'ondelette, en adaptant la taille des fenêtres d'analyse à la taille de l'objet analysé. Lorsque le facteur

d'échelle  $a$  est important, l'ondelette est très étalée. La représentation des coefficients d'ondelettes en fonction des positions  $b$  fournit alors une représentation de la forme générale de la fonction. Lorsque le facteur d'échelle est faible, ceci correspond à une représentation des singularités comme si nous regardons la fonction de près [17] .

#### **I.4.4 La transformée en ondelettes discrète :**

A l'opposé de la transformée en ondelettes continue dans laquelle l'ondelette est dilatée et translatée de manière continue, la transformée en ondelettes discrète translate et dilate l'ondelette selon des valeurs discrètes. Nous discrétisons alors la représentation continue, en échantillonnant à la fois le paramètre d'échelle  $a$  et le paramètre de translation temporelle  $b$  [21].

### **I.5 Présentation de quelques ondelettes:**

#### **I.5.1 Ondelettes continues**

Nous présentons dans ce qui suit quelques ondelettes continues. Nous nous sommes contentés de présenter les ondelettes qui sont utilisées dans notre travail.

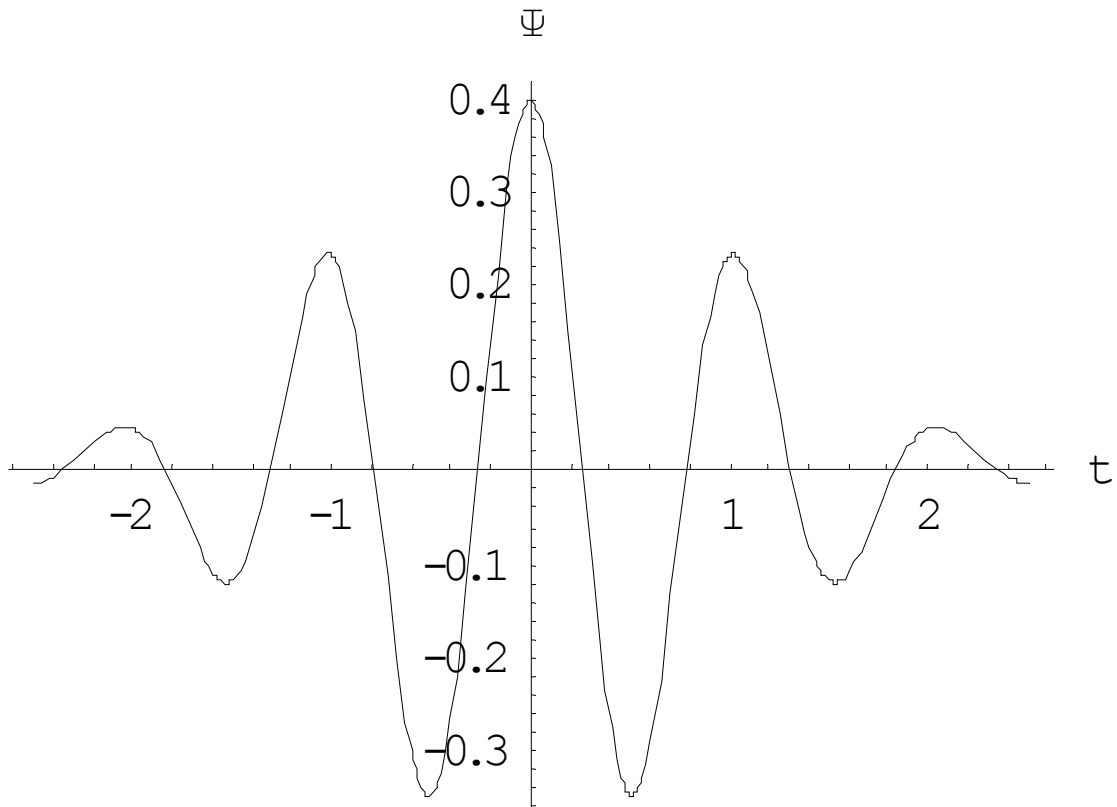
#### **L'ondelette de Morlet :**

L'ondelette de Morlet est une ondelette complexe. Son expression est

$$\Psi(t) = e^{-i\omega_0 t} e^{-t^2/2} - \sqrt{2} e^{-\omega_0^2/4} e^{-i\omega_0 t} e^{-t^2} \quad (\text{I-10})$$

Sa transformée de Fourier s'écrit :

$$\hat{\Psi}(\omega) = e^{-(\omega-\omega_0)^2/2} - e^{-\omega_0^2/4} e^{-(\omega-\omega_0)^2/4} \quad (\text{I-11})$$



**Figure I.1: L'ondelette de Morlet (partie réelle)**

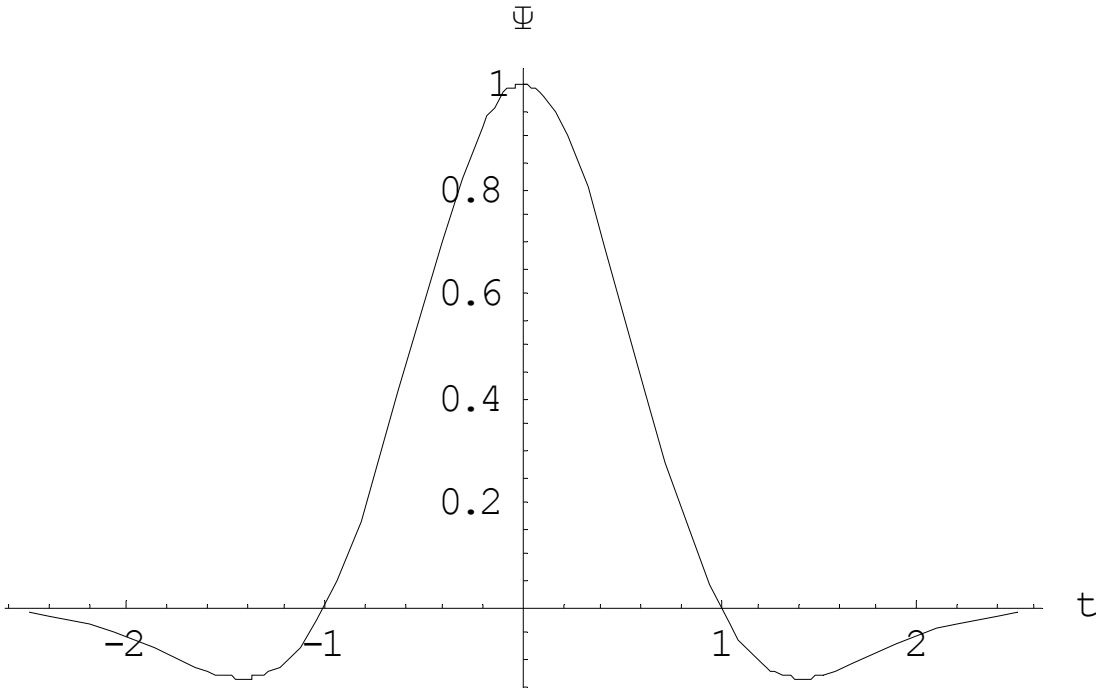
### L'ondelette chapeau mexicain

Le chapeau mexicain est une ondelette réelle construite à partir de la dérivé seconde de la fonction gaussienne, son expression est :

$$\Psi ( t ) = ( 1 - t ^ 2 ) e ^ { - t ^ 2 / 2 } \quad (\text{I-12})$$

Sa transformée de Fourier est donnée par :

$$\hat{\Psi} ( \omega ) = \omega ^ 2 e ^ { - \omega ^ 2 } \quad (\text{I-13})$$



**Figure I.2: L'ondelette chapeau mexicain (partie réelle)**

**CHAPITRE II :**  
**LES RESEAUX DE NEURONES**

## II.1 Introduction

C'est à la fin des années cinquante qu'un premier modèle d'un réseau de neurones artificiel apparaît grâce aux travaux de F. Rosenblatt qui conçoit le perceptron [6,7].

Dans l'objectif de rendre notre travail compréhensible, nous allons évoquer, dans ce qui suit, les différentes notions de base qui constituent les fondements théoriques des réseaux de neurones et de leurs apprentissages.

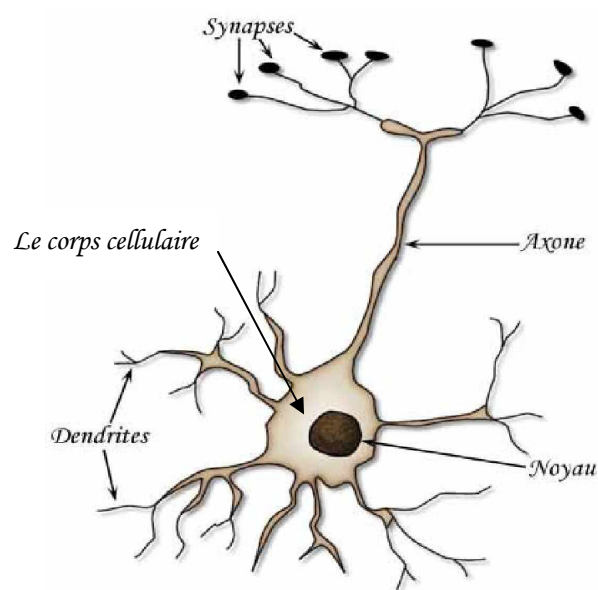
## II.2 Le neurone biologique

### II.2.1 Définition

Les cellules nerveuses, appelées neurones, sont les éléments de base du système nerveux central. Elles assurent la réception et l'intégration des signaux en provenance de neurones voisins, elles engendrent ensuite un influx nerveux, qui sera conduit et transmis à un autre neurone capable de le recevoir [4].

### II.2.2 Structure du neurone biologique

Un neurone est constitué de trois parties: le corps cellulaire, les dendrites et l'axone (voir la figure II.1) [4].



**Figure II.1** : Le neurone biologique

### **a) Le corps cellulaire**

Le corps cellulaire, aussi appelé soma, constitue le centre et l'élément principal du neurone. Ayant un diamètre de quelques microns, il contient le noyau du neurone et effectue les transformations biochimiques nécessaires à la synthèse des enzymes et d'autres molécules qui assurent la vie du neurone [4]. Il fait la somme des influx qui lui parviennent. Si cette somme dépasse un certain seuil, il envoie lui-même un influx nerveux vers d'autres neurones par l'intermédiaire de l'axone [22].

### **b) L'axone**

L'axone, qui est la fibre nerveuse, sert de moyen de transfert pour les signaux émis par le corps cellulaire aux autres neurones. Il est généralement plus long que les dendrites, il se ramifie à son extrémité, là où il communique avec d'autres neurones [4].

### **c) Les dendrites**

Chaque neurone possède une chevelure de dendrites qui se ramifient pour former une espèce d'arborescence autour du corps cellulaire, dont l'objectif est de capter les signaux qui lui parviennent des autres neurones. Les dendrites sont les récepteurs principaux du neurone. Elles sont de fines extensions tubulaires de quelques dixièmes de microns de diamètre et d'une longueur de quelques dizaines de microns [4, 22]. Pour former le système nerveux, les neurones sont connectés les uns aux autres suivant des répartitions spatiales complexes. Les connexions entre deux neurones se font en des endroits appelés synapses où ils sont séparés par un petit espace synaptique de l'ordre d'un centième de microns [4]. Donc la synapse n'est qu'une jonction entre deux neurones et généralement entre l'axone d'un neurone et une dendrite d'un autre neurone [5].

## **II.2.3 Principe de fonctionnement**

Le neurone reçoit les informations venant d'autres neurones grâce à ses dendrites, il va ensuite y avoir sommation au niveau du corps cellulaire de toutes ces informations. Si la somme des signaux dépasse un certain seuil, il renvoie un signal qui chemine le long de l'axone pour être transmis aux autres neurones jusqu'aux terminaisons synaptiques [5].

### II.3 Le neurone artificiel

#### II.3.1 Définition

Le neurone formel est une représentation mathématique et informatique du neurone biologique. Il est mathématiquement une fonction algébrique non linéaire, paramétrée, à valeurs bornées, de variables réelles appelées entrées. Il possède généralement plusieurs entrées et une sortie. Il calcule la somme pondérée des entrées reçues, puis applique à cette valeur une fonction d'activation généralement non linéaire. La valeur finale obtenue est la sortie du neurone. Les variables du neurone sont les entrées du neurone  $\{x_i\}$ , les paramètres du neurone sont les poids de connexion  $\{w_i\}$ , et la valeur de la fonction est la sortie  $Y$  du neurone [23]. Nous représentons graphiquement un neurone formel comme indiqué sur la figure II.2 qui représente le reflet de l'inspiration biologique.

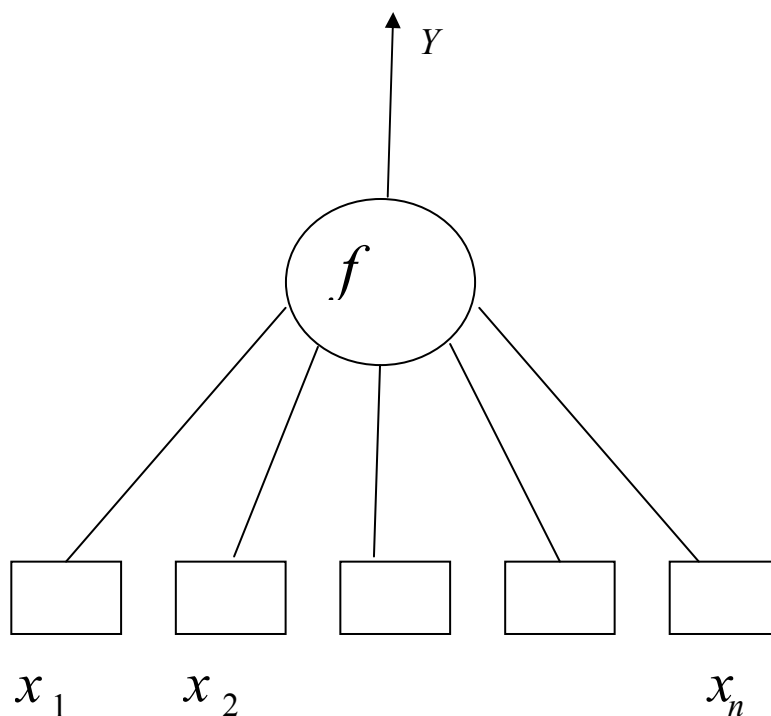


Figure II.2: Le neurone formel

### II.3.2 Structure du neurone artificiel

Le neurone artificiel est l'élément de base d'un réseau de neurones. Il reçoit un nombre variable d'entrées en provenance de neurones amont. A chacune de ces entrées est associé un poids  $w_i$  représentatif de la force de la connexion  $i$ . Chaque neurone élémentaire est doté d'une sortie unique qui se ramifie ensuite pour alimenter un nombre variable de neurones aval. A chaque connexion est associé un poids [5]. Le neurone artificiel réalise une fonction d'une sommation des entrées qui lui parviennent d'autres neurones, pondérées par les paramètres  $\{w_i\}$ , à laquelle on ajoute un terme constant  $w_0$  appelé biais [22]. Cette combinaison linéaire est appelée le potentiel d'action, elle est donnée par l'expression:

$$v = \sum_{i=1}^n w_i x_i + w_0 \quad (\text{II-1})$$

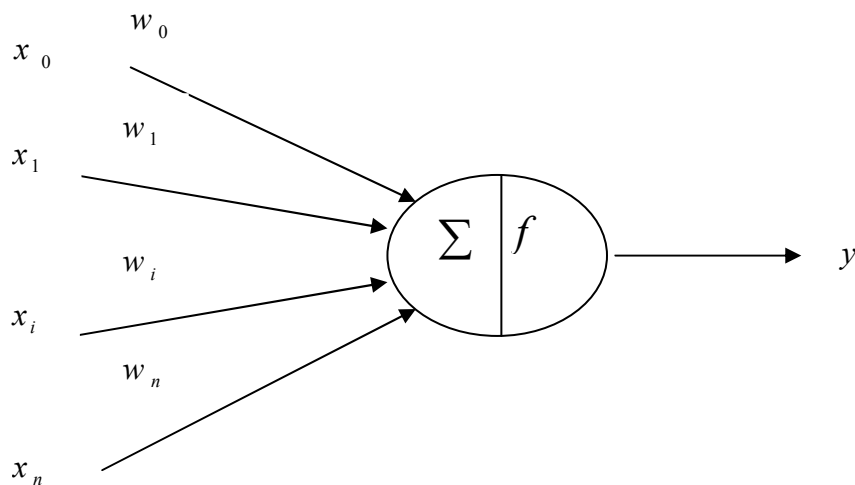
Le biais  $w_0$  est le coefficient de pondération de l'entrée 0 dont la valeur est fixée à 1.

Le neurone applique ensuite à cette sommation une fonction non linéaire  $f$ , appelée fonction d'activation, la valeur de la sortie du neurone est donc:

$$y = f(v) \quad (\text{II-2})$$

Les coefficients de pondération  $w_i$ ,  $i = 0, \dots, n$  s'appellent les poids synaptiques [22].

La structure d'un neurone artificiel est représentée par la figure suivante :



**Figure II.3:** Structure du neurone artificiel

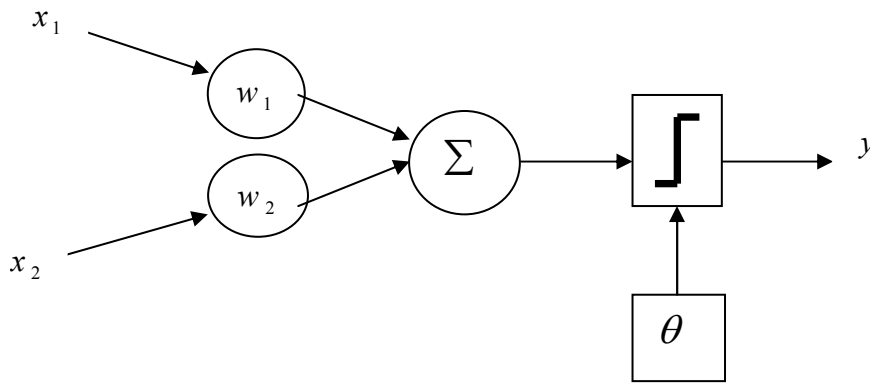
### II.3.3 Modélisation d'un neurone formel

Les réseaux de neurones formels sont à l'origine d'une tentative de modélisation mathématique du neurone biologique. Cela suppose que nous aurons une correspondance pour chaque élément composant le neurone biologique, donc une modélisation pour chacun d'entre eux. Chaque réseau de neurones est constitué d'un ensemble de neurones qui sont reliés entre eux via des synapses. Par analogie, avec son homologue biologique, le neurone formel est aussi constitué de synapses qui sont définies par les poids de connexions. Il est constitué également, d'un centre qui est l'équivalent du noyau cellulaire où se déroule les activités mathématiques telles que la sommation des entrées, ainsi que la fonction d'activation. Les entrées du neurone formel ne sont que l'équivalent des dendrites et la sortie du neurone est l'analogue de l'axone [7].

### II.4 Le modèle de Mac Culloch et Pitts

La première modélisation d'un neurone biologique a été présentée en 1943 par W. Mac Culloch et W. Pitts. Ils proposent un modèle simple de neurone formel. Il s'agit d'un neurone binaire, dont la sortie vaut 0 ou 1. Pour calculer cette sortie, le neurone effectue une somme pondérée de ses entrées qui, en tant que sorties d'autres neurones formels, valent aussi 0 ou 1. La valeur numérique de chaque entrée est modulée par le poids synaptique  $w_i$  correspondant, puis s'active suivant la valeur de la sommation pondérée.

Le neurone de W. Mac Culloch et W. Pitts applique une fonction d'activation à seuil. Si la somme pondérée par les coefficients synaptiques dépasse une certaine valeur, la sortie du neurone vaut 1, sinon elle vaut 0 [4].



**Figure II.4:** Le neurone formel de Mc Culloch et Pitts avec deux entrées et une fonction d'activation à seuil.

### II.5 Modélisation générale

D'une façon générale, nous pouvons définir un neurone formel par les cinq éléments suivants : la nature de ses entrées, la fonction de ses entrées qui définit le prétraitement effectué sur les entrées, la fonction d'activation d'état ou de transfert du neurone qui définit son état interne en fonction de son entrée totale, la fonction de sortie qui calcule la sortie du neurone en fonction de son état d'activation et la nature de la sortie du neurone [4]. La fonction d'entrée totale est, en général, linéaire, elle est donnée par l'expression suivante:

$$E = \sum_{i=1}^n w_i x_i \quad (\text{II-3})$$

$w_i$  est le poids de la  $i$ ème connexion et  $x_i$  est la  $i$ ème entrée du neurone. La sortie du neurone est définie ainsi:

$$y = f(E) \quad (\text{II-4})$$

### II.5.1 La fonction d'activation

Nous avons vu antérieurement que la fonction d'activation permet de définir l'état interne du neurone en fonction de son entrée totale en introduisant une non linéarité dans le fonctionnement du neurone [4]. Nous déduisons donc que son choix s'avère un facteur constitutif substantiel des réseaux de neurones. Assurément, les propriétés de la fonction d'activation influent sur celle du neurone formel, il est donc notable de bien désigner celle-ci afin d'obtenir un modèle bénéfique en pratique ainsi l'identité n'est pas toujours suffisante, bien au contraire, et le plus souvent des fonctions non linéaires et plus complexes seront nécessaires [24]. Il est pareillement indispensable en pratique que la fonction d'activation présente une certaine forme de régularité, afin de faciliter le calcul du gradient de l'erreur commise par un réseau de neurones, lors de son apprentissage. La fonction d'activation doit donc être dérivable. En effet, la plupart des neurones formels utilisés représentent une variante du neurone de Mac Culloch et Pitts dans lesquels la fonction de Heaviside est remplacée par une autre fonction d'activation. La fonction d'activation  $f$  peut être paramétrée de manière quelconque. Deux types de paramétrages sont fréquemment utilisés:

- La sortie du neurone est une fonction non linéaire d'une combinaison linéaire des entrées  $\{x_i\}$  pondérées par les paramètres  $\{w_i\}$ . Dans ce cas, nous optons pour une fonction d'activation en forme de "s". A titre d'exemple, nous désignons, dans ce qui suit, quelques fonctions d'activation couramment utilisées [23]:

#### a) La fonction sigmoïde

C'est une fonction croissante, continue et bornée. Elle est dérivable, la dérivée de cette fonction possède l'avantage d'être simple à calculer, son expression est [25]:

$$y = f(x) = \frac{1}{1 + \exp[-x]} \quad (\text{II-5})$$

**b) La fonction tangente hyperbolique:** c'est une version symétrique de la fonction sigmoïde, son expression est [3, 25]:

$$y = f(x) = \frac{\exp[x] - \exp[-x]}{\exp[x] + \exp[-x]} \quad (\text{II-6})$$

La sortie d'un neurone dans le cas d'une fonction d'activation tangente hyperbolique a pour équation [23, 25]:

$$y = \tanh(v) = \tanh\left(w_0 + \sum_{i=1}^{n-1} w_i x_i\right) \quad (\text{II-7})$$

où  $v$  est le potentiel d'action, et  $w_0$  est le biais tels que nous les avons définis antérieurement.

- Les paramètres sont attachés à la non linéarité du neurone, ils interviennent directement dans la fonction d'activation  $f$ , cette dernière peut être une fonction radiale ou RBF (Radial Basis Function), ou encore une ondelette. A titre d'exemple, la sortie d'un neurone RBF à non linéarité gaussienne a pour équation [23]:

$$y = \exp\left[-\frac{\sum_{i=1}^n (x_i - w_i)^2}{2w_{n+1}^2}\right] \quad (\text{II-8})$$

où les paramètres  $w_i, i = 1 \dots n$ , sont les coordonnées du centre de la gaussienne et  $w_{n+1}$ , son écart type. La différence pratique entre les deux types de neurones que nous venons d'introduire est que les neurones tels que les RBF ou les ondelettes ont des non linéarités locales, qui tendent vers zéro dans toutes les directions de l'espace des entrées, leur zone d'influence est donc limitée dans l'espace, ce qui n'est pas le cas des neurones à potentiels à fonction d'activation sigmoïde [23].

## II.6 Les réseaux de neurones artificiels

### II.6.1 Définition

Le réseau de neurones artificiel est un ensemble d'éléments très petits appelés neurones, connectés suivant une architecture bien définie afin de réaliser des tâches différentes. Un neurone formel réalise donc une somme pondérée suivie d'une non linéarité. L'intérêt des neurones réside dans les propriétés qui résultent de leur association en réseaux et des fonctions non linéaires réalisées par chaque neurone [23].

### II.6.2 Architecture des réseaux de neurones

L'architecture d'un réseau de neurones est définie par la structure de ses neurones et de ses connexions. Nous distinguons deux types d'architectures de réseaux de neurones : les réseaux de neurones non bouclés et les réseaux de neurones bouclés [23].

### II.6.3 Les réseaux de neurones non bouclés ou statiques

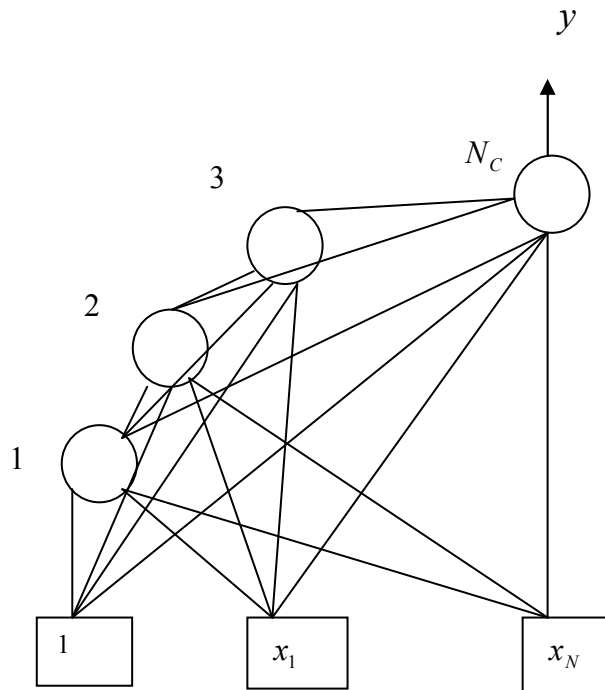
Un réseau de neurones non bouclé réalise une ou plusieurs fonctions algébriques de ses entrées, par composition des fonctions réalisées par chacun de ses neurones. Dans ce type de réseaux, l'information transite des entrées vers les sorties sans retour en arrière. Si nous représentons le réseau comme un graphe dont les nœuds sont les neurones et les arêtes sont les connexions entre ceux-ci, le graphe d'un réseau non bouclé est acyclique [23]. Dans un réseau de neurones non bouclé, le temps ne joue aucun rôle fonctionnel. Les réseaux de neurones non bouclés sont des objets statiques: si les entrées sont indépendantes du temps, les sorties le sont également. Les réseaux de neurones non bouclés sont utilisés principalement pour effectuer des tâches d'approximation de fonction non linéaire, de classification ou de modélisation de processus statiques [26].

#### II.6.3.1 Structure d'interconnexion

Les connexions entre les neurones qui composent le réseau décrivent la topologie du modèle. Elle peut être quelconque, mais le plus souvent il est possible de distinguer une certaine régularité [5]. Nous distinguons deux familles de réseaux de neurones non bouclés en fonction de la topologie des connexions entre les neurones [27].

**a) Un réseau complètement connecté:**

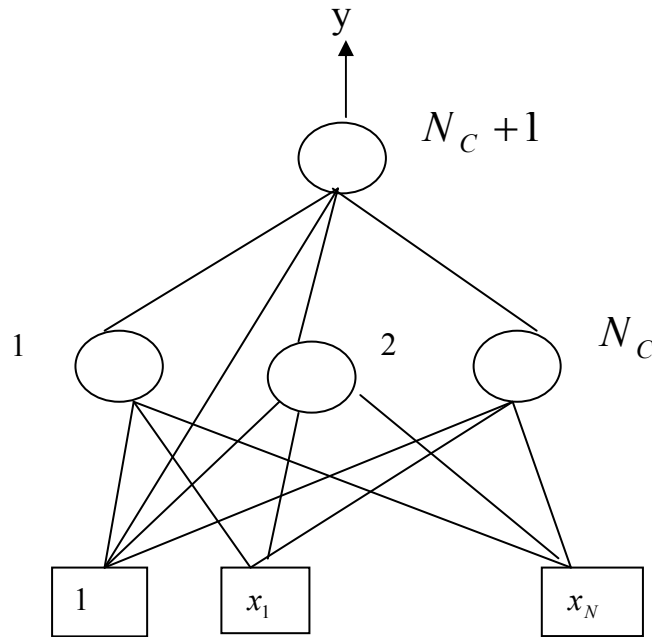
C'est la structure d'interconnexion la plus générale possible. Dans ce type de réseaux, il y a une hiérarchie parmi les neurones cachés, chacun d'entre eux est connecté aux entrées du réseau ainsi qu'aux sorties des neurones cachés précédents. Cela veut dire que chaque neurone est connecté à tous les neurones du réseau (Figure II.5) [5,27].



**Figure II.5:** Réseau de neurones non bouclé complètement connecté à  $N$  entrées et  $N_C$  neurones.

**b) Les réseaux à couches:**

Dans ce type de réseaux, les neurones sont arrangés par couches, les neurones d'une même couche n'étant pas connectés entre eux et les connexions ne se font qu'avec les neurones des couches avales. Chaque neurone est connecté à tous les neurones de la couche suivante seulement. Dans ce type de réseau, nous appelons la couche d'entrée l'ensemble des neurones d'entrée, la couche de sortie l'ensemble des neurones de sortie. Les couches intermédiaires n'ayant aucun contact avec l'extérieur sont appelées couches cachées [5,27]. Le réseau de neurones à une couche cachée et un neurone de sortie linéaire est un cas particulier de ce type de réseau (Figure II.6).



**Figure II.6:** Réseau de neurones non bouclé à N entrées et une couche de  $N_C$  neurones cachés et une sortie linéaire.

#### II.6.4 Les réseaux de neurones bouclés ou dynamiques

Contrairement aux réseaux de neurones non bouclés dont le graphe des connexions est acyclique, les réseaux bouclés peuvent avoir une topologie de connexions quelconque, comprenant notamment des boucles [26]. La sortie d'un neurone du réseau peut donc être fonction d'elle-même et cela n'est concevable que si la notion de temps est prise en considération. Ainsi à chaque connexion d'un réseau de neurones bouclé est attaché outre un poids, un retard, multiple entier de l'unité de temps choisie. Un réseau de neurones bouclé à temps discret réalise une ou plusieurs équations aux différences non linéaires, par composition des fonctions réalisées par chacun des neurones et des retards associés à chaque connexion [23]. Les réseaux de neurones bouclés sont utilisés pour effectuer des tâches de modélisation de systèmes dynamiques ou de commande de processus [26].

## **II.7 Apprentissage des réseaux de neurones**

L'apprentissage des réseaux de neurones est la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que celui-ci remplisse au mieux la tâche qui lui est affectée [23]. C'est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré [5]. Il consiste en un entraînement du réseau. Nous lui présentons des entrées et nous lui demandons de modifier progressivement, les valeurs des poids synaptiques, pour que le comportement global du réseau, se rapproche d'un comportement désiré par l'utilisation d'un algorithme d'adaptation paramétrique appelé algorithme d'apprentissage.

### **II.7.1 Les différents types d'apprentissage**

Le mécanisme d'apprentissage diffère suivant la tâche pour laquelle le réseau est utilisé. Nous distinguons deux types d'apprentissages, cette distinction repose sur la forme des exemples d'apprentissage [5].

### **II.7.2 Apprentissage supervisé**

Dans l'apprentissage supervisé des réseaux de neurones artificiels, les exemples qui sont des couples entrées, sorties désirées sont connues au début de l'apprentissage. Les poids des connexions du réseau sont déterminés durant l'ajustement en minimisant l'erreur qui est la différence entre la sortie désirée et la sortie réelle du réseau [5]. Ce type d'apprentissage est utilisé dans un contexte d'approximation de tâches et de modélisation statique et dynamique [23].

### **II.7.3 Apprentissage non supervisé**

Dans ce type de réseau, nous ne disposons que d'un ensemble de données ou des valeurs d'entrées, représentées par des vecteurs de grande dimension, que l'on cherche à regrouper selon des critères de ressemblance qui sont inconnus a priori. Ce type de réseau représente la capacité des réseaux de neurones à apprendre sans connaître la sortie désirée. Les réseaux à apprentissage non supervisé sont utilisés dans le domaine de la visualisation ou de l'analyse de données [23].

## II.8 Domaines d'applications des réseaux de neurones

Les réseaux de neurones servent aujourd'hui à toutes sortes d'applications dans divers domaines [3]. Ils sont utilisés dans des applications de reconnaissance de formes, à titre d'exemple: la lecture automatique de codes postaux [28,29], en reconnaissance de visages [30,31], détection de formes [32,33] en contrôle non destructif comme la détection de défauts dans des rails par courant de Foucault [34], en fouille de données comme le filtrage de documents [35] et en robotique à l'instar du pilotage d'un véhicule autonome [36,37]. Deux grandes familles d'applications se distinguent néanmoins: l'approximation de fonctions de régression et la classification [38].

Un réseau de neurones non bouclé réalise une relation algébrique entre ses entrées et ses sorties, nous pouvons donc affecter à un tel réseau la tâche qui consiste à réaliser une fonction algébrique non linéaire, celle-ci peut être connue analytiquement, le réseau réalise une tâche d'approximation de fonctions, ou inconnue analytiquement mais nous disposons ici d'un ensemble de mesures de variables d'un processus de nature quelconque physique, chimique, économique, et du résultat de ce processus, et nous cherchons un modèle mathématique du processus qui décrit l'évolution de ses sorties en fonction de ses entrées. Le réseau réalise alors une tâche de modélisation statique ou une régression [23]. Les premières applications industrielles des réseaux de neurones à apprentissage supervisé étaient relatives à la reconnaissance de formes ou de signaux. Ceci est dû au fait que tout problème de classification peut se ramener à un problème très voisin de celui de la régression non linéaire. En effet, classer un ensemble d'objets revient, en d'autres termes, à attribuer à chacun une classe ou catégorie parmi plusieurs classes définies à l'avance. La classification de données d'un système revient donc à discriminer les objets d'une classe par rapport aux autres classes par l'intermédiaire de mesures accessibles du système, ou encore à approcher différentes fonctions discriminatrices de telle sorte que les ensembles de sorties associés à un jeu de valeurs d'entrées d'une classe donnée soient disjoints les uns des autres [23, 38].

**CHAPITRE III :**

**ADAPTATION DES RESEAUX**

**DE NEURONES-ONDELETTES**

**A L'APPROXIMATION DE**

**TACHES**

### **III.1 Introduction**

Les réseaux de neurones sont devenus en quelques années des outils précieux dans des domaines très divers [26]. Ils possèdent une propriété remarquable, ce sont des approximateurs universels parcimonieux [8]. La propriété d'approximateurs universels n'est pas spécifique aux réseaux de neurones uniquement, il existe bien d'autres familles de fonctions paramétrées possédant cette propriété ; c'est le cas notamment des ondelettes [8]. Dans ce chapitre, nous allons adapter la technique des réseaux de neurones-ondelettes pour l'approximation de diverses tâches. Cette technique permet d'établir un lien entre les méthodes neuronales et la théorie de la décomposition en ondelettes, issue des techniques du traitement du signal [8]. Ce modèle qui utilise les ondelettes comme fonctions d'activation peut être considérée comme une alternative aux réseaux de neurones classiques (les perceptrons multicouches), où la fonction d'activation utilisée est la fonction sigmoïde, et aux réseaux à fonctions radiales où la fonction d'activation choisi est une gaussienne.

### **III.2 Propriété fondamentale des réseaux de neurones formels: l'approximation parcimonieuse.**

La propriété fondamentale des réseaux de neurones est l'approximation parcimonieuse c'est à dire que le modèle fait intervenir le plus petit nombre possible de paramètres ajustables. Cette expression révèle deux propriétés spécifiques, d'une part les réseaux de neurones sont des approximateurs universels et d'autre part, ils sont des approximateurs parcimonieux et c'est cette deuxième propriété qui leur est particulière et fait tout leur intérêt [23].

#### **III.2.1 L'approximation universelle**

La propriété d'approximation universelle peut s'énoncer de la façon suivante:

Toute fonction bornée suffisamment régulière peut être approchée uniformément, avec

L'APPROXIMATION DE TACHES

une précision arbitraire, dans un domaine fini de l'espace de ses variables, par un réseau de neurones comportant une couche de neurones cachés en nombre fini, possédant, tous la même fonction d'activation [39,40]. C'est cette propriété qui justifie notre choix de l'architecture de réseaux de neurones non bouclé à une couche cachée et à un neurone de sortie linéaire.

### **III.2.2 La parcimonie**

La spécificité des réseaux de neurones réside dans le caractère parcimonieux de l'approximation à précision égale. Si l'approximation dépend des paramètres ajustables de manière non linéaire, elle est plus parcimonieuse que si elle dépend linéairement des paramètres. Les réseaux de neurones nécessitent donc moins de paramètres ajustables ou de poids de connexions que les autres approximateurs, plus précisément le nombre de poids varie linéairement avec le nombre de variables de la fonction à approcher, alors qu'il varie exponentiellement pour la plupart des autres approximateurs [41]. La parcimonie est donc d'autant plus importante que le nombre d'entrées du modèle est grand [23].

### **III.3 Mise en œuvre des réseaux de neurones**

La mise en œuvre des réseaux de neurones comporte à la fois, une partie conception dont l'objectif est de permettre de choisir l'architecture du réseau adaptée à la tâche pour laquelle ce réseau est choisi, et une partie calcul numérique qui consiste essentiellement à réaliser l'apprentissage du réseau de neurones. Pour le premier volet de cette mise en œuvre, nous y reviendrons dans la partie III-5 de ce chapitre. Le deuxième aspect de cette mise en œuvre des réseaux de neurones est le phénomène d'apprentissage. En effet, à partir d'une architecture bien déterminée et d'exemples disponibles, l'apprentissage supervisé des réseaux de neurones formels, consiste à calculer les coefficients optimaux qui représentent les poids synaptiques de sortie, de telle manière que les sorties du réseau de neurones soient aussi proches que possibles des sorties désirées. La plupart des algorithmes d'apprentissage des réseaux de neurones formels sont des algorithmes d'optimisation, ils cherchent à minimiser, par

L'APPROXIMATION DE TACHES

des méthodes d'optimisation non linéaires, une fonction de coût ou l'erreur qui constitue une mesure de l'écart entre les réponses réelles du réseau et ses réponses désirées. Cette optimisation se fait de manière itérative, en modifiant les poids en fonction du gradient de la fonction de coût. Ce gradient est estimé par une méthode spécifique aux réseaux de neurones, dite méthode de rétropropagation. Les poids sont initialisés aléatoirement avant l'apprentissage, puis modifiés itérativement jusqu'à obtention de la sortie désirée. Chaque itération du processus d'apprentissage nécessite donc la mise en œuvre de deux volets distincts :

- L'évaluation du gradient de la fonction de coût.
- La modification des paramètres en fonction de ce gradient, afin d'approcher un minimum de la fonction de coût [23].

**III.3.1 La fonction de coût**

La définition de la fonction de coût est fondamentale, car elle définit l'écart entre la sortie calculée par le réseau de neurones et la sortie désirée. La fonction de coût la plus utilisée est celle des moindres carrés. Elle est donnée par l'expression suivante, si nous disposons d'un ensemble d'apprentissage comprenant  $n$  exemples [4]:

$$E(\vec{w}) = \sum_{k=1}^n \left[ y(\vec{x}^k) - g(\vec{x}^k, \vec{w}) \right]^2 \quad \text{(III-1)}$$

$(\vec{x}^k)$  est le vecteur des variables d'entrées,  $y(\vec{x}^k)$  est la valeur de la sortie désirée lorsque le vecteur d'entrées vaut  $(\vec{x}^k)$ ,  $g(\vec{x}^k, \vec{w})$  est la valeur de la sortie calculée par le réseau de neurones et  $\vec{w}$  est le vecteur des poids du réseau de neurones.

**III.3.2 Evaluation du gradient de la fonction de coût par rétropropagation**

L'évaluation du gradient de la fonction de coût pour un réseau de neurones peut être effectuée d'une façon économique à l'aide d'un algorithme appelé algorithme de rétropropagation [42,43]. Mathématiquement, cet algorithme utilise les règles de dérivation composées.

L'APPROXIMATION DE TACHES

Dans cet algorithme, de même que nous sommes capables de propager un signal provenant des cellules d'entrées vers la couche de sortie, nous pouvons, en suivant le chemin inverse, rétropropager l'erreur commise en sortie vers les couches internes cela s'explique pratiquement ainsi:

A chaque itération, un exemple est présenté en entrée du réseau. Une sortie réelle est calculée. Ce calcul est effectué de la couche d'entrée à la couche de sortie. Cette phase est appelée propagation avant. Ensuite l'erreur est calculée puis elle est rétropropagée dans le réseau, dans le sens sortie entrée, donnant lieu à une modification des poids [4].

**III.2.1 L'algorithme de la rétropropagation**

Nous considérons dans cette partie, un réseau de neurones non bouclé. Nous rappelons que le neurone  $i$  calcule une grandeur  $y_i$  qui est une fonction non linéaire de son potentiel  $v_i$  donné par l'expression [23]:

$$v_i = \sum_{j=1}^{n_i} w_{ij} x_j \quad (\text{III-2})$$

$n_i$  : Nombres d'entrées du neurone  $i$ .

La sortie du neurone  $i$  est donc donnée par l'expression:

$$y_i = f \left( \sum_{j=1}^{n_i} w_{ij} x_j \right) = f (v_i) \quad (\text{III-3})$$

Nous savons que les  $n_i$  entrées du neurone  $i$  peuvent être soit les sorties d'autres neurones, soit les entrées du réseau. Donc  $x_j$  désignera indifféremment soit la sortie  $y_j$  du neurone  $j$ , soit l'entrée  $j$  du réseau. La fonction de coût dont nous cherchons à évaluer le gradient est donnée par l'expression (III-1):

L'APPROXIMATION DE TACHES

$$E(\vec{w}) = \sum_{k=1}^N \left[ y(\vec{x}^k) - g(\vec{x}^k, \vec{w}) \right]^2 = \sum_{k=1}^N E^k(\vec{w}) \quad (\text{III-4})$$

Pour évaluer son gradient, il suffit donc d'évaluer le gradient du coût partiel  $E^k(\vec{w})$  relatif à l'exemple k et de faire ensuite la somme sur tous les exemples. Nous avons vu précédemment que l'algorithme de rétropropagation consiste essentiellement en l'application répétée de la règle des dérivées composées. Nous remarquons aussi que la fonction de coût partiel ne dépend du paramètre  $w_{ij}$  que par l'intermédiaire de la valeur de la sortie du neurone  $i$ , qui est elle même fonction uniquement du potentiel du neurone  $i$ , nous écrivons donc [4,23]:

$$\left( \frac{\partial E^k}{\partial w_{ij}} \right)_k = \left( \frac{\partial E^k}{\partial v_i} \right)_k \left( \frac{\partial v_i}{\partial w_{ij}} \right)_k = \delta_i^k x_j^k \quad (\text{III-5})$$

Evaluons à présent les quantités  $\delta_i^k$  pour un neurone de sortie  $i$  et pour un neurone caché  $i$ .

a) Pour le neurone de sortie  $i$ :

$$\delta_i^k = \left( \frac{\partial E^k}{\partial v_i} \right)_k = \left( \frac{\partial}{\partial v_i} \left[ \left( y^k(\vec{x}^k) - g(\vec{x}^k, \vec{w}) \right)^2 \right] \right)_k = -2 \left[ y^k - g(\vec{x}^k, \vec{w}) \right] \left( \frac{\partial g(\vec{x}^k, \vec{w})}{\partial v_i} \right)_k \quad (\text{III-6})$$

or la sortie du réseau est la sortie  $y_i$  du neurone de sortie donné par l'expression

$$y_i = f(v_i) \quad (\text{III-7})$$

donc:

$$\delta_i^k = -2 \left[ y^k - g(\vec{x}^k, \vec{w}) \right] \left( \frac{\partial f}{\partial v_i} \right)_k = -2 \left[ y^k - g(\vec{x}^k, \vec{w}) \right] f'(v_i^k) \quad (\text{III-8})$$

L'APPROXIMATION DE TACHES

b) Pour un neurone caché  $i$ , la fonction de coût ne dépend du potentiel du neurone  $i$  que par l'intermédiaire des potentiels des neurones  $m$  qui reçoivent la valeur de la sortie du neurone  $i$ , ou à tous les neurones qui, dans le graphe des connexions du réseau, sont adjacents au neurone  $i$ , entre ce neurone et la sortie [4,23]:

$$\delta_i^k \equiv \left( \frac{\partial E^k}{\partial v_i} \right)_k = \sum_m \left( \frac{\partial E^k}{\partial v_m} \right)_k \left( \frac{\partial v_m}{\partial v_i} \right)_k = \sum_m \delta_m^k \left( \frac{\partial v_m}{\partial v_i} \right)_k \quad (\text{III-9})$$

or :

$$v_m^k = \sum_i w_{mi} x_i^k = \sum_i w_{mi} f(v_i^k) \quad (\text{III-10})$$

d'où:

$$\left( \frac{\partial v_m}{\partial v_i} \right)_k = w_{mi} f'(v_i^k) \quad (\text{III-11})$$

Finalement nous obtenons la relation:

$$\delta_i^k = \sum_m \delta_m^k w_{mi} f'(v_i^k) = f'(v_i^k) \sum_m \delta_m^k w_{mi} \quad (\text{III-12})$$

Ainsi les quantités  $\delta_i^k$  peuvent être calculées récursivement, en parcourant le graphe des connexions à l'envers, depuis la sortie vers les entrées du réseau ce qui explique le terme de rétropropagation [4, 23].

### III. 3.3 Modification des paramètres en fonction du gradient de la fonction de coût

Nous avons vu antérieurement, que nous évaluons le gradient de la fonction de coût par rapport aux paramètres du réseau, à chaque itération du processus d'apprentissage. Cette étape est suivie par une modification des poids, afin d'approcher un minimum de la fonction de coût. Nous allons présenter, dans ce qui suit, un algorithme de minimisation itérative des paramètres du réseau utilisé dans notre travail.

#### III.3.4 La méthode du gradient simple

La méthode du gradient simple consiste à modifier les paramètres par la formule suivante, à l'itération  $k$  de l'apprentissage:

$$\vec{w}_{k+1} = \vec{w}_k - \mu_k \vec{\nabla} E(\vec{w}_k) \quad (\text{III-13})$$

$\mu_k$  est appelé le pas du gradient ou le pas d'apprentissage et  $\vec{\nabla} E(\vec{w}_k)$  est le vecteur gradient de la fonction de coût, vecteur des dérivées premières de  $E(\vec{w}_k)$  par rapport à  $\vec{w}_k$  défini ainsi [8]:

$$\vec{\nabla} E = \begin{bmatrix} \frac{\partial E}{\partial w_1} \\ \cdot \\ \cdot \\ \frac{\partial E}{\partial w_m} \end{bmatrix} \quad (\text{III-14})$$

Numériquement, nous partons d'une initialisation aléatoire des poids et nous poursuivons le déroulement de l'algorithme jusqu'à satisfaction d'un critère d'arrêt. Un réglage du pas du gradient  $\mu_k$  est important car si le pas du gradient est trop petit, la décroissance du coût est très lente, en revanche une grande valeur entraîne une

L'APPROXIMATION DE TACHES

oscillation autour de la solution. Cette méthode est fiable si la valeur de départ est plus proche du minimum global que d'un minimum local, si la valeur de départ est mal choisie, nous obtenons dans ce cas un minimum local au lieu d'un minimum global, pour pallier cet inconvénient nous recourons à [4,23]:

- Une modification des poids synaptiques initiaux
- Une augmentation du nombre de couches cachées ainsi que le nombre de neurones dans ces couches.

### III.4 Théorème de Kolmogorov

Le théorème de Kolmogorov s'énonce ainsi:

Toute fonction continue sur le cube unité à  $n$  dimensions  $E^n$ ,  $E = [0,1]$  s'écrit sous la forme suivante [44]:

$$f(x_1, x_2, \dots, x_n) = \sum_{q=1}^{2n+1} h_q \left( \sum_{p=1}^n \phi_{pq}(x_p) \right) \quad \text{(III-15)}$$

où  $h_q$  et  $\phi_{pq}$  sont des fonctions continues, réelles d'une seule variable.

En 1987, Hecht Nielson [45] a montré une ressemblance entre la structure du développement des fonctions continues de Kolmogorov et d'autres fonctions élémentaires avec un réseau de neurones non bouclé ayant trois couches. Il a reformulé le théorème de Kolmogorov ainsi:

Toute fonction continue définie sur le cube de  $n$  dimensions  $E^n$ ,  $E = [0,1]$ , peut être implémentée exactement par un réseau de neurones ayant trois couches et  $(2n+1)$  unités neuronales dans la couche cachée avec des fonctions de transfert sigmoïdales des neurones de la couche cachée,  $n$  étant le nombre d'entrées du réseau [45].

L'APPROXIMATION DE TACHES

**III.5 Architecture du réseau utilisée**

L'objectif de ce mémoire est d'approcher une tâche bien définie, en utilisant l'approche des réseaux de neurones- ondelettes. La première partie à réaliser pour accomplir cette tâche est la conception du réseau, ce qui revient essentiellement à un choix judicieux de l'architecture du réseau de neurones utilisée adaptée à la tâche pour laquelle ce réseau est choisi. Notre tâche est celle de l'approximation de fonctions dont les expressions analytiques sont connues, cela revient intuitivement à opter pour une architecture d'un réseau de neurones non bouclé à apprentissage supervisé, à une couche cachée avec un nombre de neurones précis et un neurone de sortie linéaire. Tel que énoncé dans la partie (III.2.1) de ce chapitre. En tenant compte du théorème de Kolmogorov et de la propriété de l'approximation universelle, nous allons adopter donc, tout au long de ce travail une architecture simplifiée d'un réseau ayant trois couches, une entrée  $x$ , une couche cachée avec un nombre fini de fonctions élémentaires  $f(x)$ , et un neurone de sortie linéaire. L'approche des fonctions ondelettes consiste à développer la tâche à approcher sur la base des fonctions translatées d'un paramètre translationnel  $b$  et dilatées d'un paramètre échelle  $\lambda$ . Ces fonctions sont données par l'expression:

$$f_{(b,\lambda)}(x) = f\left(\frac{x-b}{\lambda}\right) \quad (\text{III-16})$$

Ce développement est donné par [46]:

$$F(x) = \int w(b,\lambda) f\left[\frac{(x-b)}{\lambda}\right] db d\lambda \quad (\text{III-17})$$

où  $b$  est le paramètre de translation et  $\lambda$  est le paramètre d'échelle. Les coefficients  $w(b,\lambda)$  sont les poids synaptiques de sortie. Nous tenons compte, dans ce qui suit, uniquement du paramètre d'échelle  $\lambda$  et nous omettons le paramètre translationnel, de ce fait, l'intégrale (III-17) devient [47]:

L'APPROXIMATION DE TACHES

$$F(x) = \int_0^{\infty} w(\lambda) f(x/\lambda) d\lambda \quad (III-18)$$

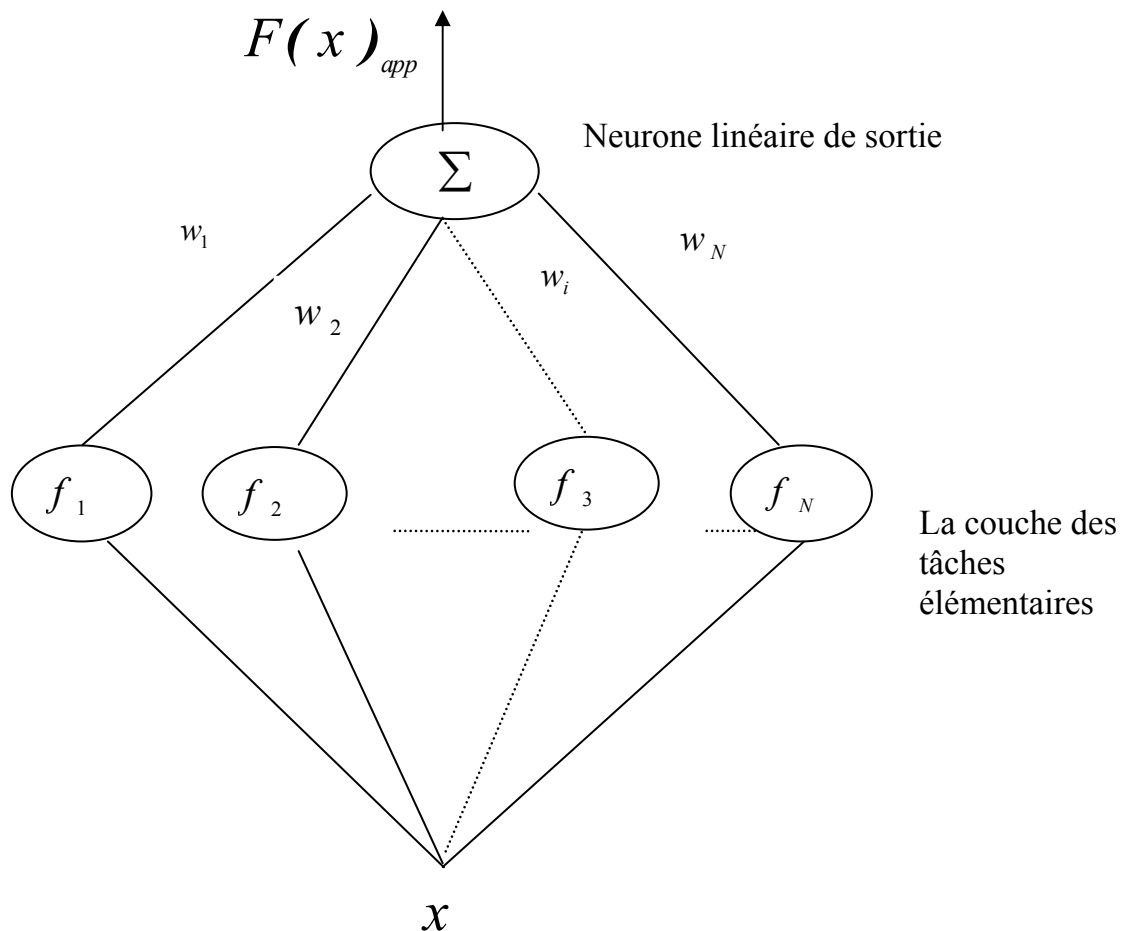
En discrétisant l'équation (III-18) en N unités ou neurones, nous obtenons la fonction approchée suivante:

$$F_{app}(x) = \sum_{i=1}^N w(\lambda_i) f(x/\lambda_i) \quad (III-19)$$

Nous adoptons ainsi une notation abrégée donnée par l'expression:

$$F_{app} = \sum_i w_i f_i \quad (III-20)$$

Cette architecture célèbre d'un réseau d'ondelettes est représentée par le schéma suivant, nous prenons un schéma de réseau simplifié où les neurones formels  $f_i$  reçoivent la même entrée  $x$ . La sortie  $F(x)$  est donnée par une combinaison linéaire des  $f_i$  pondérée par les poids synaptiques  $w_i$ .



**Figure III.1: Architecture d'un réseau d'ondelettes**

### III-6 L'apprentissage du réseau

Pour trouver la meilleure approximation de  $F_{app}$ , nous effectuons l'apprentissage par rapport à deux types de paramètres, les poids synaptique  $w_i$  et les échelles  $\lambda_i$  cachées à l'intérieur des tâches élémentaires. La première opération d'apprentissage consiste à optimiser les poids synaptiques de sortie en connexion avec la couche intermédiaire des tâches élémentaires. Le second apprentissage consiste à optimiser les paramètres d'échelle de cette couche intermédiaire de neurones. Cette procédure d'apprentissage se concrétise réellement par la minimisation de la norme du carré de l'erreur commise, d'abord par rapport aux poids synaptiques  $w_i$ , ensuite par rapport aux paramètres d'échelles  $\lambda_i$  [47]. L'erreur est donnée par l'expression :

$$\mathcal{E} = \|F - F_{app}\| \quad (III-21)$$

$F$  est la valeur réelle de la tâche et  $F_{app}$  est sa valeur approchée.

#### 6-1 Apprentissage des poids synaptiques de sortie $w_i$

La première minimisation par rapport aux poids synaptiques  $w_i$  se traduit par l'équation suivante [47]:

$$\frac{\partial}{\partial w_i} \left( \langle F - F_{app} | F - F_{app} \rangle \right) = 0 \quad (III-22)$$

En substituant le développement donné par l'équation (III-20) dans l'expression (III-22), nous obtenons :

$$\frac{\partial}{\partial w_i} \left\langle F - F_{app} \left| F - F_{app} \right. \right\rangle = \frac{\partial}{\partial w_i} \left\langle F - \sum_{j=1}^N w_j f_j \left| F - \sum_{k=1}^N w_k f_k \right. \right\rangle$$

$$\begin{aligned}
 &= \frac{\partial}{\partial w_i} \left\{ \langle F|F \rangle - \sum_{k=1}^N w_k \langle F|f_k \rangle + \sum_{k,j=1}^N w_j \langle f_j|f_k \rangle w_k - \sum_{j=1}^N w_j \langle f_j|F \rangle \right\} \\
 &= \frac{\partial}{\partial w_i} \left\{ \langle F|F \rangle - 2 \sum_{j=1}^N w_j \langle f_j|F \rangle + \sum_{j,k=1}^N w_j \langle f_j|f_k \rangle w_k \right\} = 0, \quad i = 1, \dots, N
 \end{aligned}
 \tag{III-23}$$

Nous dérivons par rapport à  $w_i$  nous obtenons ceci:

$$\begin{aligned}
 &= \frac{\partial}{\partial w_i} \left\{ \langle F|F \rangle - 2 \sum_{j=1}^N w_j \langle f_j|F \rangle + \sum_{j,k=1}^N w_j \langle f_j|f_k \rangle w_k \right\} = 0 \\
 &-2 \frac{\partial}{\partial w_i} \sum_{j=1}^N w_j \langle f_j|F \rangle + \frac{\partial}{\partial w_i} \sum_{j,k=1}^N w_j \langle f_j|f_k \rangle w_k = 0 \\
 &-2 \langle f_i|F \rangle + \sum_{j=1}^N \langle f_j|f_i \rangle w_j + \sum_{k=1}^N \langle f_i|f_k \rangle w_k = 0 \\
 &-2 \langle f_i|F \rangle + 2 \sum_{j=1}^N \langle f_i|f_j \rangle w_j = 0
 \end{aligned}$$

d'où

$$\sum_{j=1}^N \langle f_i|f_j \rangle w_j = \langle f_i|F \rangle
 \tag{III-24}$$

Dont la solution est donnée par :

$$w_i = \sum_{j=1}^N (g^{-1})_{ij} \langle f_j | F \rangle, \quad i = 1, \dots, N \quad (\text{III-25})$$

où  $g$  représente la matrice dont les éléments sont donnés par:

$$g_{jk} = \langle f_j | f_k \rangle \quad (\text{III-26})$$

### III.6.2 Apprentissage des tâches élémentaires (des paramètres d'échelle $\lambda_i$ ) :

Cette étape consiste à minimiser le carré de la norme de l'erreur par rapport aux paramètres d'échelle  $\lambda$ . Cette minimisation est donnée par l'expression suivante [47]:

$$\frac{\partial}{\partial \lambda_i} \left( \langle F - F_{app} | F - F_{app} \rangle \right) = 0 \quad (\text{III-27})$$

$$\frac{\partial \varepsilon}{\partial \lambda_i} = \frac{\partial}{\partial \lambda_i} \langle F - F_{app} | F - F_{app} \rangle = 0$$

$$\frac{\partial \varepsilon}{\partial \lambda_i} = \frac{\partial}{\partial \lambda_i} \left\langle F - \sum_{i=1}^N w(\lambda_i) f(x/\lambda_i) \middle| F - F_{app} \right\rangle = 0$$

$$\frac{\partial \varepsilon}{\partial \lambda_i} = \frac{-2w_j}{\lambda_i^2} \langle x f'(x/\lambda_i) | F - F_{app} \rangle = 0, \quad i = 1, \dots, N \quad (\text{III-28})$$

où  $f'$  est la dérivée de la tâche élémentaire avant toute translation et dilatation [47].

### III.7 Résultats numériques et discussions

Nous allons exposer, dans ce qui suit, nos résultats numériques dans le cadre de l'approche des réseaux de neurones ondelettes. Considérons la première tâche cible à approcher dont l'expression analytique est [47]:

$$F(x) = 0.10167 \exp(-x/5) \{0.60717 \tanh[3(x-1.66133)] - 4.33575 \tanh[3(x-9.56591)]\} \quad (\text{III-29})$$

Dans la recherche de la meilleure approximation de  $F$ , nous avons considéré un schéma simplifié d'un réseau de neurones avec un nombre de neurones cachés  $N=5$ . Nous débutons le processus d'apprentissage par la minimisation de la norme de l'erreur quadratique moyenne MSE relative aux poids synaptiques  $w_i$ . L'apprentissage a été initialisé par les valeurs 1/4, 1/2, 1,2 et 4 pour les  $\lambda_i$ .

#### III.7.1 Etude de l'Influence de la fonction d'activation

La première fonction d'activation utilisée est la fonction sigmoïde donnée par l'expression (II-5) :

$$f_{sig}(x) = \frac{1}{1 + \exp(-x)}$$

Un second exemple classique de fonction d'activation choisie est une lorentzienne dont l'expression est:

$$f_{loren}(x) = \frac{1}{1 + x^2} \quad (\text{III-30})$$

La troisième fonction d'activation choisie est la gaussienne définie ainsi:

L'APPROXIMATION DE TACHES

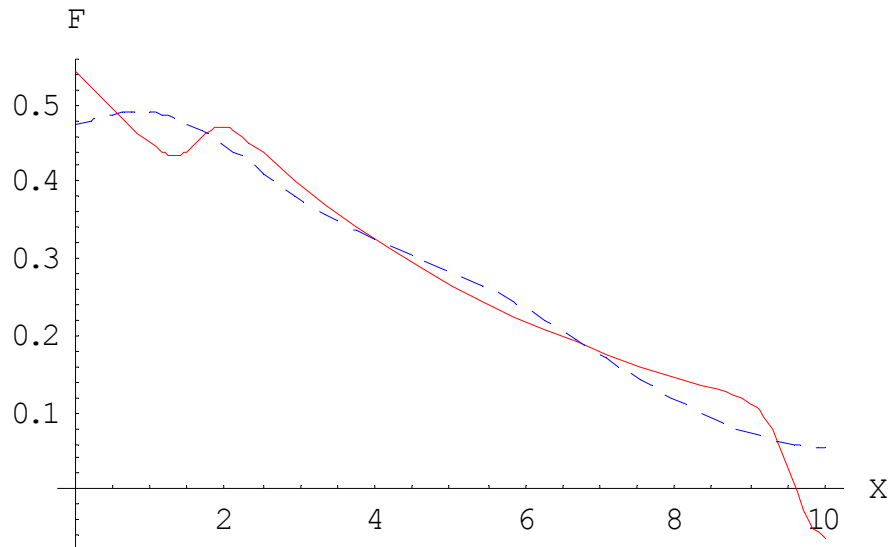
$$f_{gau}(x) = \frac{1}{\sqrt{2}} \exp\left(\frac{-x^2}{2}\right) \quad (\text{III-31})$$

Nous avons exécuté par la suite notre code de calcul en employant l'ondelette Chapeau Mexicain comme fonction d'activation donnée par l'expression:

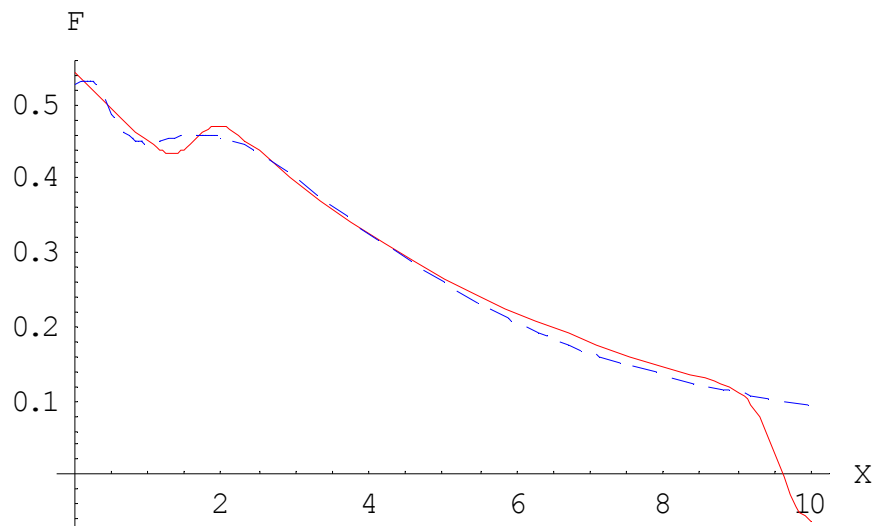
$$f(x) = (1 - x^2) \exp\left(\frac{-x^2}{2}\right) \quad (\text{III-32})$$

Après environ 80 pas, le réseau converge et une saturation de  $\|F_{app}\|^2$  débute où un processus de comparaison peut être effectué entre  $F$  et  $F_{app}$ . La figure (III.2), représente la tâche à approcher et son approximation dans le cas où la fonction d'activation choisie est une sigmoïde. Une approximation qui semble plus au moins satisfaisante: on note un léger écart entre les deux courbes, cet écart se réduit et l'approximation devient meilleure dans l'intervalle [6,7]. Une approximation plus appréciable découle de l'usage d'une lorentzienne comme fonction d'activation, où l'écart entre les deux courbes est ostensiblement léger comme le désigne la figure (III.3). Un autre résultat appréciable émane de l'utilisation de la gaussienne comme fonction d'activation comme le montre la figure (III.4), où le rapprochement des deux courbes est évident. Cette conséquence est notamment dû à l'aspect gaussien qui semble mieux convenir que la sigmoïde dans l'approximation de la tâche étudiée. Comme nous l'avons indiqué dans notre introduction principale, des travaux antérieurs ont montré que les fonctions ondelettes sont des approximateurs universels [8]. Nous les avons alors utilisées comme fonctions d'activation. La figure (III.5) fait justement apparaître le résultat acquis dans le cas de l'ondelette chapeau mexicain. Une première analyse de cette dernière figure montre une nette amélioration de l'approximation. Ce perfectionnement de l'approximation est notable en le comparant avec celui obtenu dans le cas de la sigmoïde ainsi que pour la lorentzienne et la gaussienne.

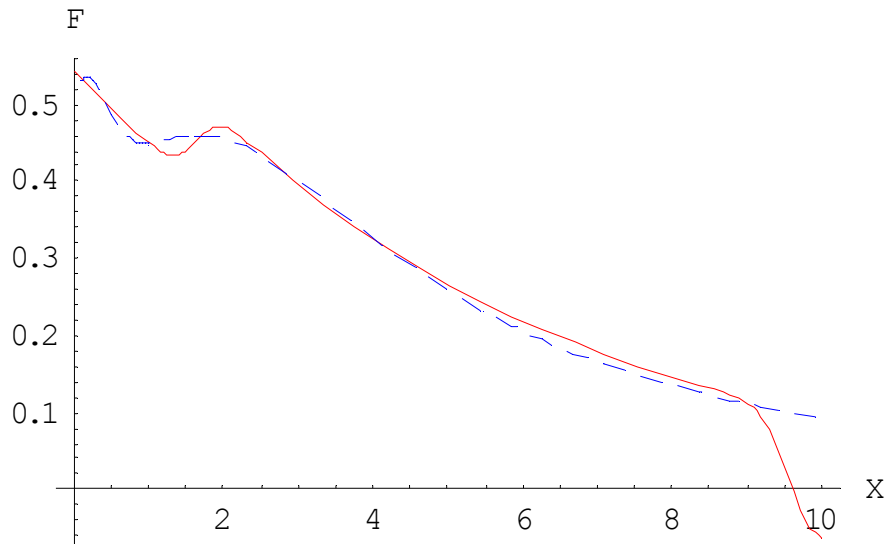
L'APPROXIMATION DE TACHES



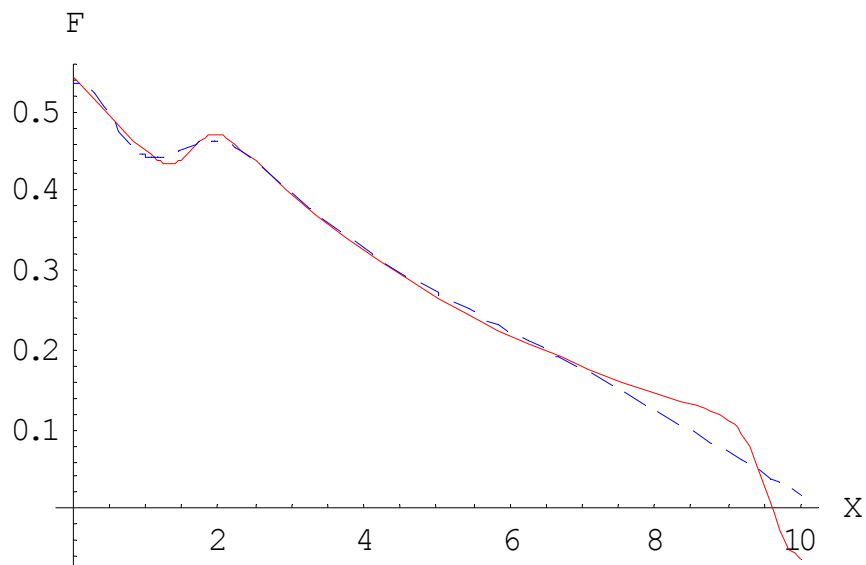
**Figure III.2:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de la sigmoïde.



**Figure III.3:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas d'une lorentzienne.



**Figure III.4:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas d'une gaussienne.



**Figure III.5:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain.

Pour mieux quantifier la progression de l'amélioration de notre approximation en fonction des différentes fonctions d'activation, nous avons choisi un autre critère de comparaison qui est l'évaluation de l'erreur quadratique moyenne MSE qui désigne l'écart moyen de l'approximation par rapport à la tâche à approcher. Pour un nombre de neurones cachés  $N=5$  et un nombre d'itérations fixé à 71, le réseau converge et une saturation de  $\|F_{app}\|^2$  débute où un processus de comparaison peut être effectué entre  $F$  et  $F_{app}$ . Nous avons récapitulé les différentes valeurs du MSE, obtenues via nos codes de calculs, relatives à chaque fonction d'activation, dans le tableau 1 :

<b>Fonction d'activation</b>	<b>MSE</b>
Sigmoïde	0.00970661
Lorentzienne	0.00935646
Gaussienne	0.0067184
Ondelette chapeau mexicain	0.00377949

**Tableau 1:** Les valeurs du MSE relatives à chaque fonction d'activation

Il est utile de signaler que des calculs du MSE ont été effectués à l'aide de l'ondelette de Morlet, comme les résultats obtenus ne sont pas très significatifs, nous avons préféré ne pas les afficher dans le tableau 1. Il apparaît clairement, en analysant les résultats affichés ci-dessus que la plus petite valeur du MSE est obtenue dans le cas de l'ondelette chapeau mexicain. Nous concluons que l'ondelette chapeau mexicain est la fonction d'activation la plus adaptée pour la meilleure approximation de cette tâche. Ce résultat affirme le rôle primordial de la fonction d'activation dans l'approche des

L'APPROXIMATION DE TACHES

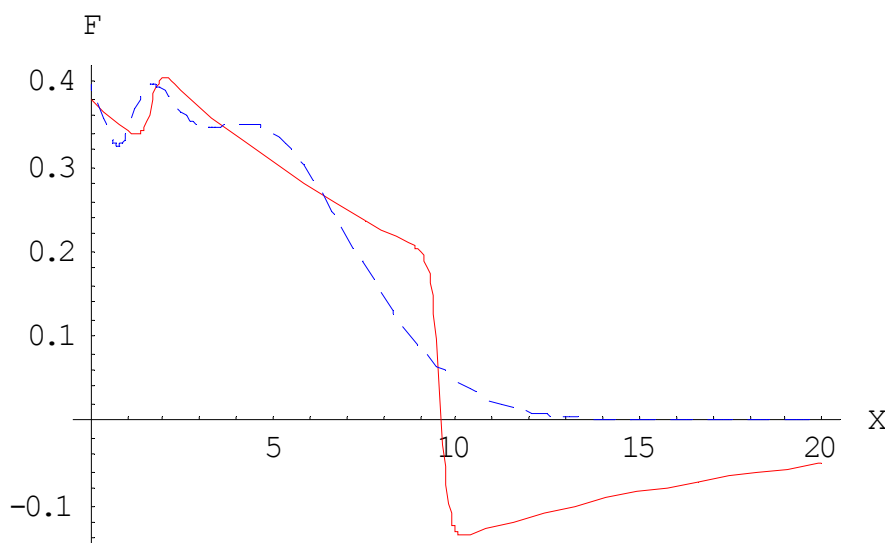
réseaux de neurones ondelettes et justifie notre choix de l'ondelette chapeau mexicain comme fonction d'activation pour tester l'influence du nombre de neurones ainsi que le nombre d'itérations sur la meilleure approximation et même pour l'approximation d'autres fonctions ultérieurement.

**III.7.2 Etude de l'Influence du nombre de neurones**

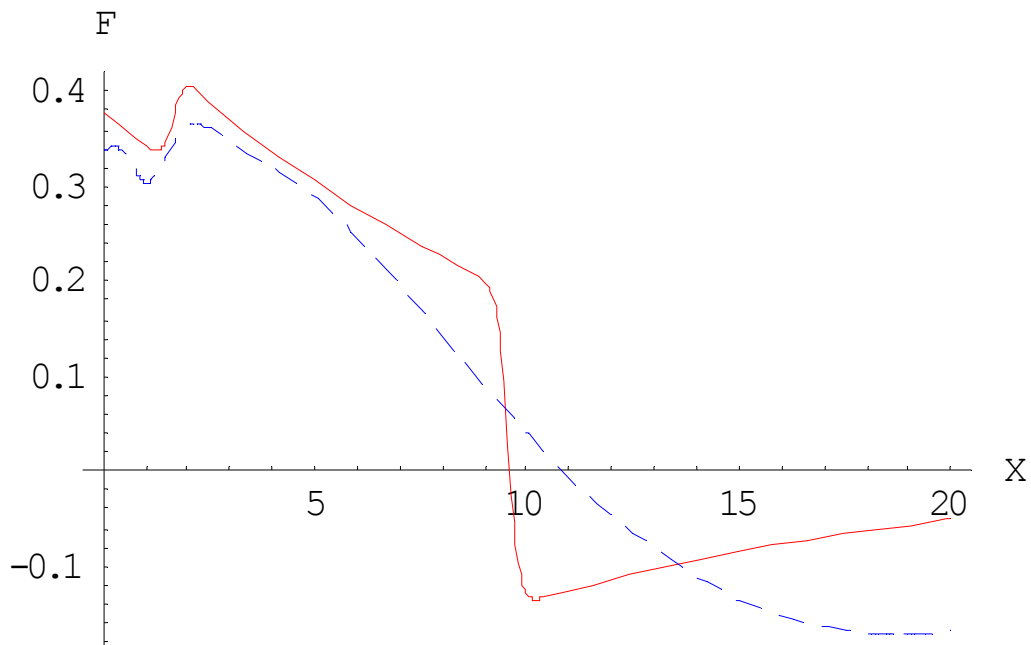
Pour approfondir notre recherche afin de déceler les paramètres qui influent sur la meilleure approximation, il s'avère utile d'examiner le rôle du nombre de neurones cachés d'un réseau de neurones ainsi que son impact sur la meilleure approximation. Il est utile de signaler que dans cette partie, nous avons modifié la tâche en optant cette fois-ci pour une autre tâche dont l'approximation est moins appréciable, pour que l'impact du nombre de neurones ainsi que le nombre d'itérations soient manifestement clairs, la tâche choisie est définie par l'expression analytique suivante:

$$F(x) = 0.10167 \exp(-x/10) \{ 0.60717 \tanh[4(x - 1.66133)] - 4.33575 \tanh[4(x - 9.56591)] \} \quad (\text{III-33})$$

Nous avons utilisé la gaussienne et l'ondelette chapeau mexicain comme fonctions d'activation. Les figures (III.6) et (III.7) représentent la nouvelle tâche et son approximation dans le cas de la fonction gaussienne et l'ondelette chapeau mexicain avec comme nombre de neurones N=5 et un nombre d'itérations égal à 71.



**Figure III.6:** Tracé de la deuxième tâche III-33 (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas d'une gaussienne.



**Figure III.7:** Tracé de la deuxième tâche III-33 (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain.

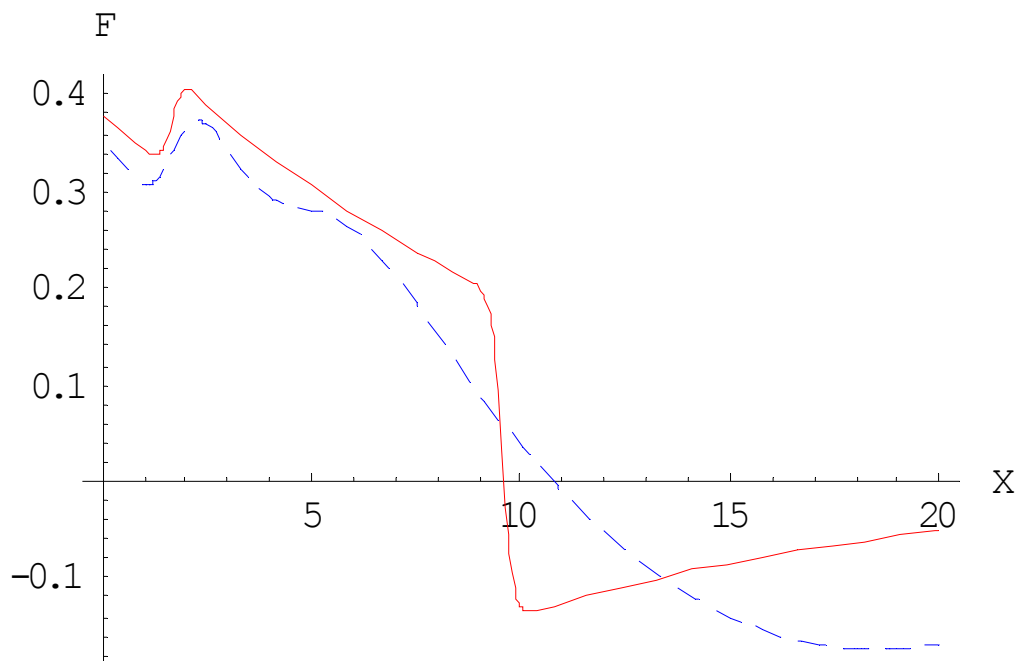
Les valeurs du MSE relatives à chaque fonction d'activation sont affichées dans le tableau 2:

<b>Fonction d'activation</b>	<b>MSE</b>
Gaussienne	0.129789
Ondelette chapeau mexicain	0.1144430

**Tableau 2:** Les valeurs du MSE relatives à chaque fonction d'activation dans le cas de la deuxième tâche.

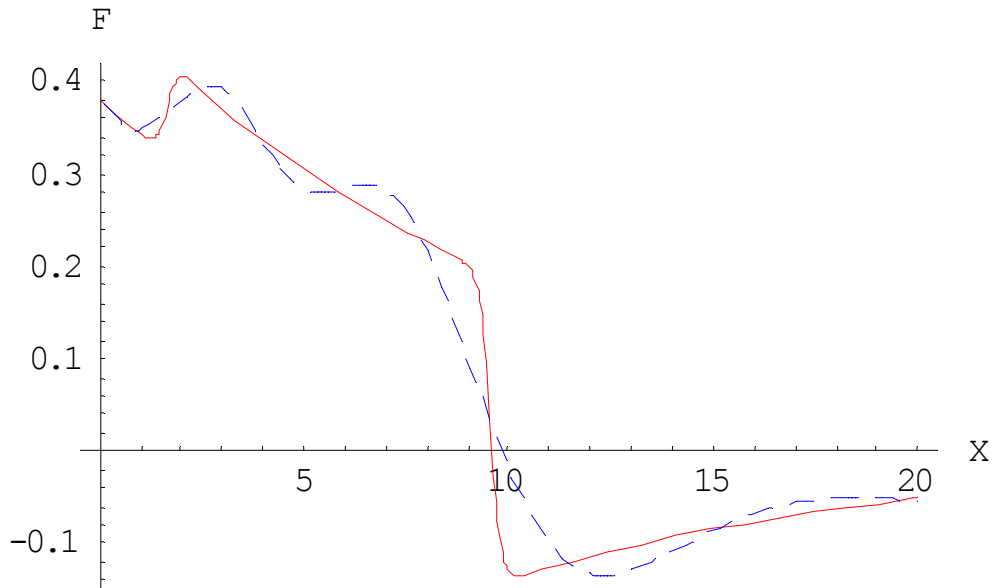
Les résultats affichés ci-dessus et l'analyse des deux graphes précédents confirment la déduction obtenue précédemment, à savoir que la meilleure approximation est celle qui émane du chapeau mexicain comme fonction d'activation.

Pour montrer l'influence du nombre de neurones, nous avons fait varier ce paramètre de  $N=5$  jusqu'à  $N=19$ , le nombre d'itérations étant toujours choisi à une valeur qui vaut 71. En se référant au résultat conclue auparavant, l'ondelette choisie est celle du chapeau mexicain. Les résultats sont représentés par les figures (III.8)- (III.13).

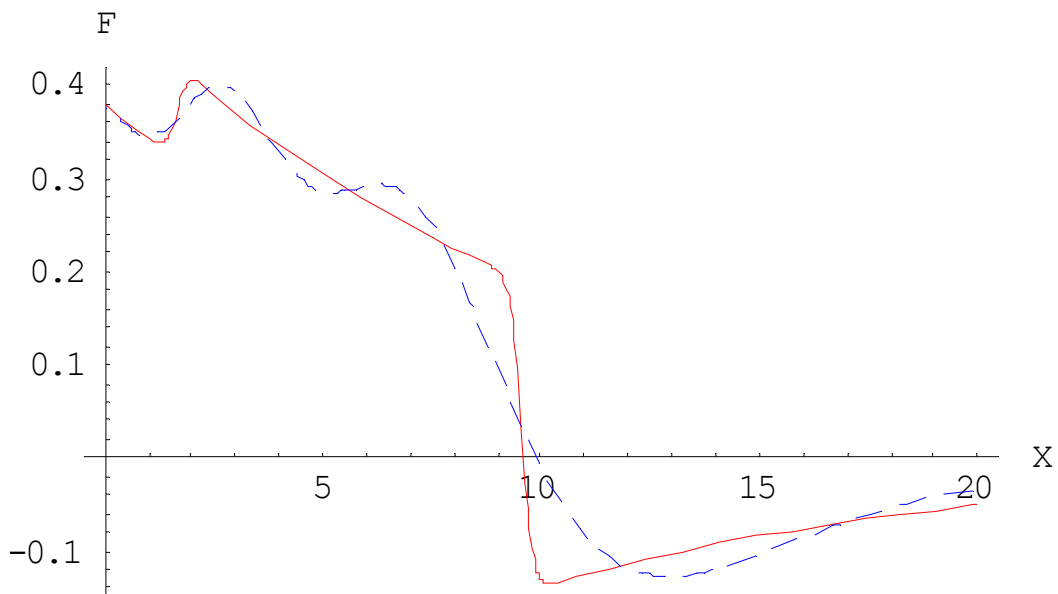


**Figure III.8:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre de neurones égal à 7.

L'APPROXIMATION DE TACHES

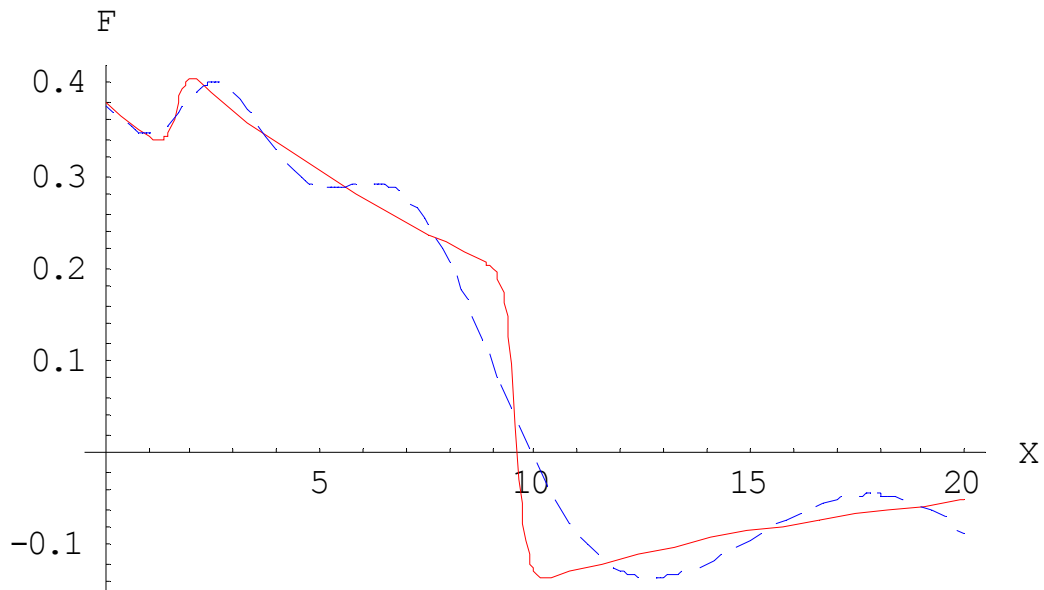


**Figure III.9:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre de neurones égal à 9.

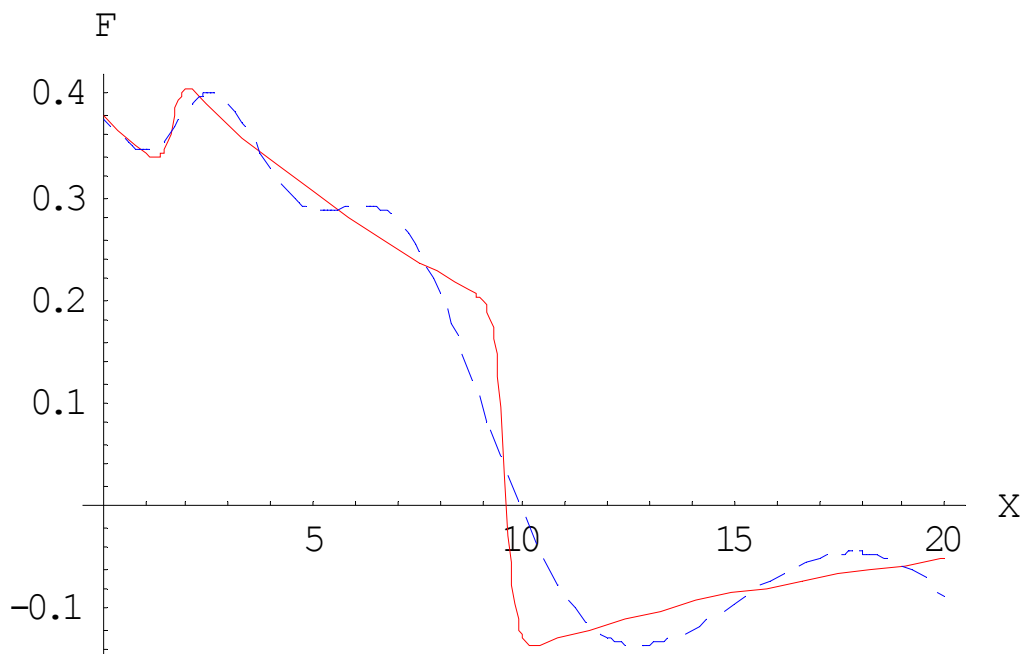


**Figure III.10:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre de neurones égal à 11.

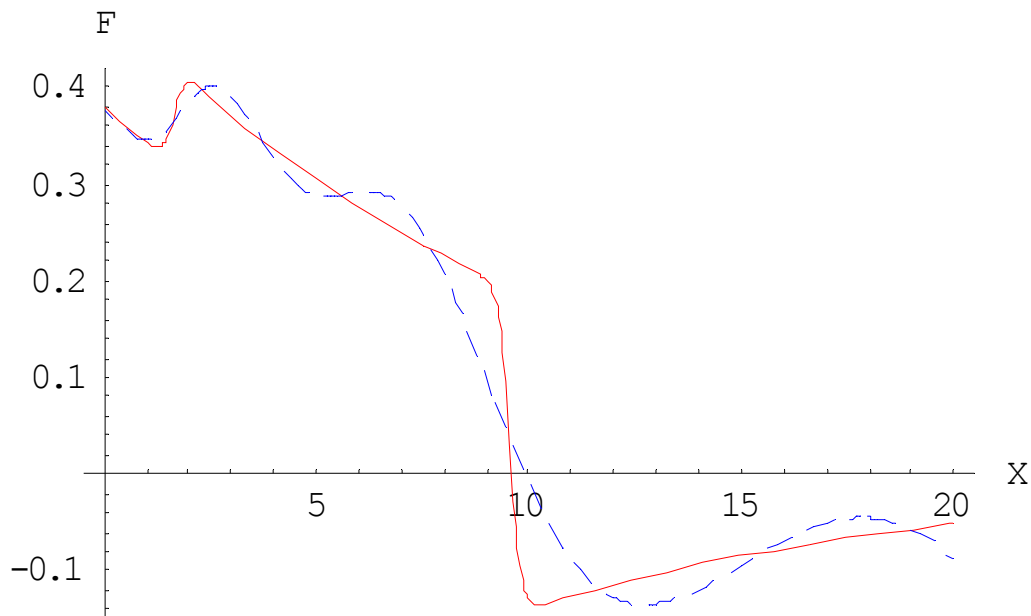
L'APPROXIMATION DE TACHES



**Figure III.11:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre de neurones égal à 13.



**Figure III.12:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre de neurones égal à 15.



**Figure III.13:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre de neurones égal à 19.

Après une analyse globale des différentes courbes ainsi que des valeurs du MSE, nous pouvons conclure qu'une meilleure approximation, ainsi qu'une diminution du MSE sont obtenues, avec l'augmentation du nombre de neurones jusqu'à un nombre de neurones  $N=9$ . Cette amélioration de l'approximation est suivie par une saturation du réseau, ceci est dû principalement aux problèmes des minima locaux, et de la négligence du paramètre translationnel. Les différentes valeurs du MSE relatives à chaque nombre de neurones sont mises en exergue dans le tableau 3:

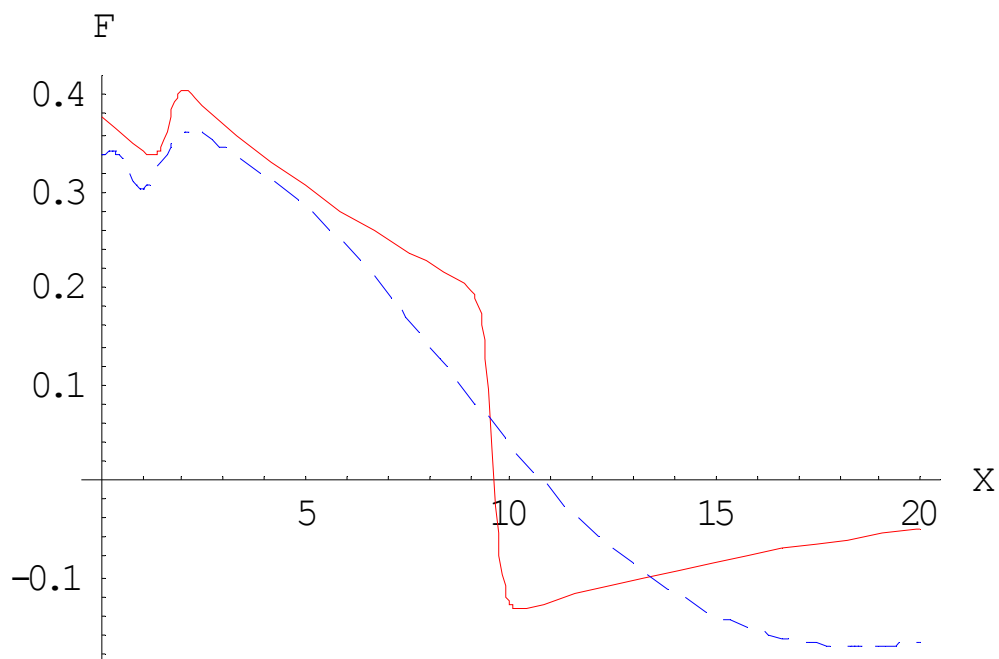
Nombre de neurones	MSE
N=5	0.1144430
N=6	0.1144170
N=7	0.1124710
N=8	0.1124710
N=9	0.0205184
N=10	0.0205189
N=11	0.0249453
N=12	0.0249453
N=13	0.0242917
N=14	0.0242917
N=15	0.0243052
N=16	0.0243052
N=17	0.0243052
N=18	0.0243109
N=19	0.0243136

**Tableau 3:** Les valeurs du MSE relatives à chaque nombre de neurones dans le cas de la deuxième tâche

Nous constatons finalement que l'écart entre l'approximation réalisée par un réseau de neurones et la fonction à approcher est inversement proportionnel au nombre de neurones cachés. L'idée la plus utile donc consisterait à choisir le nombre de neurones cachés le plus grand possible, puisque c'est ce qui assure la plus grande précision à l'approximation d'une fonction.

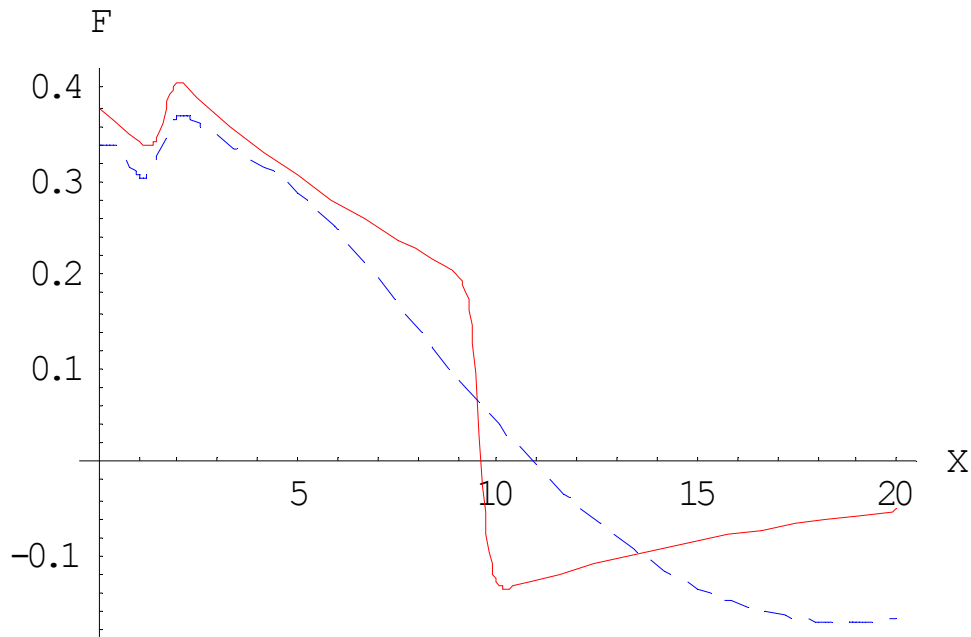
### III.7.3 Etude de l'Influence du nombre d'itérations

En plus du nombre de neurones, le nombre d'itérations est un paramètre qui a assurément un impact palpable sur la meilleure approximation. A cet effet, nous avons modifié le nombre d'itérations de 100 jusqu'à 2000 avec un nombre de neurones  $N=5$  et la même fonction d'activation que précédemment. Les figures (III.14)-(III.17) indiquent qu'une augmentation du nombre d'itérations aboutit à une légère amélioration de l'approximation.

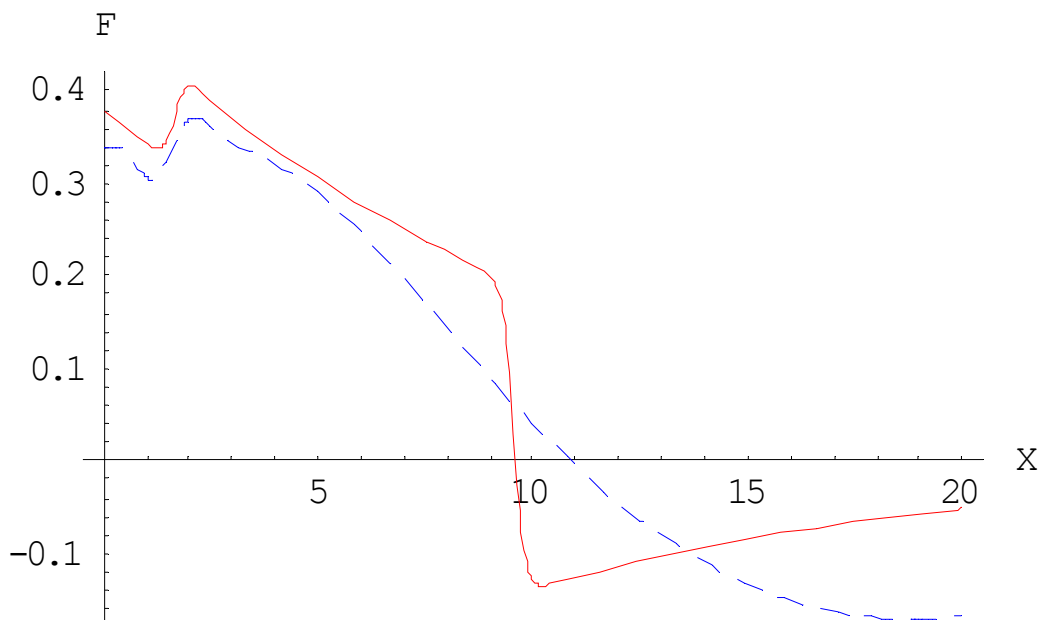


**Figure III.14:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig 3.7) avec un nombre d'itérations égal à 100.

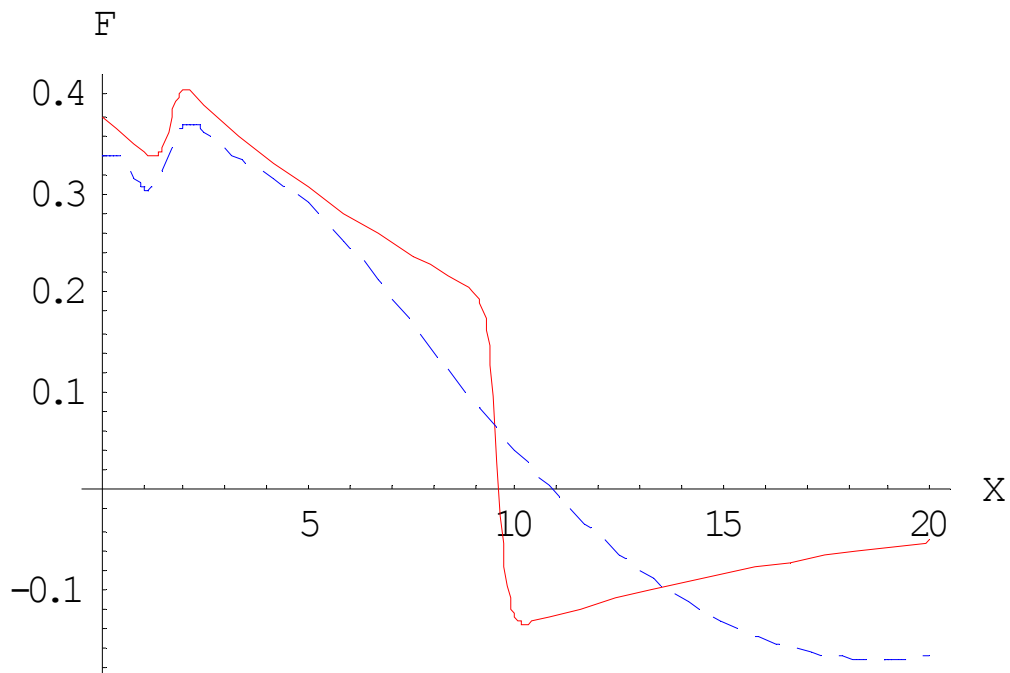
L'APPROXIMATION DE TACHES



**Figure III.15:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre d'itérations égal à 500.



**Figure III.16:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre d'itérations égal à 800.



**Figure III.17:** Tracé de la tâche (ligne continue) et de sa meilleure approximation (ligne discontinue) dans le cas de l'ondelette chapeau mexicain (Fig III.7) avec un nombre d'itérations égal à 2000.

L'augmentation du nombre d'itérations entraîne nécessairement une diminution de la valeur du MSE comme le désigne le tableau 4:

Nombre d'itérations	MSE
iter=100	0.115047
iter=150	0.11443
iter=200	0.114301
iter=250	0.114256
iter=300	0.114233
iter=400	0.114212
iter=500	0.114202
iter=600	0.114197
iter=700	0.114195
iter=800	0.114193
iter=900	0.114192
iter=1000	0.114191
iter=2000	0.114192

**Tableau 4:** Les valeurs du MSE relatives à chaque nombre d'itérations dans le cas de la deuxième tâche.

Finalement, nous déduisons que les paramètres qui influent sur la meilleure approximation sont: la fonction d'activation, le nombre de neurones ainsi que le nombre d'itérations. C'est pour cette raison justement que nous devons, a priori opter pour un choix judicieux de ces trois facteurs. Il est pareillement utile de préciser que dans cette approche nous avons considéré uniquement le paramètre d'échelle, le paramètre translationnel étant négligé et la prise en compte de ce dernier influe positivement sur l'obtention d'une approximation meilleure [48].

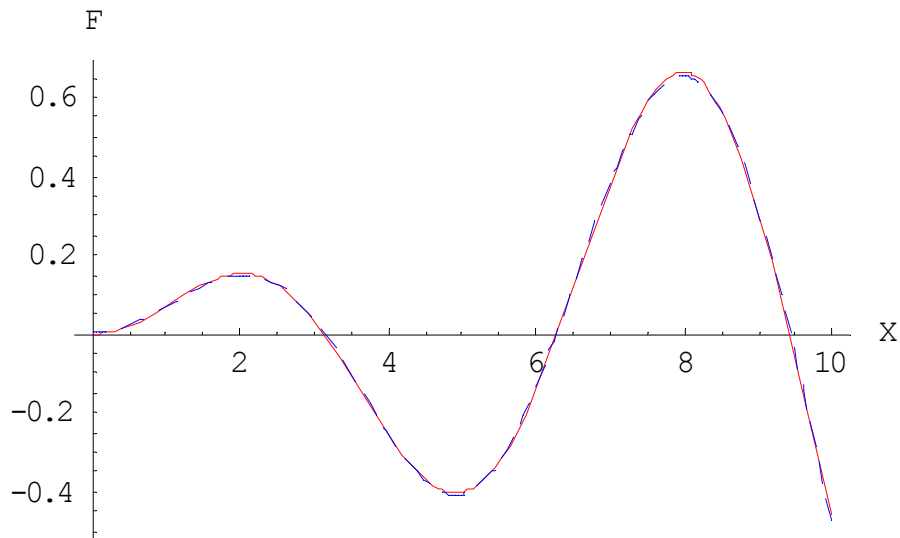
### III.7.4 Approximation d'autres fonctions

Dans ce volet, nous allons valider le modèle des réseaux de neurones-ondelettes choisi en l'utilisant pour l'approximation d'autres fonctions. Ces fonctions qui sont relativement lissées et qui ne sont pas à caractère oscillatoire à l'instar de la tâche utilisée précédemment. Il est indispensable de citer que pour l'approximation de ce type de fonctions, le choix d'un nombre de neurones et d'un nombre d'itérations réduit entraîne une très bonne approximation. Nous avons considéré les tâches définies par les expressions suivantes communément utilisées dans la littérature [49] :

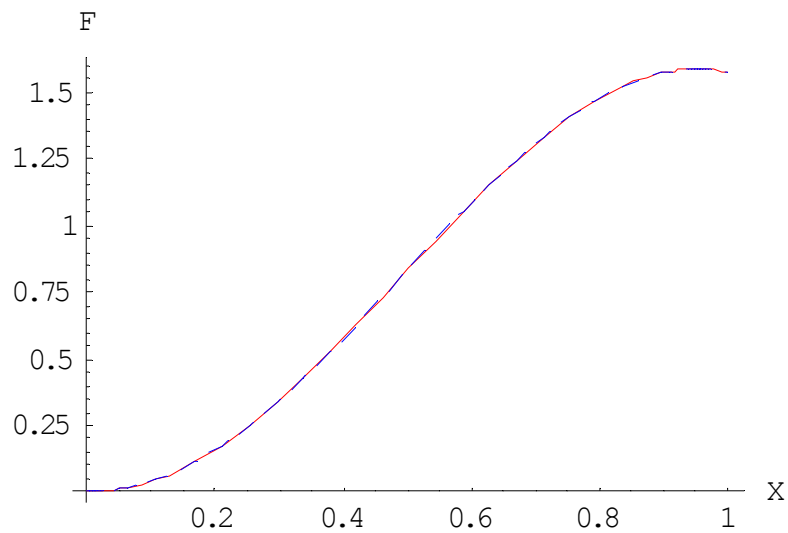
$$F(x) = \sin x \sin 2x \quad (\text{III-34})$$

$$F(x) = x \sin x \quad (\text{III-35})$$

Les résultats obtenus sont représentés par les figures (III.18) et (III.19). Nous concluons qu'une très bonne approximation a été obtenue. La fonction à approcher et sa meilleure approximation sont en très bon accord et ce dans les deux cas considérés et avec des valeurs de MSE très faibles qui valent respectivement 0.000344618 et  $1.44025 \cdot 10^{-9}$ .



**Figure III.18:** Tracé de la tâche  $F(x) = x \sin x$  (ligne continue) et de sa meilleure approximation (ligne discontinue).



**Figure III.19:** Tracé de la tâche  $F(x) = \sin x \sin 2x$  (ligne continue) et de sa meilleure approximation (ligne discontinue).

**CHAPITRE IV :**

**APPLICATION DES RESEAUX**

**DE NEURONES- ONDELETES**

**A L'APPROXIMATION DES**

**POTENTIELS SEXTIQUES**

## **IV-1 Introduction**

Durant plusieurs années, un grand intérêt a été conféré à l'équation de Schrödinger, en s'intéressant principalement à la résolution exacte de cette équation. Malheureusement cette équation n'admet pas toujours une solution exacte [50].

Au fait, la résolution du problème aux valeurs propres de l'hamiltonien d'un système physique, est l'une des tâches que doit réaliser un physicien en mécanique quantique. En effet, dès le début du 20<sup>ème</sup> siècle, de nombreuses méthodes de résolution de l'équation de Schrödinger ont vu le jour. Les premiers modèles étaient destinés à des formes simples de potentiels dont la résolution peut se faire exactement. Jusqu'à 1979, les potentiels étaient classés en deux types: les potentiels exactement solubles [51,52] et non exactement solubles [53]. Ces méthodes proposées s'appliquent à une grande classe de potentiels, ce sont les potentiels *Q.E.S* (Quasi- exactement solubles) [53]. Actuellement, l'oscillateur anharmonique est généralement utilisé comme un modèle mathématique pour la description d'un grand nombre de phénomènes physiques.

Dans cette optique et afin de tester l'efficacité de notre méthode, nous l'appliquons pour un ensemble de potentiels caractérisés par leurs problèmes numériques particulièrement dans la résolution de l'équation de Schrödinger qui n'aboutit pas à des résultats satisfaisants. Dans la première partie, l'accent est mis particulièrement sur l'approximation d'un ensemble de potentiels sextiques. En utilisant l'approche des réseaux de neurones ondelettes, nous déduisons la nouvelle expression approchée relative à chaque potentiel sextique considéré. Dans une seconde partie, cette forme approchée du potentiel est introduite dans l'équation de Schrödinger indépendante du temps à une dimension et la résolution de cette équation permet d'obtenir l'énergie de l'état fondamental ainsi que l'énergie du premier état excité.

Un processus de comparaison est effectué par la suite entre les énergies trouvées dans le cadre de notre approche et celles trouvées dans le cadre de la méthode des potentiels quasi- exactement solubles *Q.E.S* [50,53].

### IV-2.1 La méthode des *Q.E.S*

Le potentiel quasi-exactement soluble que l'on veut utiliser dans ce travail est de la forme [50, 53]:

$$V(x) = \frac{n(n-1)}{2x^2} + (2\alpha^2 - 6\beta - 4n\beta)x^2 + 8\alpha\beta x^4 + 8\beta^2 x^6 \quad (\text{IV-1})$$

$n$  est le niveau d'énergie.  $\alpha, \beta$  sont des paramètres à déterminer

En écrivant l'équation de Schrödinger indépendante du temps à une dimension avec  $\hbar=m=1$ :

$$-\frac{1}{2}\Psi_n''(x) = [E_n - V(x)]\Psi_n(x) \quad (\text{IV-2})$$

et en choisissant une fonction d'essai de la forme:

$$\Psi_n(x) = x^n e^{-\alpha x^2 - \beta x^4} \quad (\text{IV-3})$$

le terme  $-\frac{1}{2}\Psi_n''(x)$  est obtenu après dérivation:

$$-\frac{1}{2}\Psi_n''(x) = \left[ -\frac{n(n-1)}{2x^2} + (2n+1)\alpha + (4n\beta + 6\beta - 2\alpha^2)x^2 - 8\alpha\beta x^4 - 8\beta^2 x^6 \right] \Psi_n(x) \quad (\text{IV-4})$$

La comparaison entre les équations : (IV-1), (IV-2) et (IV-4), permet d'avoir la relation suivante:

$$E_n = (2n+1)\alpha \quad (\text{IV-5})$$

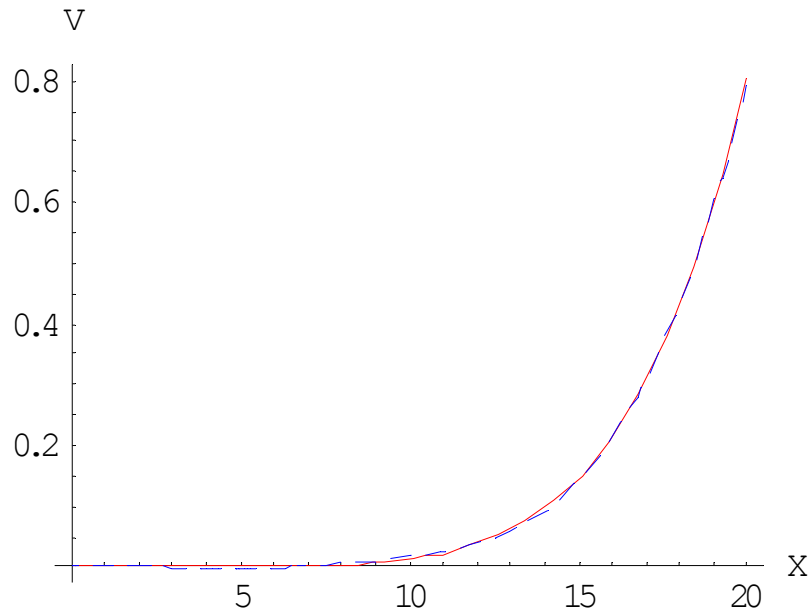
Cette équation (IV-5) nous donne le spectre des énergies du potentiel quasi-exactement soluble donné par l'équation (IV-1).

### **IV-2.2 Résultats numériques et discussions**

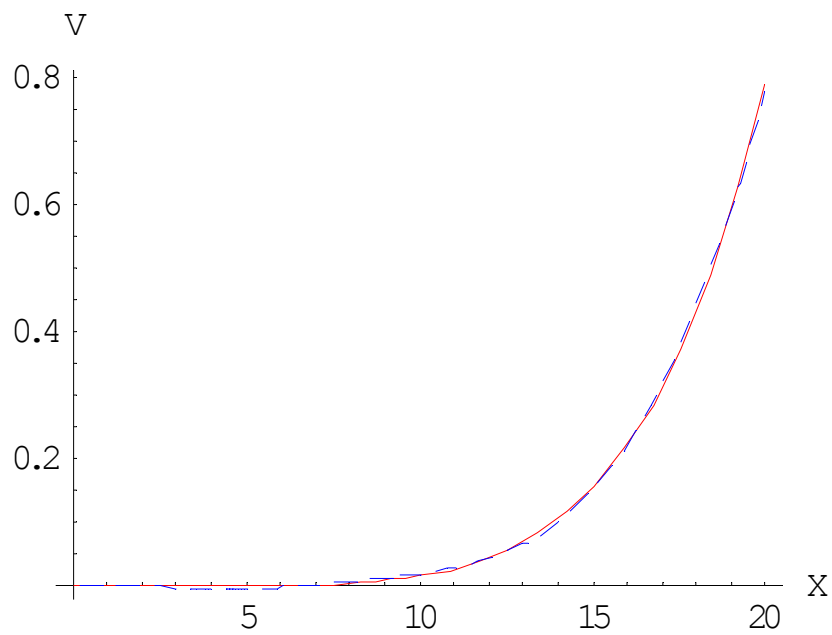
Pour la première partie et afin d'approcher un ensemble de potentiels sextiques dans le cadre de la méthode des réseaux de neurones-ondelettes. Nous avons choisi l'ondelette chapeau mexicain comme fonction d'activation. Pour des valeurs des paramètres d'échelles initialisées à 4, 5, 19, 33, 35, et un nombre de neurones  $N=5$  et un nombre d'itérations égal à 3001, le réseau converge et une saturation de  $\|F_{app}\|^2$  débute où un processus de comparaison peut être effectué entre  $F$  et  $F_{app}$ . Les résultats obtenus sont représentés par les figures (IV.1)-(IV.6). L'analyse globale des courbes révèle une très bonne approximation, où les deux courbes, le potentiel à approcher et sa meilleure approximation sont en très bon accord. Cet accord est confirmé par les faibles valeurs du MSE, affichées ci- dessous :

Le potentiel	MSE
$V_1(x) = 1.02x^2 + 0.48x^4 + 0.32x^6$	0.000211692
$V_2(x) = 0.4808x^2 + 0.0416x^4 + 0.0008x^6$	0.000201829
$V_3(x) = 0.1368x^2 + 2.848x^4 + 0.32x^6$	0.000205943
$V_4(x) = 2.7248x^2 + 13.632x^4 + 0.72x^6$	0.000211883
$V_5(x) = -0.4592x^2 + 0.416x^4 + 0.08x^6$	0.000210931
$V_6(x) = 4.3368x^2 + 2.848x^4 + 0.32x^6$	0.000211066

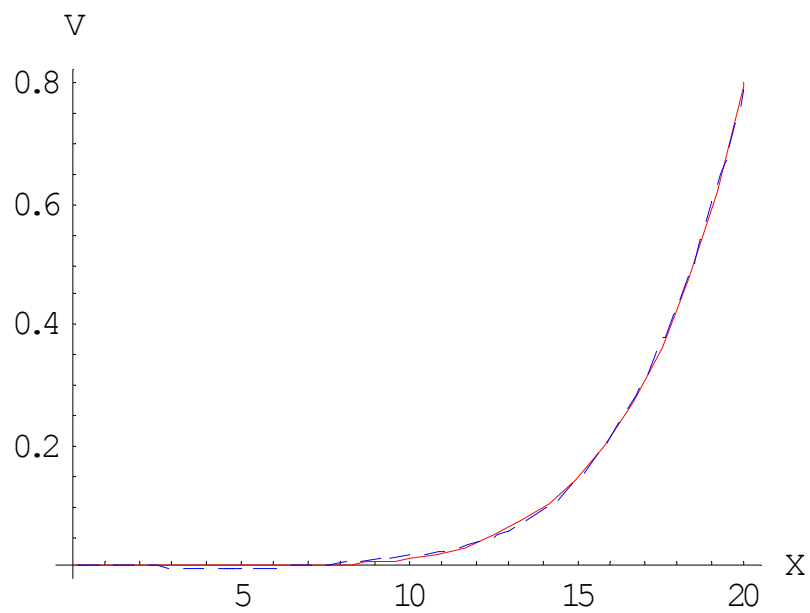
**Tableau 1:** Les valeurs du MSE relatives à chaque approximation de potentiel sextique.



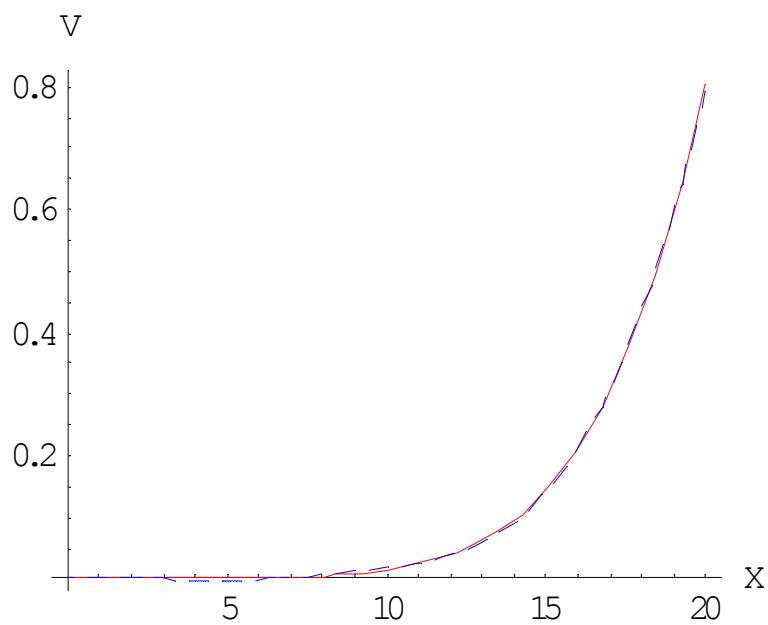
**Figure IV.1:** Tracé de  $V_1(x)$  (ligne continue) et de sa meilleure approximation (ligne discontinue).



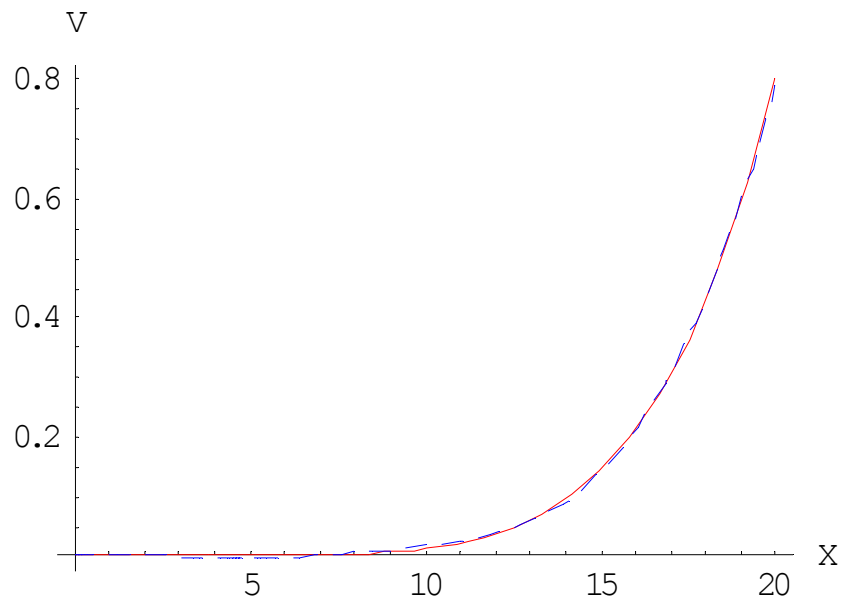
**Figure IV.2:** Tracé de  $V_2(x)$  (ligne continue) et de sa meilleure approximation (ligne discontinue).



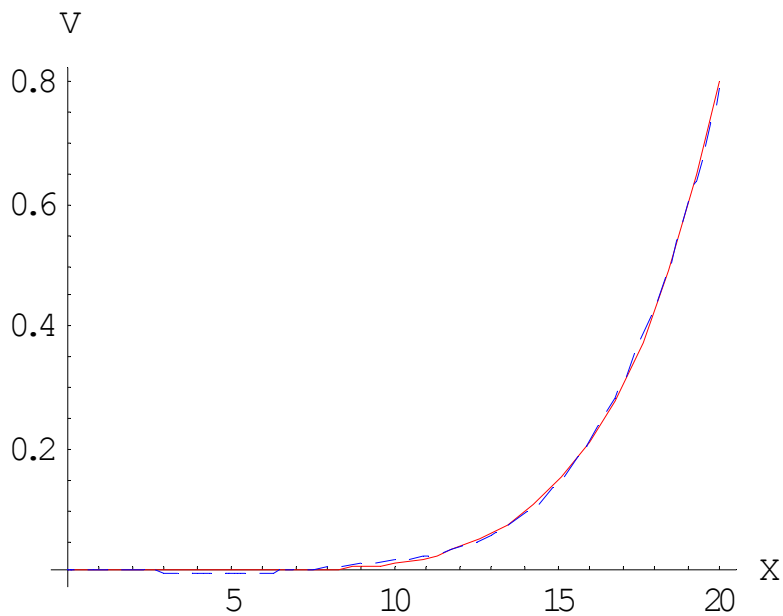
**Figure IV.3:** Tracé de  $V_3(x)$  (ligne continue) et de sa meilleure approximation (ligne discontinue).



**Figure IV.4:** Tracé de  $V_4(x)$  (ligne continue) et de sa meilleure approximation (ligne discontinue).



**Figure IV.5:** Tracé de  $V_5(x)$  (ligne continue) et de sa meilleure approximation (ligne discontinue).



**Figure IV.6:** Tracé de  $V_6(x)$  (ligne continue) et de sa meilleure approximation (ligne discontinue).

Un résultat préliminaire satisfaisant de l'efficacité de notre approche en matière d'approximation de tâches, cependant il demeure insuffisant. L'expression approchée relative à chaque potentiel présenté auparavant est substituée dans l'équation de Schrödinger indépendante du temps à une dimension. La résolution de cette équation permet d'obtenir l'énergie de l'état fondamental pour les quatre premiers potentiels et l'énergie du premier état excité pour les deux derniers potentiels. Un processus de comparaison sera établi par la suite entre les énergies trouvées par notre modèle et celles résultantes de la méthode *Q.E.S.* Ecrivons maintenant l'expression approchée dans le cadre de notre modèle :

$$V(x) = \sum_{i=1}^N w_i f(x / \lambda_i) \quad (\text{IV-6})$$

$N$  est le nombre de neurones et  $f$  est la fonction d'activation.

L'expression (IV.6) est introduite dans l'équation de Schrödinger indépendante du temps comme suit :

$$\left( \frac{-\hbar^2}{2m} \frac{d^2}{dx^2} + \sum_{i=1}^N w_i f(x / \lambda_i) \right) \psi(x) = E \psi(x) \quad (\text{IV-7})$$

En utilisant la nouvelle expression du potentiel approchée, nous résolvons cette équation numériquement, nous utilisons une méthode conçue pour la résolution de l'équation de Schrödinger indépendante du temps à une dimension communément appelée méthode de Numérov [54,55]. Les valeurs des poids synaptiques  $w_i$  ainsi que les paramètres d'échelles  $\lambda_i$ , issus de nos codes de calculs relatifs à chaque approximation de potentiel sextique sont mises en exergue dans les tableaux qui suivent :

$$V_1(x) = 1.02x^2 + 0.48x^4 + 0.32x^6 :$$

i	1	2	3	4	5
$w_i$	0.455133	-0.780835	10.4939	-155.757	145.609
$\lambda_i$	6.18232	7.03124	19.2453	33.0544	35.0459

$$V_2(x) = 0.4808x^2 + 0.0416x^4 + 0.0008x^6 :$$

i	1	2	3	4	5
$w_i$	0.360602	-0.623417	9.32924	-140.78	131.716
$\lambda_i$	6.00115	6.87096	19.2314	33.0516	35.0436

$$V_3(x) = 0.1368x^2 + 2.848x^4 + 0.32x^6$$

i	1	2	3	4	5
$w_i$	0.407814	-0.70148	8.80618	-60.9496	52.4400
$\lambda_i$	6.10446	6.96396	19.2386	34.0486	40.0304

$$V_4(x) = 2.7248x^2 + 13.632x^4 + 0.72x^6$$

i	1	2	3	4	5
$w_i$	0.457059	-0.784007	10.5117	-155.985	145.804
$\lambda_i$	6.18559	7.03412	19.2454	33.0544	35.0459

$$V_5(x) = -0.4592x^2 + 0.416x^4 + 0.08x^6 :$$

i	1	2	3	4	5
$w_i$	0.447077	-0.767499	10.3997	-154.568	144.492
$\lambda_i$	6.16844	7.01897	19.2443	33.0542	35.0458

$$V_6(x) = 4.3368x^2 + 2.848x^4 + 0.32x^6$$

i	1	2	3	4	5
$w_i$	0.439104	-0.754287	10.3052	-153.358	143.37
$\lambda_i$	6.15446	7.00066	19.2432	33.0539	35.0456

Les énergies des états fondamentaux relatives aux quatre premiers potentiels ainsi que l'énergie du premier état excité relative aux deux derniers potentiels sont consignées dans le tableau 2 [50,53]:

	$E_{0app}$	$E_{0QES}$ [50,53]	$\frac{\Delta E}{E_{Q.E.S}} \%$
$V_1(x) = 1.02x^2 + 0.48x^4 + 0.32x^6$	0.330	0.300	10
$V_2(x) = 0.4808x^2 + 0.0416x^4 + 0.0008x^6$	0.522	0.522	0
$V_3(x) = 0.1368x^2 + 2.848x^4 + 0.32x^6$	1.650	1.780	7
$V_4(x) = 2.7248x^2 + 13.632x^4 + 0.72x^6$	5.340	5.680	5
	$E_{1app}$	$E_{1QES}$	$\frac{\Delta E}{E_{Q.E.S}} \%$
$V_5(x) = -0.4592x^2 + 0.416x^4 + 0.08x^6$	1.470	1.560	6
$V_6(x) = 4.3368x^2 + 2.848x^4 + 0.32x^6$	5.347	5.340	0.1

**Tableau 2:** Les valeurs des énergies relatives à chaque potentiel sextique

Une analyse détaillée des résultats affichés ci-dessus révèle une adéquation probante entre nos valeurs des énergies issues de l'expression approchée du potentiel et celles calculées dans le cadre de la méthode *Q.E.S* particulièrement pour potentiels  $V_2, V_4, V_5, V_6$ . Cette adéquation demeure moins performante pour les potentiels  $V_1, V_3$ . Nous pensons que ceci serait dû principalement au caractère approximatif de notre approche. Nous précisons que l'introduction du paramètre translationnel permettra probablement de pallier ce défaut [45,48].

## *Conclusion générale*

Les travaux présentés dans cette thèse ont deux objectifs. Le premier est l'adaptation des réseaux de neurones-ondelettes pour l'approximation d'une tâche dont l'expression analytique est connue. Nous avons utilisé pour cela, une architecture bien définie d'un réseau en considérant différentes fonctions d'activation. Une étude comparative a été faite et a montré que l'ondelette « chapeau mexicain » est la mieux adaptée pour approcher la tâche choisie. Ainsi, nous pouvons déduire que les réseaux d'ondelettes constituent une sérieuse alternative aux réseaux de neurones classiques (les perceptrons multicouches), où la fonction d'activation utilisée est la fonction sigmoïde, et aux réseaux à fonctions radiales où la fonction d'activation choisie est une gaussienne. Un choix judicieux de la fonction d'activation s'avère un facteur qui pèse sur la meilleure approximation. Néanmoins, il demeure insuffisant. Pour déceler les différents paramètres qui influent sur l'approximation, hormis la fonction d'activation, nous avons testé l'impact du nombre de neurones ainsi que le nombre d'itérations sur nos résultats. Pour montrer l'influence du nombre de neurones, nous avons fait varier ce paramètre, le nombre d'itérations étant toujours choisi à une valeur fixe, en se référant au résultat conclu auparavant, l'ondelette choisie étant celle du chapeau mexicain. Nous avons conclu que l'accroissement du nombre de neurones entraîne un perfectionnement percevable sur la meilleure approximation. L'idée la plus utile donc consisterait à choisir le nombre de neurones cachés le plus grand possible, puisque c'est ce qui assure la plus grande précision à l'approximation d'une fonction. En plus du nombre de neurones, le nombre d'itérations est un paramètre qui a assurément un impact palpable sur la meilleure approximation. A cet effet, nous avons modifié le nombre d'itérations avec un nombre de neurones fixe et la même fonction d'activation que précédemment. Nous avons déduit que l'augmentation du nombre d'itérations aboutit à une petite amélioration de

l'approximation. Finalement, nous avons conclu que les paramètres qui influent sur la meilleure approximation sont: la fonction d'activation, le nombre de neurones ainsi que le nombre d'itérations. C'est pour cette raison justement que nous devons a priori opter pour un choix judicieux de ces trois facteurs. Il est pareillement utile de préciser que dans cette approche nous avons considéré uniquement le paramètre d'échelle, le paramètre translationnel étant négligé et la prise en compte de ce dernier influe positivement sur l'obtention d'une approximation meilleure. Nous avons, par la suite, validé le modèle des réseaux de neurones-ondelettes choisi en l'utilisant pour l'approximation d'autres fonctions. Ces fonctions qui sont relativement lissées et qui ne sont pas à caractère oscillatoire à l'instar des tâches utilisées précédemment. Il est indispensable de citer que pour l'approximation de ce type de fonctions, un choix d'un nombre de neurones et un nombre d'itération réduit entraîne une très bonne approximation. Nous avons conclu qu'une très bonne approximation a été obtenue. La fonction à approcher et sa meilleure approximation sont en très bon accord. Le deuxième objectif de cette thèse est l'introduction de cette nouvelle technique, dans le domaine de la physique. Dans cette optique et afin de tester l'efficacité de notre méthode, nous l'avons appliqué pour l'approximation d'un ensemble de potentiels caractérisés par leurs problèmes numériques particulièrement dans la résolution de l'équation de Schrödinger qui n'aboutit pas à des résultats satisfaisants. Dans la première partie, l'accent est mis particulièrement sur l'approximation de ces potentiels. Les résultats acquis en matière d'approximation sont remarquables. En utilisant l'approche des réseaux de neurones ondelettes, nous avons déduit la nouvelle expression approchée relative à chaque potentiel sextique considéré. Dans une seconde partie, cette forme approchée du potentiel est introduite dans l'équation de Schrödinger indépendante du temps à une dimension et la résolution de cette équation permet d'obtenir l'énergie de l'état fondamental ainsi que l'énergie du

## *Conclusion générale*

---

premier état excité. Un processus de comparaison est effectué par la suite entre les énergies trouvées dans le cadre de notre approche et celles trouvées dans le cadre de la méthode des *Q.E.S.* Cette comparaison confirme la performance de notre approche.

Au vu, de l'importance des résultats obtenus, nous pouvons affirmer que l'utilisation du modèle des réseaux de neurones-ondelettes établit une sérieuse alternative aux autres méthodes de résolution de l'équation de Schrödinger pour les potentiels quasi-exactement solubles dont l'expression analytique est inconnue.

# *Bibliographie*

- [1] P. MAURAN, *Introduction aux réseaux de neurones*, Notes de cours sous forme HTML, Ecole supérieure d'informatique, Paris 2003.
- [2] P. BROUZENG, M. PATY et M. PIPART, *La science en questions*, (Editions, Messidory/La Frandole 1985).
- [3] M. PARIZEAU, *Réseaux de neurones*, Notes de cours sous forme HTML, Université Laval 2004.
- [4] E. DAVALO, P. NAIM et F.M. CLEMENT, *Des réseaux de neurones*, (Editions Eyrolles 1993).
- [5] C. TOUZET, *Les réseaux de neurones artificiels, Introduction au connexionnisme*, Notes de cours sous forme HTML, Juillet 1992.
- [6] O.SAHA et A.BOUTALEB, *Application des réseaux de neurones aux matériaux dans le génie civil état de l'art*, Notes de cours sous forme HTML, Ecole Normale Supérieure de l'Enseignement Techniques d'Oran.
- [7] S.BEAULNE, *Les réseaux de neurones*, Notes de cours sous forme HTML, Université de Québec.
- [8] Y. OUSSAR, *Réseaux d'ondelettes et réseaux de neurones pour la modélisation statique et dynamique des processus*, Thèse de doctorat de l'Université Pierre et Marie Curie, Paris 1998. Disponible sur le site <http://www.neurones.espci.fr>
- [9] A. BILIAU, *Analyse en ondelettes de données gravimétriques*, Thèse de License en Sciences Mathématiques, Université Catholique de Louvain, Juin 1992.

- [10] J. DEMAYER, M. BERBONNE et S. FORTHOMME, *Les ondelettes*, Notes de cours sous forme HTML, Université Libre de Bruxelles, Département de Physique.
- [11] J. MORLET, G. ARENS, E. FOUGEAU et D. GIARD, *Geophysics*, **1**, 203221 (1982).
- [12] P. GOUPILLAUD, A. GROSSMAN et J. MORLET, *Elservier Science Publishers, Amsterdam* **23**, 102 (1984).
- [13] A. GROSSMAN et B. TORRESANI, *Les ondelettes*, Notes de cours sous forme HTML, Centre de Physique Théorique, CNRS, Marseille, et Laboratoire d'Analyse, Topologie et Probabilités, Université de Provence, Marseille.
- [14] D. GABOR, *J. Inst Elect. Eng, London* **93**, 457 (1946).
- [15] Y. BENTALEB, *Analyse par ondelettes des signaux sismiques, Application aux ondes de surface*, Thèse de Doctorat de l'Université de Mohammed V – Agdal, Faculté des Sciences, Mai 2007, Maroc.
- [16] J. MORLET, *Sampling Theory and Theory Wave Propagation*, In 12<sup>th</sup> International Conference on Pattern Recognition, October 1994.
- [17] B. TORRESANI, *Analyse continue par ondelettes*, (Inter Editions/CNRS Editions, 1995).
- [18] Y. MEYER, *Les ondelettes, Algorithmes et Applications*, (Armand Colin Editeur, 1992).
- [19] S. MALLAT, *Une exploration des signaux en ondelettes*, (Les Editions de l'Ecole Polytechnique 1992).
- [20] C. BERNARD, *Ondelettes et problèmes mal posés : la mesure du flot optique et l'interpolation irrégulière*, Thèse de doctorat, Centre de Mathématique Appliquée, Palaiseau, France.

- [21] F.TRUCHETET, *Ondelettes pour le signal numérique*, (Edition Hermès 1998).
- [22] P. BORNE, M. BENREJEB et J. HAGGEGE, *Les réseaux de neurones, Présentation et applications*, (Editions Technip 2007).
- [23] G.DREYFUS, J-M.MARTINEZ, M.SAMULIDES, M.B.GORDON, F.BADRAN, S.THIRIA et L.HERAULT, *Réseaux de neurones, Méthodologie et Applications*, sous la direction de Gérard Dreyfus, 2ème édition, (Eyrolles 2002).
- [24] S.BARTHELEMY, *Les réseaux de neurones*, Notes de cours sous forme HTML, Juin 2000, France.
- [25] J.M. TORRES-MORENO, *Apprentissage et généralisation par des réseaux de neurones : étude de nouveaux algorithmes constructifs*, Thèse de doctorat de l'institut national polytechnique de Grenoble 1997.
- [26] G.DREYFUS, *Les réseaux de neurones*, Notes de cours sous forme HTML, Ecole supérieure de physique et de chimie Industrielles de la Ville de Paris (ESPCI), Laboratoire d'électronique, Septembre 1998, Paris.
- [27] M.Y. AMMAR, *Mise en œuvre des réseaux de neurones pour la modélisation de cinétiques réactionnelles en vue de la transposition Batch/Continu*, Thèse de doctorat de l'institut polytechnique de Toulouse 1998, France.
- [28] N.MOZAYYANI, A.R BAIG. et G.VAUCHER, *Neural Networks Proceedings, IEEE World Congress on Computational Intelligence*, **1**, 160 (1998).
- [29] S.KNERR, L.PERSONNAZ et G.DREYFUS, *IEEE Transactions, Neural Networks*, **3**, 962 (1992).

- [30] F.YANG et M.PAINDAVOINE, Neural Networks, IEEE Transactions, **14** (5), 1162 (2003).
- [31] L.E. HUI, K.P .SENG et K.M .TSE, IEEE Region 10 Conference, **1**, 84, (2004).
- [32] C.M.BISHOP, Neural Network for pattern recognition, Oxford University Press, 1995.
- [33] A.R. DE ALMEIDA, E.O. FREIRE, C.A.RONNOW, J.E.S.VIANNA et R.M.ROSI, Proceedings of the 43 rd IEEE Midwest Symposium, **3**, 1344, (2000).
- [34] L.OUKHELLOU et P. AKNIN, *Modified Fourier Descriptors: A new parametrization of eddy current signatures applied to the rail defect classification*, III International workshop on advances in signal processing for non destructive evaluation of materials
- [35] M.STRICKER, F.VICHOT, G.DREYFUS et F.WOLINSKI, *Training context-sensitive neural networks with few relevant examples for the TREC-9 routing*, Proceedings of the TREC-9 Conference.
- [36] I.RIVALS, D.CANAS, L.PERSONNAZ et G.DREYFUS, *Modeling and control of mobile robots and intelligent vehicles by neural networks*, Proceedings of the IEEE Conference on Intelligent Vehicles.
- [37] I. RIVALS, *Modélisation et commande de processus par réseaux de neurones: application au pilotage d'un véhicule autonome*, Thèse de Doctorat de l'Université Pierre et Marie Curie 1995, Paris.
- [38] L. GATET, *Intégration de réseaux de neurones pour la télémétrie laser*, Thèse de doctorat de l'université de Toulouse Septembre 2007, France.
- [39] K HORNIK, M. STINCHCOMBE et H. WHITE , *Multilayer feed forward networks are universal approximators*, Neural Networks **2**, 359 (1989).
- [40] K HORNIK, M. STINCHCOMBE et H. WHITE, Neural networks **3**, 551 (1990).

- [41] A.BARRON, IEEE Transaction on Information Theory **39**, 930 (1993).
- [42] D.E. RUMELHART, G.E. HINTON et R.J. WILLIAMS, *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, MIT Press, 318, (1986).
- [43] P.J. WERBOS, *Beyond regression: new tools for prediction and analysis in the behavioural sciences*, PH. D. thesis, Harvard University, 1974.
- [44] A.N.KOLMOGOROV, Dokl. Akad. Nauk SSSR **114**, 953 (1957).
- [45] L.AIT. GOUGAM, M. TRIBECHÉ et F.MEKIDECHE CHAFA, *Neural Networks* **21**, 1311 (2008).
- [46] S.MALLAT, IEEE Transactions on Pattern Analysis and Machine, (1989).
- [47] B.G.GIRAUD et A.TOUZEAU, *Physical Review* **65**, 016109 (2001).
- [48] L.AIT.GOUGAM, *Application des réseaux de neurones-ondelettes à l'approximation de tâches et à l'analyse de certaines oscillations localisées*, Thèse de doctorat de l'université de Bab Ezzouar, Juillet 2009, Algérie.
- [49] B.CROSE et P.VAN DER SMAGT, *An introduction to neural networks*, (Edition Eighth 1996).
- [50] F.CHAFA, A.CHOUCHAOUI, M.HACHMENE et F.Z.IGHEZOU, *Annals of physics* **322**, 1034 (2007).
- [51] M.F.MANNING, *Phys.Rev.***48**, 161 (1935)
- [52] C.L.YANG, *Jour.Phys.A* **18**, 2531 (1985).
- [53] A.DIAF, *Application des intégrales de Feynman en mécanique quantique; Etude des potentiels sextiques*, Thèse de magister de l'Université de Bab Ezzouar, Juin 2001, Algérie.

[54] J.L.M.QUIROZ, GONZALEZ et D.THOMSON, Computers in physics, **11**, N.5, (1997).

[55] G.SOOKAHET, La méthode de Numerov, Notes de cours sous forme HTML.