

M/621.862

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université 8 Mai 1945 – Guelma  
Faculté des Sciences et de la Technologie  
Département de Génie Electrotechnique et Automatique



**Domaine : Sciences et Technologie**  
**Filière :** Automatique et Informatique Industrielle  
**Spécialité :** Commande et Diagnostic des Systèmes Industriels



**Mémoire de fin d'études**  
**pour l'obtention du diplôme de Master Académique**

---

---

**Elimination d'interférence par réseau de neurone**

---

---

Présenté par :  
**Benguesmia Saber Montasser Amenellhe**  
**Abderrahim Soufyane**

Sous la direction de :

**Debbache Mehdi**

Juin 2015

1413109



## Remerciements

L'aboutissement de ce travail de mémoire a été possible grâce à la collaboration d'un groupe de personnes, auxquelles je dois exprimer mes remerciements.

Je tiens tout d'abord à remercier Mr Debbache Mehdi pour m'avoir proposé ce sujet, aussi pour la confiance qu'il m'a accordée en acceptant de diriger mon travail, pour sa grande disponibilité ainsi que pour les précieux conseils qu'il a su me donner pendant ces deux années.

Je remercie les membres de jury d'avoir bien voulu être rapporteurs et examinateurs de ce travail. Leurs disponibilités ont rendu cette soutenance possible.

Je souhaite aussi remercier ma famille et mes amis pour leur soutien. J'adresse aussi ma sympathie à tous mes collègues du Département de Génie Electrotechnique et Automatique, à tous ceux qui m'ont apporté leur aide, à un moment ou autre : le personnel administratif et de la bibliothèque.

Enfin, nous remercions toute personne ayant contribué de près ou de loin à la réalisation de ce travail.

## *Sommaire*

---

<b>Introduction générale</b>	<b>01</b>
<b>Chapitre 1</b>	
<b>1. Introduction</b>	<b>04</b>
<b>2. Historique</b>	<b>05</b>
<b>3. l'interférence</b>	<b>05</b>
<b>3.1 Définition</b>	<b>05</b>
<b>3.2 Annulation d'interférence</b>	<b>05</b>
<b>3.3 Le but d'élimination d'une interférence</b>	<b>06</b>
<b>3.4 Différents types d'interférences</b>	<b>06</b>
<b>3.4.1 Interférences entre porteuses (ICI)</b>	<b>06</b>
<b>3.4.2 Rapport Signal à Bruit (SNR)</b>	<b>06</b>
<b>3.4.3 Interférences inter symboles (ISI)</b>	<b>07</b>
<b>3.4.4 Interférence d'accès multiples (MAI)</b>	<b>07</b>
<b>4. Bruit</b>	<b>07</b>
<b>4.1. Bruit thermique</b>	<b>07</b>
<b>4.2. Bruit de grenaille</b>	<b>08</b>
<b>5. Echos parasites</b>	<b>08</b>
<b>5.1. Notions sur le Clutter</b>	<b>08</b>
<b>5.2. Cibles interférentes</b>	<b>09</b>
<b>6. Les différentes méthodes d'élimination d'une interférence</b>	<b>09</b>
<b>6.1. Filtrage adaptatif</b>	<b>10</b>
<b>6.1.1 L'identification</b>	<b>11</b>
<b>6.1.2. La prédiction</b>	<b>11</b>
<b>6.2. Principe du filtrage adaptatif</b>	<b>11</b>
<b>6.2.1 Par algorithme du gradient</b>	<b>11</b>
<b>6.3. Filtre de Wiener</b>	<b>13</b>
<b>6.3.1 Filtre de Wiener de type FIR</b>	<b>14</b>
<b>6.3.3. Algorithmes de filtrage adaptatif</b>	<b>14</b>
<b>6.3.3.1. Algorithme RLS</b>	<b>15</b>
<b>6.3.3.2 Algorithme LMS</b>	<b>16</b>
<b>7. Les domaines d'applications d'élimination d'une interférence</b>	<b>16</b>
<b>7.1. Domaine de l'aviation</b>	<b>16</b>
<b>7.2. Domaine de l'automobile</b>	<b>16</b>

<b>7.3. Domaine des téléphones portables</b>	<b>16</b>
<b>7.4. Militaire</b>	<b>16</b>
<b>7.5. Sécurité publique</b>	<b>17</b>
<b>7.6. Médical</b>	<b>17</b>
<b>8. Conclusion</b>	<b>17</b>
<b>Chapitre 2</b>	
<b>1. Introduction</b>	<b>18</b>
<b>2. Eléments de base des réseaux de neurones</b>	<b>18</b>
<b>2-1 Historique</b>	<b>18</b>
<b>2-2 Définition</b>	<b>20</b>
<b>2.3 Neurone biologique au modèle mathématique</b>	<b>21</b>
<b>2.3.1 Le modèle neurophysiologique</b>	<b>21</b>
<b>2.3.1.1 Structure</b>	<b>21</b>
<b>2.3.2 Les modèles mathématiques</b>	<b>22</b>
<b>2.3.2.1 Composant (le neurone artificiel)</b>	<b>22</b>
<b>2.3.2.1.1 Structure</b>	<b>22</b>
<b>2.3.2.1.2 Modélisation d'un neurone formel</b>	<b>23</b>
<b>2.3.2.1.3 Les entrées</b>	<b>24</b>
<b>2.3.2.1.4 Fonction d'activation</b>	<b>24</b>
<b>2.3.2.1.5 Fonction de sortie</b>	<b>27</b>
<b>2.4 Architecture des réseaux de neurones</b>	<b>27</b>
<b>2.4.1. Réseaux de neurones non bouclés</b>	<b>27</b>
<b>2.4.2 Réseaux de neurones bouclés</b>	<b>29</b>
<b>2.4.2.1. Propriétés des réseaux bouclés</b>	<b>30</b>
<b>2.5 Variables descriptives</b>	<b>32</b>
<b>2.6 Structure d'interconnexion</b>	<b>32</b>
<b>1. Réseau multicouche (au singulier)</b>	<b>32</b>
<b>2. Réseau à connexions locales</b>	<b>33</b>
<b>3. Réseau à connexions récurrentes</b>	<b>33</b>
<b>4. Réseau à connexion complète</b>	<b>33</b>
<b>2.7 Les types d'apprentissage des réseaux de neurones</b>	<b>34</b>
<b>2.7.1 L'apprentissage supervisé</b>	<b>34</b>
<b>2.7.2 L'apprentissage non supervisé</b>	<b>34</b>

## *Sommaire*

---

<b>3. Les réseaux de neurones les plus utilisés</b>	<b>35</b>
<b>4. Les différentes applications de RNA</b>	<b>35</b>
4.1 Reconnaissance des formes	36
4.2 Modélisation	36
4.3 Traitement de la parole	36
4.4 Détection d'anomalies en médecine	36
4.5 Traitements dépendant du temps	37
<b>5. Conclusion</b>	<b>37</b>
<b>Chapitre 3</b>	
<b>1. Introduction</b>	<b>38</b>
<b>2. Loi d'apprentissage</b>	<b>38</b>
<b>3. Algorithme de rétro-propagation du gradient</b>	<b>39</b>
3.1 La première étape	39
3.2 La deuxième étape	40
<b>4. Application des réseaux de neurones multicouches</b>	<b>41</b>
<b>5 Réseaux de neurones linéaires adaptatifs</b>	<b>42</b>
5.1. Architecture	42
<b>6-Partie Application</b>	<b>43</b>
<b>6-1 Elimination d'interférence par réseaux de neurones (type Adaline)</b>	<b>43</b>
6-1-1 Algorithme	43
<b>7. Conclusion</b>	<b>45</b>
<b>Conclusion général</b>	<b>46</b>

## Liste des figures

Figure. 1.1 Cas d'un seuil de détection très haut

Figure.1.2 : Principe de l'identification

Figure.1.3 : Principe de la prédiction.

Figure.1.4 : Principe du filtrage adaptatif.

Figure 2-1 : Représentation schématique d'un neurone.

Figure 2.2 Un neurone biologique avec son arborisation dendritique

Figure 2.3 : Le neurone formel.

Figure 2.4 Mise en correspondance neurone biologique / neurone artificiel

Figure 2.5 Différents types de fonctions de transfert pour le neurone artificiel,  
a : fonction à seuil (S , la valeur du seuil), b : linéaire par morceaux, c : sigmoïde.

Figure 2.6 : Fonction de Heaviside

Figure 2.7 : Fonction signe

Figure 2.8 : Fonction linéaire.

Figure 2.9 : Fonction linéaire à seuil.

Figure 2.10 : Fonction sigmoïde

Figure 2.11 : Un réseau de neurones non bouclé à N entrées,  
Une couche de  $N_c$  neurones cachés et  $N_s$  neurones de sortie.

Figure 2.12 : Forme canonique d'un réseau de neurones bouclé

Figure 2.13 Définition des couches d'un réseau multicouche.

Figure 2.14 Réseau à connexions locales

Figure 2.15 Réseau à connexions récurrentes

Figure 2.16 Réseau à connexions complète

Figure 2.17 Réseaux de neurones le plus utilisés.

Figure.3.1 : Réseaux de neurones linéaires adaptatifs

Figure 3.2 : Elimination d'une interférence pour Signal bruitée (Signal monodimensionnel)

Figure 3.3 : Elimination d'une interférence pour une image bruitée (Signal bidimensionnel).

# *Introduction*



## **Introduction générale**

Dans de nombreux systèmes de communication tant filaire que sans fil, le phénomène d'interférence est présent et est souvent un sérieux frein au développement de ces dits systèmes. En effet, cette interférence modifie et donc perturbe le signal d'intérêt. Cette interférence apparaît sous de multiples formes. On peut songer par exemple à l'interférence entre symboles liée à la présence d'un canal de propagation de type multi-trajets, à l'interférence entre utilisateurs (comme en 3G avec le CDMA), à l'interférence entre porteuses (dans un système OFDM mal synchronisé fréquentiellement) ou encore à l'interférence entre différents canaux (comme le problème de diaphonie en ADSL ou bien encore le problème des canaux adjacents en communication sans fil). En fait, il est possible de parler d'interférence dès que plusieurs nœuds (utilisateurs, antennes, etc.) sont présents dans un réseau donné et veulent partager une certaine ressource commune (ou dimension) disponible, soit-elle spatiale, fréquentielle ou temporelle. Une gestion intelligente de l'interférence est alors nécessaire afin de limiter son impact négatif.

Il est toutefois important de noter que, malgré un impact a priori négatif comme expliqué ci-dessus, l'interférence peut dans certaines circonstances devenir bénéfique si elle est gérée de manière adéquate. C'est, par exemple, le cas du canal radio-mobile multi-trajets couramment rencontré dans le contexte des communications mobiles urbaines. En effet, grâce aux nombreuses réflexions de l'onde émise sur les différents obstacles environnants (immeubles, voitures en mouvement, etc.), il est possible de recevoir le signal émis au niveau du récepteur et ceci même si le récepteur n'est pas en ligne de mire de l'émetteur. Clairement ce phénomène de multi-trajets est ici utile même s'il conduit à engendrer de l'interférence entre symboles. De même pour contrer l'aspect aléatoire de l'amplitude de ces trajets, il est courant maintenant d'introduire de la diversité dans la transmission par le biais d'antennes multiples qui engendrent-elles-mêmes de l'interférence. Si ces interférences créées ne sont pas bien gérées, le bénéfice est faible voire nul. En revanche, si ces interférences sont gérées de manière pertinente, elles deviennent profitables.

Dans le contexte de systèmes subissant de l'interférence, l'un des défis est de trouver des protocoles intelligents gérant efficacement cette interférence. Aujourd'hui, il existe plusieurs manières de la traiter. Tout d'abord, grâce à des techniques de codage, il est souvent possible d'orthogonaliser les liens de transmissions, ce qui annule l'interférence. Bien que très répandue en pratique (dans les réseaux de téléphonie mobile, par exemple), cette approche a l'inconvénient de ne pas exploiter convenablement les degrés de liberté du système et n'est pas

optimale d'un point de vue de la théorie de l'information. D'un autre côté, une autre technique consiste à ignorer complètement l'interférence. Dans ce cas, l'information de chaque lien est transmise simultanément et les différents récepteurs ajoutent l'interférence présente au bruit thermique. Nous savons aujourd'hui que pour de faibles niveaux d'interférence, les résultats de cette approche sont acceptables. Toutefois, si celle-ci devient trop importante, cette technique devient clairement sous-optimale. Finalement, la troisième façon de gérer l'interférence est de la considérer comme un vrai signal et ainsi de profiter de certaines de ses caractéristiques pour améliorer les performances du système. Naturellement, ceci se fait au coût d'une complexité accrue, car une détection conjointe adaptée doit être mise en place. C'est, par exemple, le cas des codes spatio-temporels, qui prennent parti de l'interférence multi-antennaire pour améliorer la fiabilité et le débit du système.

Dans cette thèse, nous suivons ce fil directeur qu'est la gestion de l'interférence et nous tâchons de proposer des solutions originales à quelques-uns des problèmes provoqués par la présence d'interférence. Nous avons ainsi abordé un sujet. Élimination d'interférence par réseau de neurone Pour cela, nous identifions un problème particulier lié à la présence d'une interférence et nous en apportons une solution originale. Dans la section suivante, nous détaillons les trois chapitres de notre étude et nous énumérons les contributions principales de ce travail.

Cette section est dédiée au plan de la thèse et à la présentation brève des principaux résultats obtenus. Nous serons concis car chaque chapitre a été écrit de manière indépendante et possède une introduction propre, où nous expliquons en détail le problème particulier que nous voulons résoudre.

Dans le premier chapitre nous avons considéré l'effet de la présence d'une interférence localisée dans une bande de fréquences proche de celle du signal utile. Nous avons proposé une technique d'annulation d'interférence utilisant un filtre de Wiener adaptatif basé sur la connaissance de quelques paramètres de covariance du spectre de l'interférence. On a étudié et observé que pour des valeurs du niveau de l'interférence proches de celui du signal l'emploi du filtre de Wiener deviennent indispensable pour assurer l'annulation d'interférence.

Dans la 2<sup>e</sup> me chapitre nous avons structuré en partie est consacrée à la présentation des réseaux de neurones artificiels. Nous commençons par donner une brève présentation de l'évolution historique de cet axe de recherche, avant de présenter le principe de fonctionnement des neurones artificiels, nous décrivons les bases essentielles des neurones biologiques ainsi

que les différentes architectures de réseaux de neurones. Nous concluons cette partie en présentant les réseaux de neurones les plus utilisés..

Dans le dernier chapitre on a présenté notre application d'annulation d'interférence par réseau de neurone linéaire type Adaline

*Chapitre I*  
*généralités sur l'interférence*  
*et les filtrages*

## 1. Introduction

Les interférences par des ondes (vagues, son, lumière) peuvent être obtenues par un raisonnement mathématique. Les particules élémentaires, comme les électrons ne devraient alors pas interférer de la même façon que ces ondes. Or d'après la mécanique quantique, on ne peut pas se restreindre à l'appellation de particule. En effet, celles-ci sont également des ondes. On devrait plutôt dire que ce ne sont ni des particules ni des ondes : ce sont des objets présentant des aspects parfois ondulatoires et parfois corpusculaires. Cette notion de dualité onde-corpuscule permet de comprendre que ce que l'on appelle souvent des particules peuvent aussi exhiber un comportement ondulatoire et donc présenter des interférences.

En radio, une interférence est la superposition de deux ou plusieurs ondes. Il est fréquent, pour les fréquences supérieures à quelques centaines de kilohertz, qu'une antenne de réception reçoive simultanément l'onde directe en provenance de l'émetteur et une (ou plusieurs) onde réfléchi(e) par un obstacle. Les deux signaux vont se superposer et, en fonction de la différence de phase entre eux, voir leurs amplitudes s'additionner ou se soustraire. Ce genre d'interférence est responsable du fading, terme anglo-saxon désignant une variation plus ou moins rapide de l'amplitude du signal reçu. Mais le phénomène ne se limite pas aux seules ondes radio. Dans le sens commun, pour la radio, cela a pris le sens de « parasite » (il s'agit en fait de l'interférence entre l'onde radio et une onde parasite)

Sur une chaîne stéréo, on peut aussi inverser le branchement d'un des deux haut-parleurs ; alors, en se promenant dans la pièce, il y aura des endroits où le son s'annule, disparaît. C'est ce phénomène qui est utilisé dans les casques anti-bruit.

Ce sont aussi des interférences qui sont à l'origine des phénomènes de diffraction (par exemple irisation d'une mince couche d'huile, décomposition de la lumière par un disque compact). Lorsque l'on accorde un instrument de musique, on prend un son de référence (diapason, la du hautbois) et on règle l'instrument de sorte que les deux sons concordent. Lorsque la différence de fréquence est faible, on perçoit des battements sonores, et ces battements ralentissent lorsque les fréquences se rapprochent.

Ce chapitre est structuré en partie est consacrée à la présentation de l'interférence. Nous commençons par donner une brève présentation de l'évolution historique de cet phénomène, avant de présenter les méthodes de filtrages, Nous concluons cette partie en présentant les domaines d'application l'élimination d'interférence.

## 2. Historique

Howells et Applebaum [1957 à 1960]

Les premiers travaux en annulation de bruit ont été effectués à la General Electric Company. Ils conçurent un système pour l'annulation d'interférence pour une antenne. En utilisant une antenne auxiliaire et un simple filtre adaptatif. Dans les années soixante, le développement du filtrage adaptatif s'intensifie

- [1965]

Le premier système d'annulation de bruit fut conçu et construit par deux étudiants à l'université Stanford. Le but était de supprimer une interférence 60 Hz à la sortie d'un électrocardiogramme. Ils se sont aperçus qu'un simple filtre (passe haut, passe bande...) n'était pas efficace, ce qui les a poussé à développer une nouvelle technique.

- [Après 1965]

L'annulation de bruit a été successivement appliqué à d'autres problèmes, tel que d'autres aspects de l'électrocardiographie, l'élimination d'interférence, l'élimination d'écho sur des longues communications téléphoniques longue distance

## 3. l'interférence

### 3.1 Définition :

L'interférence est un phénomène qui se produit lorsque 2 ondes de même nature se rencontrent et interagissent l'une avec l'autre. On peut ainsi observer les phénomènes d'interférence en optique, en télécommunications (ondes électromagnétiques) mais aussi en acoustique (ondes mécaniques) ou avec d'autres types d'ondes comme les ondes sonores. Ce phénomène n'est néanmoins observable que dans des cas bien particuliers que nous allons voir

### 3.2 Annulation d'interférence :

**Définition 1** L'annulation d'interférence (bruit) comprend l'ensemble des techniques qui permettent d'annuler l'interférence contenu dans un signal. Le signal peut être représenté par une image ou un son. Le bruit étant un composant indésirable

### 3.3 Le but d'élimination d'une interférence :

Il est nécessaire de l'éliminer le plus possible pour recueillir le signal qui nous intéresse. Les méthodes d'annulation de bruits touchent au domaine du génie électrique, plus particulièrement celui des télécommunications

### 3.4 Différents types d'interférences

#### 3.4.1 Interférences entre porteuses (ICI)

Dans cette partie, nous allons étudier l'effet des interférences ICI. Pour cela, nous allons d'abord présenter la distribution de puissance sur l'ensemble des sous porteuses à la réception, en réponse à l'émission d'un symbole sur une sous-porteuse particulière. Dans un canal mobile de types Rayleigh, est un processus complexe Gaussien à bande étroite, stationnaire au sens large, avec un spectre de Jakes [1], on a donc :

$$R_{h1}(\Delta t) = E[h_1(t)h_1(t - \Delta t)^*] = \sigma_1^2 J_0(2\pi f_d \Delta t) \quad (1.1)$$

En posant

$$t = q_1 T_s \quad \text{et} \quad t - \Delta t = q_2 T_s \quad (1.2)$$

on aura :

$$E[h_1(q_1 T_s)h_1(q_2 T_s)^*] = \sigma^2 J_0(2\pi f_d T_s (q_1 - q_2)) \quad (1.3)$$

Où  $J_0(\cdot)$  est la fonction de Bessel de première espèce d'ordre 0.

#### 3.4.2 Rapport Signal à Bruit (SNR) :

Les courbes de performance sont généralement exprimées en fonction du Rapport Signal à Bruit (RSB) obtenu après la démodulation OFDM, c'est à dire en sortie de la TFD. Il est cependant utile de pouvoir relier ce SNR à la puissance du bruit AWGN donné par  $\frac{E_b}{N_0}$ .

Le Rapport Signal à Bruit (SNR) pour le kème symbole  $[X_{(n)}]_k$  est donné (pour des symboles normalisés, décarrèlés et un canal normalisé) par [2]

$$SNR_k = \frac{\sum_{m=1}^N E[|[H_{(n)}]_{m,k}[X_{(n)}]_k|]}{E[|[Z_{(n)}]_k|^2]} = \frac{\sum_{m=1}^N P_{m,k}}{\sigma^2} = \frac{1}{\sigma^2} \quad (1.4)$$

### **3.4.3 Interférences inter symboles (ISI):**

Les interférences inter symboles ISI constituent un problème destructif qui rend complexe la récupération des données dans les systèmes OFDM quelque soient les méthodes d'estimation du canal. Toutes fois, la solution peut paraître simple car il suffit d'effectuer une séparation temporelle entre les symboles successifs, en insérant un intervalle de garde. L'insertion d'un intervalle de temps mort (ne contient pas de données) peut résoudre parfaitement le problème de l'ISI mais rend l'estimation du canal très compliquée. A cet effet, on insère un préfixe cyclique au niveau de l'entête de chaque symbole OFDM, en lieu et place du temps mort. Ce préfixe contient les mêmes données que celles contenues dans le symbole transmis. Ce préfixe va être à son tour perturbé par les différentes versions retardées des symboles précédents, mais ceci est sans effet car les données qu'il contient ne sont d'aucune importance. En contre partie, le symbole utile (à détecter) sera perturbé par les versions retardées de ce préfixe contenant les mêmes données, de ce fait ce type de perturbations est parfaitement maîtrisable.

### **3.4.4 Interférence d'accès multiples (MAI)**

L'interférence d'accès multiples (MAI) se produit quand un certain nombre d'utilisateurs partage un canal commun simultanément. Dans ce contexte, les signaux d'autres utilisateurs apparaissent comme une interférence pour un utilisateur donné.

## **4. Bruit**

Le bruit est une énergie électromagnétique indésirable avec des paramètres aléatoires qui interfèrent la capacité du radar à fonctionner dans un mode idéal et sans erreur. Le bruit apparaît typiquement comme constitué de variations aléatoires superposées au signal d'écho reçu, ainsi plus la puissance du signal désiré est faible, plus il est difficile de le discerner du bruit

Parmi les différents types de bruit qui affectent les systèmes, existe le bruit thermique et le bruit de grenaille qui sont les plus considérables et les plus connus du point de vue statistique [2].

### **4.1. Bruit thermique**

Le bruit thermique est lié principalement, à l'agitation thermique des électrons des éléments composant le récepteur, ce qui fait que ses caractéristiques principales dépendent de



la température de ces éléments. Aussi le bruit ambiant, qui résulte de la température du milieu extérieur éclairé par l'antenne est classé généralement avec le bruit thermique. Le bruit thermique est un bruit blanc, du fait que sa densité spectrale est constante (indépendante de la fréquence). D'autre part ce bruit obéit à des lois normales [2].

#### **4.2. Bruit de grenaille**

C'est un bruit électronique, appelé aussi "shot noise", et causé par le fait que le courant électrique n'est pas continu mais constitué de porteurs de charge élémentaires (électrons). Ce qui est dû principalement dans le cas du radar, aux heurtes des photons sur la surface de l'antenne, et ce qui génère des fluctuations aléatoires dans les courants électroniques. Généralement dans les systèmes radars, le bruit de grenaille est négligé par rapport au bruit thermique, du fait de la faiblesse de l'énergie des photons [2]. D'autres classifications du bruit sont présentées dans [3], où le bruit ambiant est dû à la nature de l'environnement, dont les principales sources sont le bruit atmosphérique, le bruit urbain ou industriel, et le bruit cosmique (bruit solaire).

### **5 .Echos parasites**

L'écho radar est la portion d'énergie du signal transmis, reflétée au récepteur. Cet écho radar est cependant contaminé par des signaux parasites et la discrimination entre ces derniers et l'écho utile, constitue l'une des difficultés majeures des radars.

#### **5.1. Notions sur le Clutter**

Ces échos indésirables appelés clutter, peuvent parvenir de différentes sources, du sol, de la pluie, de la terre ou d'autres précipitations, aussi des oiseaux, des insectes ou d'aurore. Le clutter a pour effet de masquer les cibles, qui passent à travers des régions occupées par les échos clutter, et de générer des fausses alarmes qui peuvent diriger l'attention des cibles réelles et produire de fausses pistes pour la poursuite. Pour réduire ces effets, les radars comptent sur la réduction de la taille de la cellule de résolution spatiale pour minimiser le clutter d'entrée, et appliquent les techniques CFAR dans la circuiterie de détection. Dans pas mal de cas le choix propre de la fréquence de polarisation peut minimiser le clutter d'entrée au radar [3]. Il existe plusieurs types de clutter, parmi lesquelles on peut trouver [4, 5]:

- **Clutter de sol** : Indépendamment de leur nature, les réflecteurs répartis sur le sol renvoient des signaux dont la puissance dépend de leur distance, de leur SER et du gain d'antenne dans leur direction.

- **Clutter de mer** : Ce type de clutter correspond à l'écho de la surface de la mer, caractérisée par sa réflectivité

## 5.2. Cibles interférentes

Ce cas est défini pour décrire des situations où le milieu de clutter est composé par un bruit blanc Gaussien homogène plus des cibles Interférentes. Ces cibles Interférentes apparaissent comme des pics dans des cellules de portée et peuvent tomber dans l'une des deux fenêtres de références ou dans les deux en même temps [6]. Lorsque les cibles interférentes s'étendent sur les cellules de références, de la cible en question, qui est la cible primaire, le seuil est élevé et la détection de la cible primaire dégrade sérieusement. Ceci est connu sous le nom de l'effet de capture. Avec un seuil trop grand (figure 1.1), certaines cibles peuvent ne pas être détectées. D'un autre côté, si le seuil n'est pas assez élevé, le nombre de fausses alarmes dues aux pics de bruits augmentent.

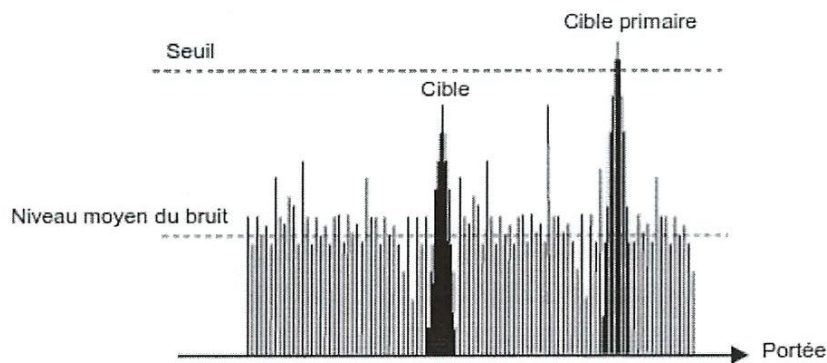


Figure. 1.1 Cas d'un seuil de détection très haut

## 6. Les différentes méthodes d'élimination d'une interférence

La méthode la plus connue et la plus ancienne est l'annulation de bruit adaptative (ANC en anglais pour 'Adaptive Noise Cancelling'), c'est-à-dire qu'on utilise le filtrage adaptatif pour annuler le bruit

Plusieurs méthodes ont été développées depuis et font toujours l'objet de nouvelles recherches. Parmi ces méthodes, citons :

Des méthodes s'appuient toujours sur le même principe de filtrage adaptatif c'est-à-dire filtrage numérique dont les coefficients évoluent en fonction des signaux reçus. Ces coefficients seront estimés par des algorithmes récursifs, au sens d'un certain critère tel que le Filtrage par LMS (Least mean squares filter) et le Filtrage par RLS (Recursive least squares filter), et la méthode d'élimination d'interférence par réseaux de neurones, qui est une technique utilisant l'outil d'intelligence artificielle (les réseaux de neurone) afin d'extraire le signal qui nous intéresse.

De façon générale les filtres adaptatifs sont des systèmes appliqués sur des données bruitées pour obtenir une information utile à un certain instant  $t$ , ces systèmes étant mis en œuvre dans trois configurations :

- Le filtrage c'est à dire l'extraction de cette information au temps  $t$  à partir des données bruitées mesurées jusqu'au temps  $t$  inclus.
- Le lissage qui utilisera aussi les données postérieures au temps  $t$ .
- La prédiction qui ne se sert que des données jusqu'au temps  $t-t$  pour déduire l'information qui nous intéresse au temps  $t$ .

### **6.1. Filtrage adaptatif**

Le filtrage adaptatif intervient quand il faut réaliser, simuler ou modéliser un système dont les caractéristiques évoluent dans le temps. Il conduit à la mise en œuvre de filtres à coefficients variables dans le temps. Les variations des coefficients sont définies par un critère d'optimisation et réalisées suivant un algorithme d'adaptation, qui est déterminés en fonction de l'application [7]. Les techniques de filtrage adaptatif trouvent tout leur sens dans les problèmes pour lesquels la composante de bruit ou le processus ont un comportement spectral inconnu. Considérons par exemple le cas d'un signal perturbé par un parasite sinusoïdal à la fréquence de 50Hz. Ce signal peut être filtré efficacement par un filtre classique coupe bande centré sur 50Hz. En revanche considérons le cas de la mesure sur électrocardiogramme du rythme cardiaque d'un bébé encore dans le ventre de sa mère. Le signal va être parasité par le rythme cardiaque de la mère. Ce signal parasite est a priori de contenu spectral inconnu et il risque même de se superposer en partie au signal correspondant au bébé. Le filtrage classique est donc ici inefficace alors que le filtrage adaptatif va se révéler performant. [8] On distingue quatre classes d'applications :

**6.1.1 L'identification :** La figure 1.2 illustre le contexte du problème d'identification. Celui-ci consiste en la détermination d'un filtre modélisant au mieux le comportement d'un processus inconnu. Seuls sont connus les signaux d'entrée/sortie de ce processus. Le filtre représentant le modèle sera estimé à partir de l'observation de la différence entre la sortie du processus et son estimation à la sortie du filtre. [7]

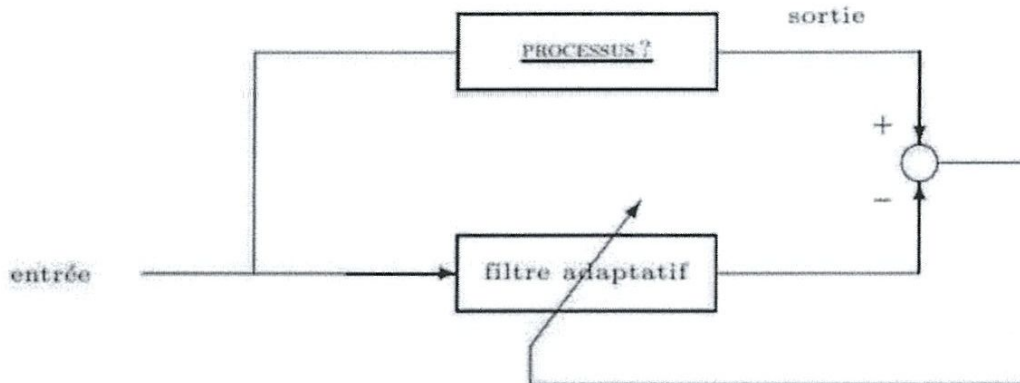


Figure.1.2 : Principe de l'identification

**6.1.2. La prédiction :** La figure 1.3 illustre le contexte du problème de prédiction. Ce problème consiste en l'estimation de la valeur future d'un signal à partir de l'information passée. Ce peut être par exemple pour prévoir la position future d'un objet, ou pour anticiper l'évolution future d'une grandeur afin de prendre au plus vite une décision. [7]

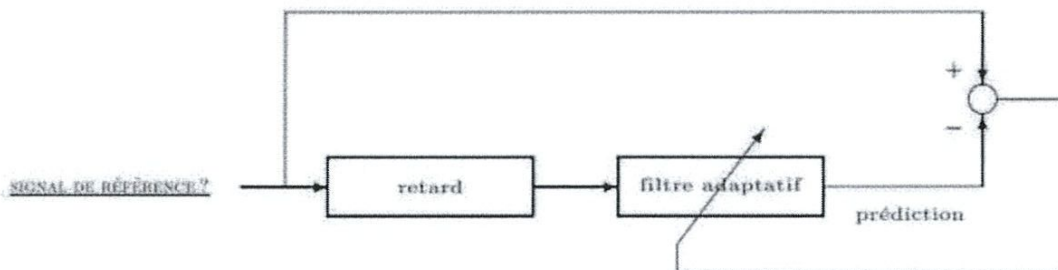


Figure.1.3 : Principe de la prédiction.

## 6.2. Principe du filtrage adaptatif

### 6.2.1 Par algorithme du gradient

Le principe du filtrage adaptatif est représenté sur la figure 1.4 ; il correspond à une opération effectuée sur un signal reçu  $x(n)$  pour fournir une sortie dont la différence avec un

signal de référence  $y(n)$  soit minimisée. Cette minimisation est obtenue en calculant les coefficients du filtre pour chaque nouvel ensemble de données, référence et signal reçu. [7]

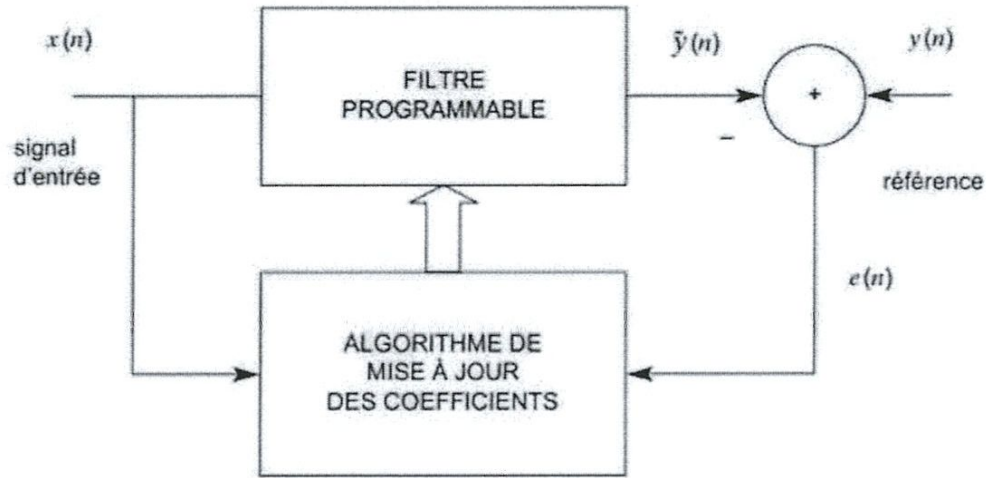


Figure.1.4 : Principe du filtrage adaptatif.

Ainsi, en supposant qu'à l'indice  $n$ ,  $n$  ensembles de données aient été reçus, les coefficients du filtre adaptatif supposé de type RIF, représentés par le vecteur  $H(n)$ , qui minimisent la fonction coût quadratique  $J(n)$  définie par :

$$J(n) = \sum_{p=1}^n [y(p) - H(n)X(p)]^2 \quad (1.5)$$

Où  $X(p)$  est le vecteur colonne d'éléments

L'estimation de la matrice d'autocorrélation du signal reçu peut s'exprimer commodément sous la forme :

$$R_N(n) = \sum_{p=1}^n X(p)X^t(p) = \sum_{p=1}^n \begin{bmatrix} x(p) \\ x(p-1) \\ \vdots \\ x(p+1-N) \end{bmatrix} [x(p), \dots, x(p+1-N)] \quad (1.6)$$

De même, l'estimation du vecteur d'inter corrélation entre référence et entrée s'écrit :

$$r_{yx}(n) = \sum_{p=1}^n y(p)X(p) \quad (1.7)$$

Quand le nouvel ensemble de données  $\{x(n+1), y(n+1)\}$  devient disponible, le vecteur des coefficients  $H(n+1)$  peut être calculé à partir de  $H(n)$ , par une mise à jour. En effet, il vient :

$$R_n(n+1) = R_N(n) + X(n+1)y(n+1) \quad (1.8)$$

$$r_{yx}(n+1) = r_{yx}(n) + X(n+1)y(n+1) \quad (1.9)$$

Et par suite :

$$R_N(n+1)H(n+1) = r_{yx}(n+1) + X(n+1)y(n+1) \quad (1.10)$$

Soit :

$$R_n(n+1)H(n+1) = [R_n(n+1) - X(n+1)X^t(n+1)]H(n) + X(n+1)y(n+1) \quad (1.11)$$

Et finalement :

$$H(n+1) = H(n) + R_n^{-1}(n+1)X(n+1)[y(n+1) - H^t(n)X(n+1)] \quad (1.12)$$

Représente l'erreur en sortie du système, calculée à l'indice  $(n+1)$ , avec les coefficients  $H(n)$  obtenus à l'indice  $n$  ; cette erreur est appelée l'erreur « a priori », alors que le même calcul avec  $H(n+1)$  correspond à l'erreur dite « a posteriori ». Les algorithmes dans lesquels les coefficients sont, à chaque valeur de l'indice, calculés par la récurrence sont les algorithmes de moindres carrés. Des algorithmes simplifiés, mais d'un grand intérêt pratique, sont obtenus en remplaçant la matrice  $(N)$  par la matrice diagonale, où  $d$  est un réel que l'on appelle le pas d'adaptation. La mise à jour des coefficients est alors faite par l'équation :

$$H(n+1) = H(n) + \delta X(n+1)e(n+1) \quad (1.13)$$

### 6.3. Filtre de Wiener

Nous exposerons l'approche statistique du problème (**filtrage de Wiener**) qui suppose la disponibilité de certaines grandeurs statistiques (moyenne et autocorrélation) du signal utile et du bruit. L'approche consiste alors à minimiser la moyenne statistique du carré de l'erreur (EQM ou MSE en anglais) entre l'information désirée et la sortie du filtre.

Ce filtrage de Wiener est inadéquat pour les situations dans lesquelles le signal ou le bruit sont non stationnaires. Dans de telles situations le filtre optimal doit être variable dans le temps. La solution à ce problème est fournie par le filtrage de Kalman.

Le filtrage adaptatif pourra aussi être utilisé dans ce cas. Par rapport au filtrage classique le filtrage adaptatif comporte une mise à jour récursive des paramètres (coefficients) du filtre.

L'algorithme part de conditions initiales prédéterminées et modifie de façon récursive les coefficients du filtre pour s'adapter au processus. Si celui-ci est stationnaire l'algorithme doit converger vers la solution optimale de Wiener, sinon il présentera une capacité à suivre des variations des grandeurs statistiques du processus si celles-ci sont suffisamment lentes. Pour présenter ces techniques adaptatives nous présenterons tout d'abord l'algorithme du gradient qui fournit un algorithme récursif de calcul des coefficients du filtre. Nous donnerons ensuite une version dans laquelle les grandeurs statistiques impliquées sont remplacées par des valeurs instantanées, on obtient alors l'algorithme très fréquemment utilisé du gradient stochastique (LMS). Nous reformulerons le problème en termes de moindres carrés et présenterons un des algorithmes récursifs basé sur cette approche : l'algorithme des moindres carrés récursifs (RLS) (dont il existe de nombreuses variantes non présentées dans le cadre de ce cours).

### 6.3.1 Filtre de Wiener de type FIR :

On se limitera ici au calcul des filtres FIR. Selon les mêmes principes, on peut calculer des filtres IIR. C'est ce qui sera vu dans la suite du cours avec les modèles ARMA utilisés en codage de parole.

Appelons  $h$ , le filtre que nous recherchons et  $N$  la longueur de sa réponse impulsionnelle donnée avec une notation matricielle par :

$$h = [h_0 \quad h_1 \quad \dots \quad h_{N-1}]^T \quad (1.14)$$

Le signal estimé  $\hat{x}(n)$  peut alors s'écrire :

$$\hat{x}(n) = \sum_{i=0}^{N-1} h_i y(n-i) \quad (1.15)$$

ou encore en introduisant la notation matricielle pour  $y(n)$  :

$$\hat{x}(n) = h^T y(n) \Leftrightarrow \hat{x}(n) = y^T(n) h \quad (1.16)$$

Avec

$$y(n) = [y(n) \quad y(n-1) \quad \dots \quad y(n-(N-1))]^T \quad (1.17)$$

### 6.3.3. Algorithmes de filtrage adaptatif :

La mise en œuvre d'un filtre (estimateur) optimal de Wiener demande la connaissance des caractéristiques du signal, du bruit et de la fonction de transfert du canal. Cela implique

également que ces caractéristiques soient stables au cours du temps, ce qui n'est pas le cas en pratique. [11]

Le filtrage adaptatif a pour objet d'approcher ces filtres optimaux. Pour cela, les coefficients de la réponse impulsionnelle du filtre sont adaptés en fonction de l'erreur par une boucle de retour comme le montre la figure ci-dessous. [10]

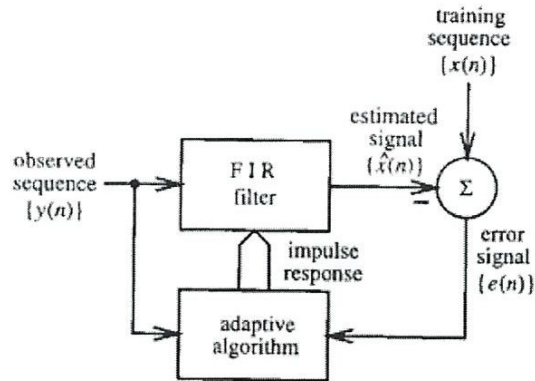


Figure.1.5 : Schéma général d'un système de filtrage adaptatif.

Cette adaptation nécessite une séquence d'apprentissage et une stratégie de mise à jour des coefficients du filtre dont l'objectif est la minimisation d'une erreur. Pour cela, on utilisera des algorithmes d'optimisation. Le détail de ces algorithmes dépasse le cadre du traitement du signal, mais on donnera ici les grandes lignes de deux approches largement utilisées en filtrage adaptatif : le RLS et le LMS. [11]

La réponse impulsionnelle d'un filtre adaptatif est donc variable dans le temps. Elle dépend du signal reçu, de la séquence d'apprentissage et de l'algorithme d'optimisation utilisé. Ces filtres peuvent être de type IIR ou FIR.

Le signal estimé s'écrit de la façon suivante :

$$\hat{x}(n) = \sum_{i=0}^{N-1} h_i(n-1)y(n-i) \Leftrightarrow \hat{x}(n) = \mathbf{h}^T(n-1)\mathbf{y} \quad (1.18)$$

Ce qui signifie qu'un point à l'instant  $n$  est calculé en utilisant la réponse impulsionnelle du filtre calculée le coup précédent par l'algorithme d'optimisation. [11]

### 6.3.3.1. Algorithme RLS :

Sachant que les propriétés statistiques nous sont inconnues, on ne va pas chercher à minimiser  $E[e^2(n)]$  mais une somme finie d'erreur au carré donnée par :



$$\xi = \sum_{k=0}^n (x(k) - \hat{x}(k))^2 \quad (1.19)$$

Quand cette fonction coût est minimisée en utilisant une réponse impulsionnelle

$$R_{yy}(n) = \sum_{k=0}^n y(k)y^T(k) \quad (1.20)$$

La réponse impulsionnelle est donc fonction des échantillons disponibles et non pas d'une moyenne statistique générale. Par analogie avec Wiener, elle est donnée par la relation :

$$r_{yx}(n) = \sum_{k=0}^n y(k)x(k) \quad (1.21)$$

### 6.3.3.2 Algorithme LMS :

D'autres techniques d'optimisation basées sur la descente en gradient peuvent être mise en œuvre. L'algorithme LMS (Least Mean Squares) dont le détail est donné ci-dessous est souvent utilisé dans les systèmes de filtrage adaptatif. [7].

Initialisation :

$$h(0)=0$$

Pour chaque échantillon  $n = [1, 2, \dots]$ , Faire

$$\hat{x}(n) = h^T(n-1)y(n) ; \quad e(n) = x(n) - \hat{x}(n)$$

$$h(n) = h(n-1) + 2\mu y(n)e(n)$$

## 7. Les domaines d'applications d'élimination d'une interférence [9]

Il existe de nombreuses applications à l'annulation de bruit. En voici quelques-unes :

**7.1. Domaine de l'aviation :** l'annulation permet aux pilotes d'avion de chasse de communiquer entre eux, malgré le bruit assourdissant qui règne dans le cockpit.

**7.2. Domaine de l'automobile :** les pilotes de course doivent pouvoir communiquer clairement avec les membres de son équipe, sans que le bruit du moteur interfère avec le signal utile (la voix).

**7.3. Domaine des téléphones portables :** kit mains libres sans fils accroché permettant de communiquer clairement, bien que le microphone soit au niveau de l'oreille.

**7.4. Militaire :** le domaine militaire regorge de technologies de pointe, et l'annulation de bruit et l'une des technologies utilisée depuis longtemps. Elle permet aux militaires dans des

situations délicates confrontés à des environnements très bruyants de pouvoir continuer à communiquer entre eux.

**7.5. Sécurité publique** : en cas d'interventions, les pompiers sous leurs masques confrontés à des conditions extrêmes nécessitent ce genre de technologie pour dialoguer.

**7.6. Médical** : lors d'un électrocardiogramme, on obtient un signal qui contient deux informations : le battement du cœur de la mère et celui du fœtus. L'annulation de bruit permet d'annuler le battement de la mère dans le signal pour permettre l'analyse du battement du bébé

## **8. Conclusion**

Dans ce chapitre nous avons présenté les notions de base sur généralité d'interférence. Après un historique sur l'apparition d'annulation d'interférence, et la présentation de la différente méthode de filtrage ainsi que leurs types (bruit, écho). Nous avons terminé ce chapitre par les différents domaines d'applications d'élimination d'une interférence.

Nous nous sommes concentrés principalement dans Le chapitre suivant sur fondements théoriques des réseaux neuronaux qui sont concernées par notre travail de recherche et qui sont généralement les plus utilisées.

*Chapitre II*  
*les réseaux de neurones*  
*artificiels*

## FONDEMENTS THÉORIQUES DES RÉSEAUX NEURONAUX

### 1. Introduction

Le terme réseaux de neurones regroupe une vaste catégorie de modèles d'apprentissage ayant tous en commun le fait de s'inspirer du fonctionnement du système nerveux : tous ces modèles mettent en jeu des unités appelées neurones dont les interactions sont gouvernées par des poids les reliant les synapses. L'apprentissage d'un réseau de neurones consistant donc à faire varier ces poids jusqu'à obtenir les valeurs désirées en sortie.

De part leur robustesse, leur flexibilité et leur grande capacité de généralisation, les réseaux de neurones ont pu s'imposer comme l'un des modèles majeurs en apprentissage machine. Afin de donner au lecteur une bonne appréciation de ce que sont les réseaux de neurones, nous avons décidé de parler de divers modèles. Nous verrons tout d'abord les réseaux à propagation avant à travers le perceptron multicouches.

Ce chapitre est structuré en partie est consacrée à la présentation des réseaux de neurones artificiels. Nous commençons par donner une brève présentation de l'évolution historique de cet axe de recherche, avant de présenter le principe de fonctionnement des neurones artificiels, nous décrivons les bases essentielles des neurones biologiques ainsi que les différentes architectures de réseaux de neurones. Nous concluons cette partie en présentant les réseaux de neurones les plus utilisés..

## 2. Eléments de base des réseaux de neurones

### 2-1 Historique

- 1890 : W. James, célèbre psychologue américain introduit le concept de mémoire associative, et propose ce qui deviendra une loi de fonctionnement pour l'apprentissage sur les réseaux de neurones connue plus tard sous le nom de loi de Hebb. [12]

- 1943 : J. Mc Culloch et W. Pitts laissent leurs noms à une modélisation du neurone biologique (un neurone au comportement binaire). Ceux sont les premiers à montrer que des réseaux de neurones formels simples peuvent réaliser des fonctions logiques, arithmétiques et symboliques complexes (tout au moins au niveau théorique). [12]

- 1949 : D. Hebb, physiologiste américain explique le conditionnement chez l'animal par les propriétés des neurones eux-mêmes. Ainsi, un conditionnement de type pavlovien tel que, nourrir tous les jours à la même heure un chien, entraîne chez cet animal la sécrétion de salive à cette heure précise même en l'absence de nourriture. La loi de modification des

propriétés des connexions entre neurones qu'il propose explique en partie ce type de résultats expérimentaux. [12]

### **Les premiers succès**

- 1957 : F. Rosenblatt développe le modèle du Perceptron [13]. Il construit le premier neuroordinateur basé sur ce modèle et l'applique au domaine de la reconnaissance de formes. Notons qu'à cet époque les moyens à sa disposition sont limités et c'est une prouesse technologique que de réussir à faire fonctionner correctement cette machine plus de quelques minutes.

- 1960 : B. Widrow, un automaticien, développe le modèle Adaline (Adaptative Linear Element). Dans sa structure, le modèle ressemble au Perceptron, cependant la loi d'apprentissage est différente. Celle-ci est à l'origine de l'algorithme de rétropropagation de gradient très utilisé aujourd'hui avec les Perceptrons multicouches. Les réseaux de type Adaline restent utilisés de nos jours pour certaines applications particulières. B. Widrow a créé dès cette époque une des premières firmes proposant neuro-ordinateurs et neuro-composants, la "Memistor Corporation". Il est aujourd'hui le président de l'International Neural Network Society (INNS) sur laquelle nous reviendrons au chapitre Informations pratiques.

- 1969 : M. Minsky et S. Papert publient un ouvrage qui met en exergue les limitations théoriques du perceptron. Limitations alors connues, notamment concernant l'impossibilité de traiter par ce modèle des problèmes non linéaires. Ils étendent implicitement ces limitations à tous modèles de réseaux de neurones artificiels. Leur objectif est atteint, il y a abandon financier des recherches dans le domaine (surtout aux U.S.A.), les chercheurs se tournent principalement vers l'IA et les systèmes à bases de règles. [14]

### **L'ombre**

- 1967-1982 : Toutes les recherches ne sont, bien sûr, pas interrompues. Elles se poursuivent, mais déguisées, sous le couvert de divers domaines comme : le traitement adaptatif du signal, la reconnaissance de formes, la modélisation en neurobiologie, etc. De grands noms travaillent durant cette période tels : S. Grossberg, T. Kohonen, ... dont nous reparlerons.

### **Le renouveau**

- 1982 : J. J. Hopfield est un physicien reconnu à qui l'on doit le renouveau d'intérêt pour les réseaux de neurones artificiels. A cela plusieurs raisons :

Au travers d'un article court, clair et bien écrit, il présente une théorie du fonctionnement et des possibilités des réseaux de neurones. Il faut remarquer la présentation anticonformiste de son article. Alors que les auteurs s'acharnent jusqu'alors à proposer une structure et une loi d'apprentissage, puis à étudier les propriétés émergentes ; J. J. Hopfield fixe préalablement le

comportement à atteindre pour son modèle et construit 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. [15]

D'autre part, entre les mains de ce physicien distingué, la théorie des réseaux de neurones devient respectable. Elle n'est plus l'apanage d'un certain nombre de psychologues et neurobiologistes hors du coup.

Enfin, une petite phrase, placée en commentaire dans son article initial, met en avant l'isomorphisme de son modèle avec le modèle d'Ising (modèle des verres de spins). Cette idée va drainer un flot de physiciens vers les réseaux de neurones artificiels.

Notons qu'à cette date, l'IA est l'objet d'une certaine désillusion, elle n'a pas répondu à toutes les attentes et s'est même heurtée à de sérieuses limitations. Aussi, bien que les limitations du Perceptron mise en avant par M. Minsky ne soient pas levées par le modèle d'Hopfield, les recherches sont relancées.

#### **La levée des limitations**

- 1983 : La Machine de Boltzmann 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 (les temps de calcul sont considérables).

- 1985 : La rétropropagation de gradient apparaît. C'est un algorithme d'apprentissage adapté aux réseaux de neurones multicouches (aussi appelés Perceptrons multicouches). Sa découverte réalisée par trois groupes de chercheurs indépendants indique que "la chose était dans l'air". Dès 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. [16]

- Actuellement les réseaux multicouches et la rétropropagation de gradient reste le modèle le plus étudié et le plus productif au niveau des applications.

### **2-2 Définition**

#### **-Définition 1 :**

Les réseaux de neurones artificiels sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit. Toute structure hiérarchique de réseaux est évidemment un réseau.

**-Définition 2 :**

Le neurone est un élément contenant plusieurs entrées et une sortie. Chaque entrée  $x_i$  est pondérée par un poids  $W_{ki}$  avant d'être sommé pour produire  $u_k$ ,  $k$  étant un indice indiquant le numéro du neurone. Le résultat  $u_k$  est soumis à une fonction de transfert  $j(.)$  normalement non linéaire. Habituellement la sortie de ce neurone  $y_k$  sert d'entrée à un autre neurone. La Figure 2-1 présente le modèle classique d'un neurone. Notez que le poids  $w_{k0}$  sert de biais servant à aligner  $u_k$  avec le zéro de  $j(.)$ .

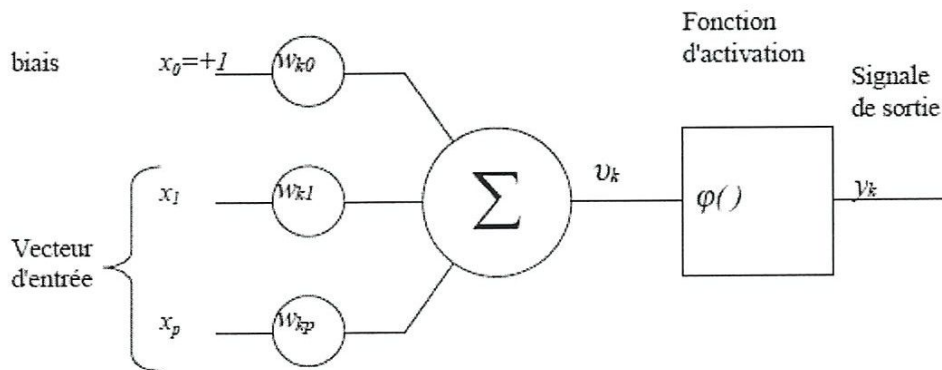


Figure 2-1 : Représentation schématique d'un neurone.

## 2.3 Neurone biologique au modèle mathématique

### 2.3.1 Le modèle neurophysiologique

#### 2.3.1.1 Structure

Le cerveau humain est composé d'environ  $10^{12}$  neurones (mille milliards), avec 1000 à 10000 synapses (connexions) par neurone !

Le neurone est une cellule composée d'un corps cellulaire et d'un noyau. Le corps cellulaire se ramifie pour former ce que l'on nomme les dendrites. Celles-ci sont parfois si nombreuses que l'on parle alors de chevelure dendritique ou d'arborisation dendritique. C'est par les dendrites que l'information est acheminée de l'extérieur vers le soma, corps du neurone. L'information traitée par le neurone chemine ensuite le long de l'axone (unique) pour être transmise aux autres neurones.

La transmission entre deux neurones n'est pas directe. En fait, il existe un espace intercellulaire de quelques dizaines d'Angstrom ( $10^{-9}$  m) entre l'axone du neurone afférent et les dendrites (on dit une dendrite) du neurone efférent. La jonction entre deux neurones est

appelée la synapse. Cet espace vide ne permet pas au signal électrique de se propager. Il existe donc des neuromédiateurs (substances chimiques) qui se fixent sur des récepteurs de l'autre côté de l'espace inter-synaptique. Lorsque suffisamment de molécules sont fixées, un signal électrique est émis. C'est ainsi que le signal se transmet. [17]

Selon le type de synapse, excitatrice ou inhibitrice, l'activité d'un neurone peut renforcer ou diminuer l'activité de ses voisins. (fig. 2.2).

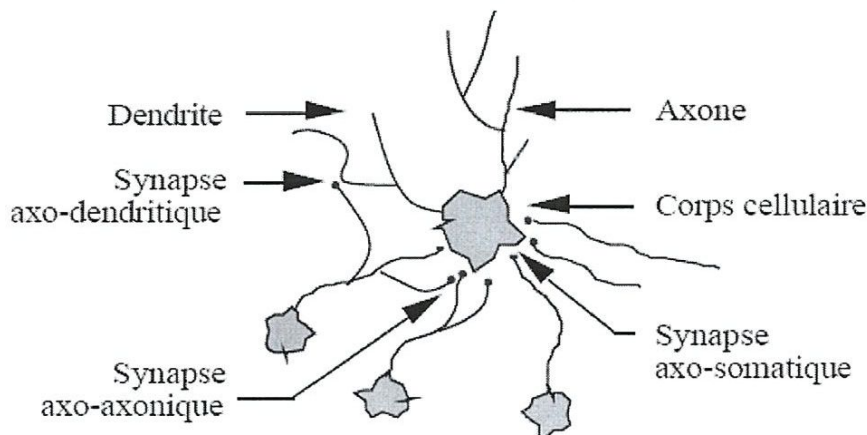


Figure 2.2 Un neurone biologique avec son arborisation dendritique

### 2.3.2 Les modèles mathématiques

Les réseaux de neurones biologiques réalisent facilement un certain nombre d'applications telles que la reconnaissance de formes, le traitement du signal, l'apprentissage par l'exemple, la mémorisation, la généralisation. Ces applications sont pourtant, malgré tous les efforts déployés en algorithmique et en intelligence artificielle, à la limite des possibilités actuelles. C'est à partir de l'hypothèse que le comportement intelligent émerge de la structure et du comportement des éléments de base du cerveau que les réseaux de neurones artificiels se sont développés. Les réseaux de neurones artificiels sont des modèles, à ce titre ils peuvent être décrit par leurs composants, leurs variables descriptives et les interactions des composants. [18]

#### 2.3.2.1 Composant (le neurone artificiel)

##### 2.3.2.1.1 Structure

La figure 2.3 montre la structure d'un neurone artificiel. Chaque neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones amonts. A chacune de ces entrées est associée un poids  $w$  abréviation de "weight" (poids en



anglais) représentatif de la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones avals a chaque connexion est associée un poids.

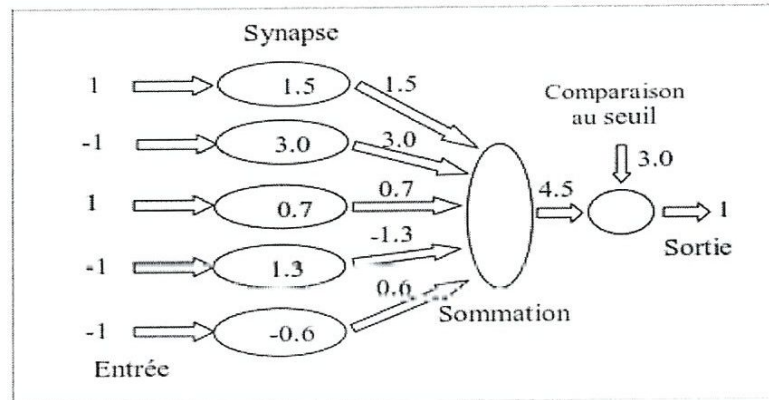


Figure 2.3 : Le neurone formel.

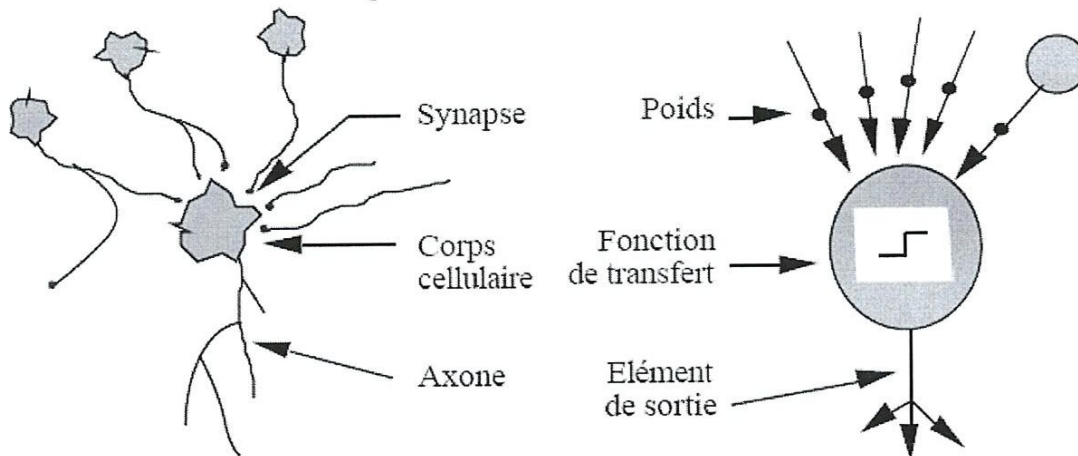


Figure 2.4 Mise en correspondance neurone biologique / neurone artificiel

2.3.2.1.2 Modélisation d'un neurone formel

Les réseaux de neurones formels sont à l'origine d'une tentative de modélisation mathématique du cerveau humain. Les premiers travaux datent de 1943 et sont l'œuvre de MM. Mac Culloch et Pitts. Ils représentent un modèle assez simple pour les neurones et explorent les possibilités de ce modèle.

La modélisation consiste à mettre en œuvre un système de réseau neuronal sous un aspect non pas biologique mais artificiel, cela suppose d'après le principe biologique, on aura

une correspondance pour chaque élément composant le neurone biologique, donc une modélisation pour chacun d'entre eux. [18]

On pourra résumer cette modélisation par le Tableau (2.1), qui nous permettra de voir clairement la transition entre le neurone biologique et le neurone formel.

Neurone biologique	Neurone artificiel
Synapses	Poids de connexions
Axones	Signal de sortie
Dendrites	Signal d'entrée
Noyau	Fonction d'activation

**Tableau 2.1 : Analogie entre le neurone biologique et le neurone formel**

### 2.3.2.1.3 Les entrées

Elles peuvent être :

- Booléennes.
- Binaires (0,1) ou bipolaires (-1,1).
- Réelles.

### 2.3.2.1.4 Fonction d'activation

On distingue deux phases. La première est habituellement le calcul de la somme pondérée des entrées ( $a$ ) selon l'expression suivante :

$$a = \sum(w_i . e_i) \quad (2.1)$$

A partir de cette valeur, une fonction de transfert calcule la valeur de l'état du neurone. C'est cette valeur qui sera transmise aux neurones aval. Il existe de nombreuses formes possibles pour la fonction de transfert. Les plus courantes sont présentées sur la figure 2.5 On remarquera qu'à la différence des neurones biologiques dont l'état est binaire, la plupart des fonctions de transfert sont continues, offrant une infinité de valeurs possibles comprises dans l'intervalle  $[0, +1]$  (ou  $[-1, +1]$ ).

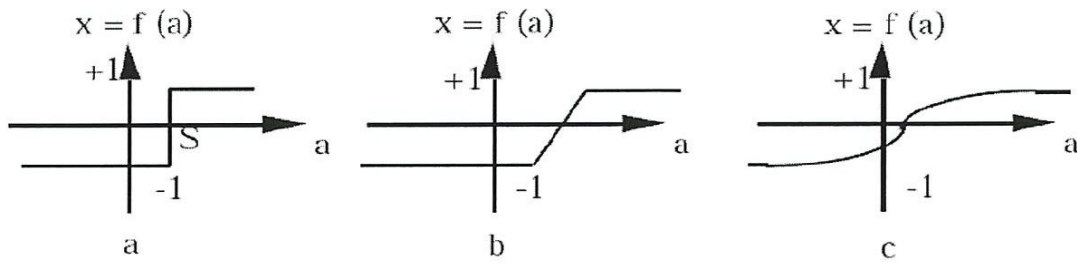


Figure 2.5 Différents types de fonctions de transfert pour le neurone artificiel, a : fonction à seuil (S , la valeur du seuil), b : linéaire par morceaux, c : sigmoïde.

Nous constatons que les équations décrivant le comportement des neurones artificiels n'introduisent pas la notion de temps. En effet, et c'est le cas pour la plupart des modèles actuels de réseaux de neurones, nous avons affaire à des modèles à temps discret, synchrone, dont le comportement des composants ne varie pas dans le temps.

**a. Fonction binaire à seuil**

Fonction Heaviside définie par la fonction (2.2), qui est illustrée par la Figure 2.6.

$$H(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si non} \end{cases} \tag{2.2}$$

Fonction signe définie par la fonction (2.3) et qui est illustrée par la Figure 2.7

$$\text{sgn}(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ -1 & \text{si non} \end{cases} \tag{2.3}$$

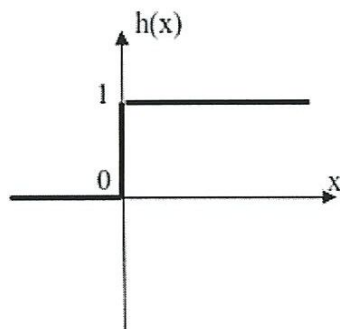


Figure 2.6 : Fonction de Heaviside

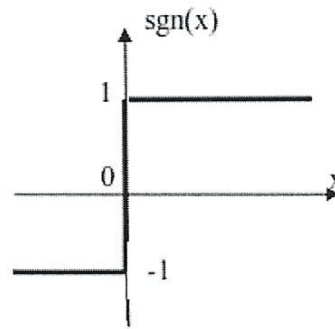


Figure 2.7 : Fonction signe

Le seuil introduit une non-linéarité dans le comportement du neurone, cependant il limite la gamme des réponses possibles à deux valeurs.

### b. Fonction linéaire

C'est l'une des fonctions d'activations les plus simples, sa fonction est définie par :

$$F(x) = x \quad (2.4)$$

Elle est représentée par la Figure 2.8

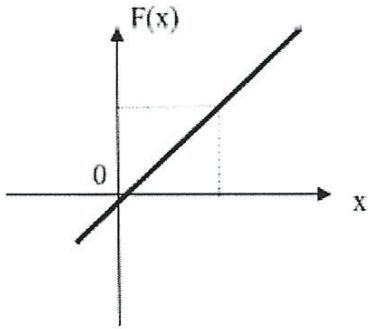


Figure 2.8 : Fonction linéaire.

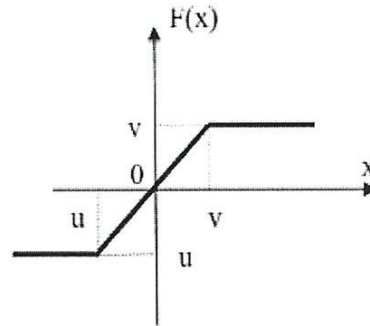


Figure 2.9 : Fonction linéaire à seuil.

### c. Fonction linéaire à seuil ou multi-seuils

On peut la définir comme suit :

$$F(x) = \begin{cases} x & \text{si } x \in [u, v] \\ v & \text{si } x \geq v \\ u & \text{si } x \leq u \end{cases} \quad (2.5)$$

Cette fonction représente un compromis entre la fonction linéaire et la fonction seuil : entre ses deux barres de saturations, elle confère au neurone une gamme de réponses possibles. En modulant la pente de la linéarité, on affecte la plage de neurone. (Figure 2.9)

### d. Fonction sigmoïde

Elle est l'équivalent continu de la fonction linéaire. Étant continue, elle est dérivable, d'autant plus que sa dérivée est simple à calculer, elle est définie par la fonction (2.6) :

$$f(x) = \frac{1}{1+e^{-x}} \quad (2.6)$$

Le tracé de cette fonction est donné par la Figure 2.10

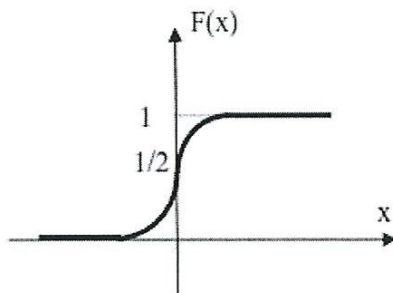


Figure 2.10 : Fonction sigmoïde

#### 2.3.2.1.5 Fonction de sortie

Elle calcule la sortie d'un neurone en fonction de son état d'activation. En général cette fonction est considérée comme la fonction identité. Elle peut être :

- Binaire (0,1) ou bipolaire (-1,1).
- Réelle.

### 2.4 Architecture des réseaux de neurones

Un neurone réalise simplement une fonction non linéaire, paramétrée, de ses entrées. L'intérêt des neurones réside dans les propriétés qui résultent de leur association en réseaux, c'est-à-dire de la composition des fonctions non linéaires réalisées par chacun des neurones.

[19] On distingue essentiellement deux familles de réseau de neurones :

#### 2.4.1. Réseaux de neurones non bouclés :

Un réseau de neurones non bouclé peut donc être imaginé comme un ensemble de neurones « connectés » entre eux, l'information circulant des entrées vers les sorties sans « retour en arrière ». On peut alors représenter le réseau par un graphe acyclique dont les nœuds sont les neurones et les arêtes les « connexions » entre ceux-ci. Si l'on se déplace dans le réseau, à partir d'un neurone quelconque, en suivant les connexions et en respectant leurs sens, on ne peut pas revenir au neurone de départ.

La représentation de la topologie d'un réseau par un graphe est très utile, notamment pour les réseaux bouclés, comme on le verra dans la section « Réseaux de neurones bouclés ». Les neurones qui effectuent le dernier calcul de la composition de fonctions sont les neurones de sortie ; ceux qui effectuent des calculs intermédiaires sont les neurones cachés (voir la Figure 2.11).

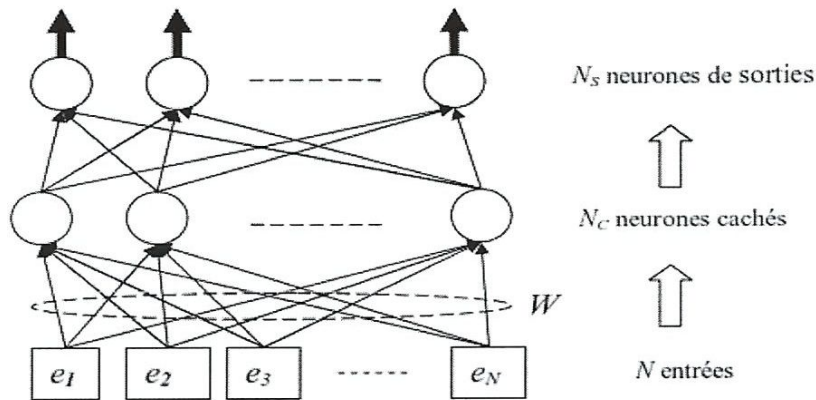


Figure 2.11 : Un réseau de neurones non bouclé à  $N$  entrées,  
Une couche de  $N_c$  neurones cachés et  $N_s$  neurones de sortie.

Un réseau de neurones non bouclé réalise donc une fonction non linéaire de ses entrées paramétrées par les coefficients  $W$  du réseau :

$$Y(k) = (E(k); W) \quad (2.7)$$

Où  $Y(k) \in \mathbb{R}^{N_s}$  est le vecteur de sorties à l'instant  $k$ ,  $E(k) \in \mathbb{R}^N$  est le vecteur des entrées et  $\phi: \mathbb{R}^N \rightarrow \mathbb{R}^{N_s}$  est la fonction non linéaire réalisée par les neurones du réseau.

Le temps ne joue aucun rôle fonctionnel dans un réseau de neurones non bouclé : si les variables sont indépendantes du temps, les sorties le sont également. Le temps nécessaire pour le calcul de la fonction réalisée par chaque neurone est négligeable et, fonctionnellement, on peut considérer ce calcul comme instantané. Pour cette raison, les réseaux non bouclés sont souvent appelés « réseaux statiques », par opposition aux réseaux bouclés ou « dynamiques ».

□ Propriété d'approximation universelle des RNFs

Toute fonction  $\hat{p}(x)$  à valeur bornées, continue ou non, dans un intervalle fermé  $[X_A, X_B]$  peut être approchée, à près, avec la fonction réalisée par un RNF à une couche d'un nombre fini de neurones cachés à fonction d'activation sigmoïde exponentielle. Plus ceci est vrai pour toute fonction d'activation universelle, c'est-à-dire toute fonction non polynomiale. En particulier, les fonctions dont les limites en  $-8$  et  $+8$  sont finies et distinctes, dites sigmoïdales, sont universelles. La précision peut toujours être améliorée en augmentant le nombre de neurones

cachés. C'est ce que l'on appelle la propriété d'approximation universelle des RNFs (Hornik et al, 1994).

#### 2.4.2 Réseaux de neurones bouclés :

L'architecture la plus générale, pour un réseau de neurones, est celle des « réseaux bouclés », dont le graphe des connexions est cyclique : lorsque l'on se déplace dans le réseau en suivant le sens des connexions, il est possible de trouver au moins un chemin qui revient à son point de départ (un tel chemin est désigné sous le terme de « cycle »). La sortie d'un neurone du réseau peut donc être fonction d'elle-même ; ceci n'est évidemment concevable que si la notion de temps est explicitement prise en considération.

Ainsi, à chaque connexion d'un réseau de neurones bouclé (ou à chaque arête de son graphe) est attaché, outre un paramètre comme pour les réseaux non bouclés, un retard, multiple entier (éventuellement nul) de l'unité de temps choisie. Une grandeur, à un instant donné, ne pouvant pas être fonction de sa propre valeur au même instant, tout cycle du graphe du réseau doit contenir au moins une arête dont le retard n'est pas nul.

Un réseau de neurones bouclé à temps discret réalise une (ou plusieurs) équation(s) aux différences non linéaires, par composition des fonctions réalisées par chacun des neurones et des retards associés à chacune des connexions.

A la fin des années 1980, les RNFs bouclés ont commencé à être utilisés comme modèles dynamiques de processus, et comme filtres non linéaires.

Tout réseau de neurones bouclés est un système dynamique non linéaire que l'on peut mettre sous la forme d'une représentation d'état faisant intervenir un (ou des) réseau(x) de neurones non bouclé(s) :

$$\begin{cases} x(k+1) = f(x(k), E(k), W) \\ Y(k) = g(x(k), E(k), W) \end{cases} \quad (2-8)$$

Où  $k$  est l'instant discret,  $E(k)$  est le vecteur des entrées externes du système,  $Y(k)$  est le vecteur des sorties, et  $X(k)$  est le vecteur des variables d'état, à l'instant  $k$ . Les fonctions  $f$  et  $g$  sont réalisées par les neurones des réseaux, interconnectés avec les paramètres. Cette représentation d'état est appelée forme canonique, et est particulièrement utile pour une formulation simple des algorithmes d'apprentissage. Elle est illustrée par la Figure (2.12)

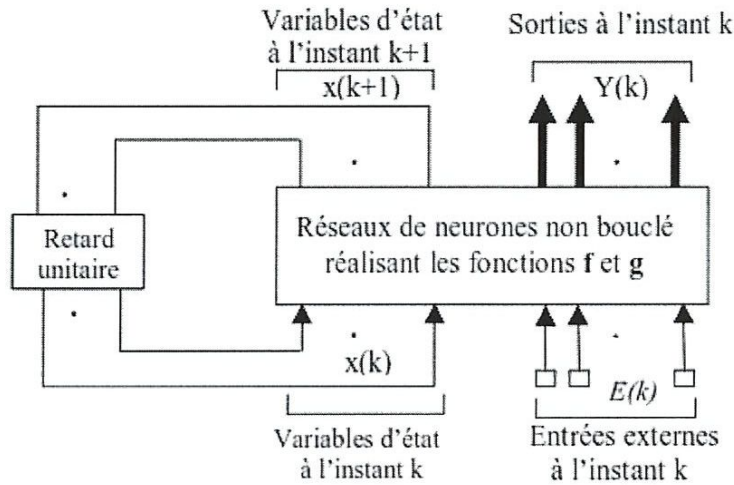


Figure 2.12 : Forme canonique d'un réseau de neurones bouclé

2.4.2.1. Propriétés des réseaux bouclés

Nous présentons ici la propriété d'approximation universelle pour les réseaux bouclés. Considérons le modèle d'état suivant, supposé représenter exactement le comportement dynamique d'un processus :

$$\begin{cases} x_p(k + 1) = \phi(x_p(k), E(k)) \\ Y_p(k) = \gamma(x_p(k), E(k)) \end{cases} \quad (2-9)$$

Où  $E(k)$  est le vecteur des entrées externes du modèle,  $Y_p(k)$  est le vecteur des sorties,  $x_p(k)$  est le vecteur des variables d'état,  $\phi$  et  $\gamma$  sont des fonctions vectorielles non linéaires. Comme il est possible d'approcher n'importe quelle fonction avec un réseau de neurones non bouclé, on peut espérer qu'un modèle de type boucle. Puisse être approché par un réseau bouclé,

Les fonctions  $\phi$  et  $\gamma$  étant réalisées par des réseaux non bouclés.

Plus précisément, la propriété d'approximation universelle pour les réseaux bouclés Peut s'énoncer de la manière suivante :

- ✓ Pour tout système dynamique
- ✓ Pour toute précision désirée, sur une séquence de taille finie  $[0; T]$ ,
- ✓ Pour des entrées bornées  $\{E(k)\}$  appartenant à  $E \subset \mathbb{R}^{N_E}$  et un état initial  $x_p(0) \in X \subset \mathbb{R}^{N_x}$ .



Il existe un réseau de neurones bouclé qui approche le comportement entrée sortie du système avec la précision sur l'intervalle  $[0 ; T]$  et sur les ensembles  $\mathbb{E}$  et  $\mathbb{X}$ . Cette propriété n'est donc pas globale, mais relative à la séquence de taille finie et à l'état initial considéré : elle ne garantit pas à elle seule que l'approximateur possède des caractéristiques dynamiques du processus qu'il est censé approcher, sa stabilité par exemple.

A cette réserve près, les RNFs se prêtent potentiellement bien à la réalisation de modèles dynamiques

Plusieurs motivations peuvent pousser l'ingénieur ou le chercheur à concevoir un modèle dynamique :

- utiliser le modèle comme « simulateur » pour prévoir l'évolution d'un processus dont la modélisation de connaissance est trop complexe ou trop incertaine ;
- utiliser le modèle comme simulateur d'un processus dont la modélisation de connaissance est possible, mais conduit à des équations différentielles, ou aux dérivées partielles, dont la résolution numérique est lourde et ne peut répondre à des contraintes de fonctionnement en temps réel : on peut alors créer un ensemble d'apprentissage, et concevoir un réseau de neurones qui fournit de très bonnes solutions dans des temps de calcul beaucoup plus courts. L'architecture de ce réseau peut avantageusement être inspirée des équations différentielles du modèle de connaissance : on conçoit alors un « modèle semi physique » ou modèle « boîte grise » ;
- utiliser le modèle comme prédicteur à très court terme (une période d'échantillonnage) afin de l'intégrer à un système de commande.

Par conséquent, le réseau de neurones récurrent peut être utilisé comme un modèle simulateur, ou comme un modèle prédicteur à un pas :

□ *Les modèles de simulation :*

Un modèle de simulation peut prédire la sortie du processus qu'il représente à long terme, il est utilisé de manière indépendante du processus, pour décrire une dynamique aussi proche que possible à celle de ce dernier. De tels modèles sont utilisés pour valider la conception d'un système avant sa fabrication, pour la prévision à long terme...etc.

Dans ce cas, le modèle neuronal ne peut pas utiliser les sorties mesurées sur le processus comme des entrées, en revanche, ses entrées seront les sorties prédites par lui-même aux instants précédents, ce qui fait que l'estimation des paramètres de tels modèles est non adaptatif [20].

□ *Les modèles prédicteur à un pas :*

Un prédicteur à un pas est utilisé en parallèle avec le processus dont il est le modèle, en effet une partie des entrées du modèle sont les sorties mesurées du processus. Il prédit la sortie du processus qui vient juste après un pas d'échantillonnage ; il est ainsi par exemple possible de prévoir la sortie du processus à l'instant  $k+1$  connaissant les sorties de ce dernier à l'instant  $k$  [21]

## 2.5 Variables descriptives

Ces variables décrivent l'état du système. Dans le cas des réseaux de neurones qui sont des systèmes non autonomes, un sous-ensemble des variables descriptives est constitué par les variables d'entrée, variables dont la valeur est déterminée extérieurement au modèle.

## 2.6 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.

### 1. Réseau multicouche (au singulier) :

Les neurones sont arrangés par couche. Il n'y a pas de connexion entre neurones d'une même couche et les connexions ne se font qu'avec les neurones des couches avales (fig. 2.13).

Habituellement, chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et celle-ci seulement. Ceci nous permet d'introduire la notion de sens de parcours de l'information (de l'activation) au sein d'un réseau et donc définir les concepts de neurone d'entrée, neurone de sortie. Par extension, on appelle couche d'entrée l'ensemble des neurones d'entrée, couche de sortie l'ensemble des neurones de sortie. Les couches intermédiaires n'ayant aucun contact avec l'extérieur sont appelés couches cachées.

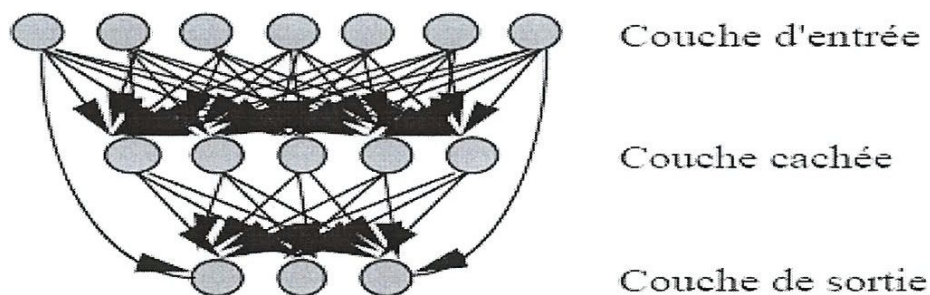


Figure 2.13 Définition des couches d'un réseau multicouche.

### 2. Réseau à connexions locales :

Il s'agit d'une structure multicouche, mais qui à l'image de la rétine, conserve une certaine topologie. Chaque neurone entretient des relations avec un nombre réduit et localisé de neurones de la couche avale (fig.2.14). Les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique.

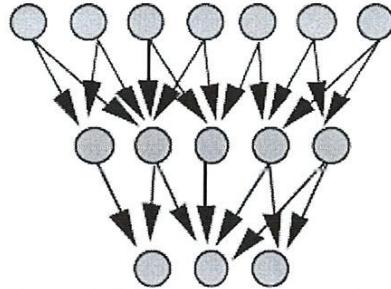


Figure 2.14 Réseau à connexions locales

### 3. Réseau à connexions récurrentes :

Les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche. Ces connexions sont le plus souvent locales (fig. 2.15).

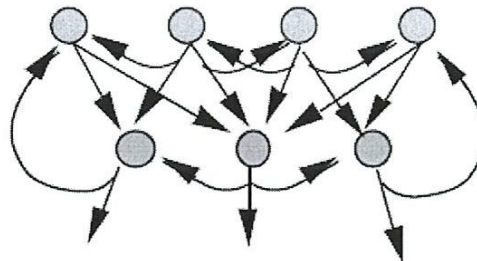


Figure 2.15 Réseau à connexions récurrentes

### 4. Réseau à connexion complète :

C'est la structure d'interconnexion la plus générale (fig.2.16). Chaque neurone est connecté à tous les neurones du réseau (et à lui-même).

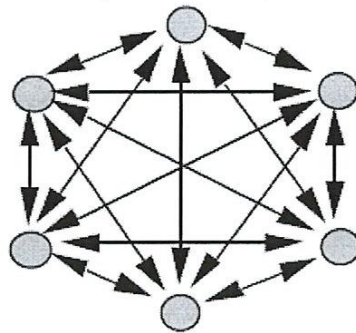


Figure 2.16 Réseau à connexions complète

Il existe de nombreuses autres topologies possibles, mais elles n'ont pas eu à ce jour la notoriété des quelques-unes que nous avons décrites ici.

### **2.7 Les types d'apprentissage des réseaux de neurones**

L'apprentissage est vraisemblablement la propriété la plus intéressante des réseaux de neurones. Elle ne concerne cependant pas tous les modèles, mais uniquement les plus utilisés. L'apprentissage est une phase du développement du réseau de neurones durant laquelle on calcule les poids des neurones de telle manière que les sorties du réseau soient aussi proches que possible des sorties désirées. Les procédures d'apprentissage sont divisées, elles aussi, en deux grandes catégories : apprentissage supervisé et apprentissage non supervisé [22]

#### **2.7.1 L'apprentissage supervisé**

L'apprentissage supervisé implique l'existence d'un professeur qui a pour rôle d'évaluer le succès (ou l'échec) du réseau quand il lui est présenté un stimulus connu (on dit que ce stimulus est un exemple appartenant à la base d'apprentissage). Cette supervision consiste à renvoyer au réseau une information lui permettant de faire évoluer ses connexions (parfois aussi sa propre architecture) afin de faire diminuer son taux d'échec. L'information peut être explicite, sous la forme d'une mesure de l'erreur commise, par exemple, ou globalement sur l'ensemble des exemples de la base. C'est ce qui se passe dans le cas du Perceptron. Mais elle peut être plus implicite (apprentissage par renforcement), sous forme d'une simple appréciation (bon ou mauvais, punition ou récompense), sans mesure d'erreur, et même être globale, sur l'ensemble des tâches que le réseau doit exécuter. Dans ce dernier cas, il est facile de comprendre que l'apprentissage est le plus difficile, la difficulté majeure consistant pour le réseau à identifier les étapes du processus qui sont responsables de l'échec ou du succès.

#### **2.7.2 L'apprentissage non supervisé**

L'apprentissage non supervisé implique la fourniture à un réseau autonome d'une quantité suffisante d'exemples contenant des répétitions (autrement dit de la redondance), telles que celui-ci en dégage les régularités automatiquement. Ces réseaux sont souvent appelés auto-organiseurs, ou encore à apprentissage compétitif. Dans l'apprentissage non supervisé, les données ne contiennent pas d'informations sur une sortie désirée, il n'y a pas de superviseur ou expert humain. Il s'agit de déterminer les paramètres du réseau de neurones suivant un critère à définir. Dans ce cas, les exemples présentés à l'entrée provoquent une auto adaptation du réseau afin de produire des valeurs de sortie qui soient proches en réponse pour des valeurs

d'entrées similaires. Ce type d'apprentissage possède souvent moins de complexité dans le calcul par rapport à l'apprentissage supervisé. Bien entendu, l'architecture du réseau, préalablement définie par son utilisateur, est une forme de supervision

### 3. Les réseaux de neurones les plus utilisés

La Figure (2.17) représente les réseaux de neurones les plus utilisés dans la littérature. Dans ce qui suit, nous allons faire une description de chacun d'eux

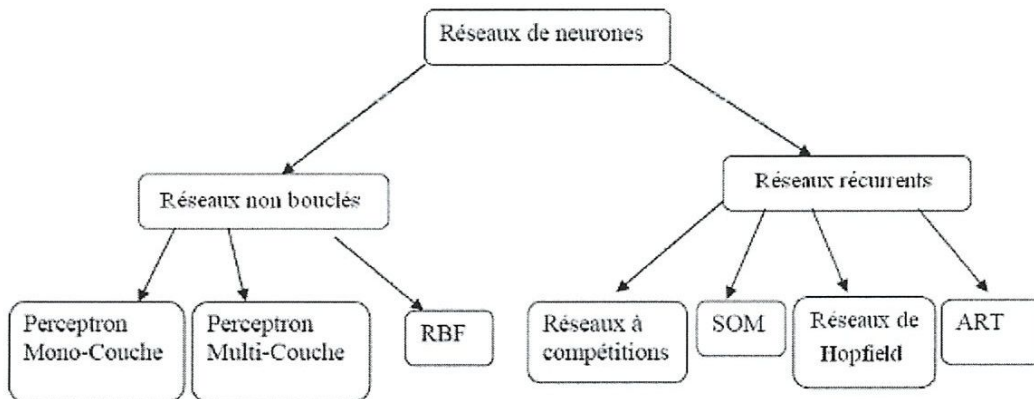


Figure 2.17 Réseaux de neurones le plus utilisés.

- Les Perceptrons
- Les réseaux à fonction radiale
- Les cartes auto-organisatrices de Kohonen (SOM)
- Les réseaux de Hopfield
- Les réseaux ART
- Réseaux de neurones linéaires adaptatifs(Adaline)

### 4. Les différentes applications de RNA

Les réseaux de neurones sont utilisés dans plusieurs domaines, citons la classification, la reconnaissance de formes, l'identification et la commande de processus, prédiction de signal, élimination d'interférence. Le choix d'utiliser tel ou tel type de réseau de neurones dépend de l'application mais aussi des capacités de traitement du processeur sur lequel ils s'exécutent.

#### **4.1 Reconnaissance des formes**

C'est un domaine privilégié d'application pour les RNA et c'est lui qui a marqué leur début. Le premier modèle dans cette thématique a été développé aux laboratoires AT&Bell dans la reconnaissance des codes postaux [23].

Le terme de reconnaissance est un terme général qui désigne les processus traitant des données pour en extraire des informations afin de parvenir à leur classification. Une partie non négligeable des applications neuronales actuelles appartient à cette catégorie, parmi lesquelles nous pouvons citer [24, 25]:

- Reconnaissance des formes.
- Reconnaissance des caractères.
- Classification et compression d'images.
- Diagnostic des pannes.

#### **4.2 Modélisation**

L'une des applications répondu des RNA est la modélisation, nous pouvons citer comme exemples : Modélisation financière, modélisation des processus de fabrication et de production, modélisation en biomédecine [26].

#### **4.3 Traitement de la parole**

Le traitement de la parole est une discipline scientifique localisée au croisement du traitement du signal numérique et du traitement du langage. Depuis une vingtaine d'années, les réseaux de neurones artificiels constituent une technique utilisée dans les systèmes de traitement automatique de la parole. Ils peuvent être employés à de nombreux niveaux dans un système de traitement automatique de la parole. De nombreuses études ont été menées pour les utiliser pour le traitement du signal parole (filtrage, annulation d'échos, séparation de sources), la modélisation acoustique mais aussi pour des tâches de plus haut niveaux telles que la modélisation linguistique [24, 27].

#### **4.4 Détection d'anomalies en médecine**

Ceci est une dérivation des reconnaissances de formes. On apprend à un réseau une image du fonctionnement normal d'un système et celui-ci sera ainsi capable d'indiquer tout état de dysfonctionnement quand certains paramètres engendrent une image anormale. Parmi les applications en médecine nous pouvons citer :

- Cardiologie [28]
- Traitement des images MRI (Magnetic Resonance Imager) [29]
- Psychologie [30], psychiatrie [31] et sociologie [32]
- Ophtalmologie [33]

#### **4.5 Traitements dépendant du temps**

Dans ce cas, on se retrouve devant des problèmes de prédiction, d'identification et de commande de processus

- Prédiction [34]
- Identification et commande de processus [35].

### **5. Conclusion**

Ce chapitre introduit les notions de base sur les réseaux de neurones artificiels. Après un historique sur l'apparition des RNA, et la présentation des modèles biologique et mathématique du neurone, les différentes architectures des réseaux de neurones ainsi que leurs types d'apprentissage sont présentés. Les RNA les plus utilisés sont aussi exposés. Nous avons terminé ce chapitre par les différentes applications des réseaux de neurones artificiels.

Nous nous sommes concentrés principalement dans Le chapitre suivant sur deux architectures neuronales qui sont concernées par notre travail de recherche et qui sont généralement les plus utilisées pour la Suppression d'interférence : le Réseaux de neurones linéaires adaptatifs " type Adalin " et la rétropropagation de l'erreur.

*Chapitre III*  
*Partie application*



**1.Introduction :**

L'élimination d'une interférence constitue aujourd'hui une part importante du traitement du signal, son principal intérêt est d'extraire un signal dit utile (ou informatif) d'un signal bruité ou en interférence avec d'autres signaux de même nature. Soit  $S_b$  constitué du signal utile  $S_u$  auquel est superposé une interférence  $S_i$  (signal non utile) tel que  $S_b = S_u + k \cdot S_i$ .  
 Pour cela on a recouru aux techniques de filtrages et aux techniques de l'intelligence artificielle telles que les réseaux de neurones.

**2. Loi d'apprentissage**

L'apprentissage du réseau consiste à modifier, à chaque pas, les poids et les biais afin de minimiser la somme des carrés des erreurs en sortie en utilisant la loi de Widrow-Hoff.

À chaque pas d'apprentissage, l'erreur en sortie est calculée comme la différence entre la cible recherchée  $t$  et la sortie  $y$  du réseau.

La quantité à minimiser, à chaque pas d'apprentissage  $k$ , est la variance de l'erreur en sortie du réseau.

$$E_k = e_k^T e_k = (t_k - y_k)^T (t_k - y_k) = \frac{1}{2} (t_k^T t_k + y_k^T y_k - 2y_k^T t_k) \quad (3.1)$$

Le gradient de cette quantité par rapport à la matrice de poids  $W$  est donné par :

$$\nabla E_{k/W} = \frac{1}{2} \nabla [y_k^T y_k - 2y_k^T t_k] / W \quad (3.2)$$

Ce gradient peut se calculer comme suit :

$$\nabla E_{k/W} = \frac{\partial E_k}{\partial W} = \frac{\partial E_k}{\partial y_k} \frac{\partial y_k}{\partial W} \quad (3.3)$$

D'après l'expression de  $E_k$ , et avec  $y_k = Wx_k + b$ , les dérivées partielles sont :

$$\frac{\partial E_k}{\partial y_k} = y_k - t_k \quad (3.4)$$

$$\frac{\partial (Wx_k + b)}{\partial W} = x_k^T \quad (3.5)$$

La mise à jour se faisant dans le sens inverse du gradient, la matrice de poids  $W$  de l'étape future ( $k+1$ ) est :

$$W(k+1) = W(k) - \eta \nabla E_{k/W} = W(k) + \eta (t_k - y_k) x_k^T \quad (3.6)$$

Avec  $\eta$  le gain d'apprentissage.

De même, on obtient l'expression de la modification du biais :

$$b(k+1) = b(k) - \eta \nabla_b E_k = b(k) + \eta(t_k - y_k) \quad (3.7)$$

Dans le cas général, un réseau de type Adaline possède une seule couche de  $S$  neurones. Si le vecteur d'entrée est de taille  $R$ , la matrice de poids  $W$  est de dimensions  $(S, R)$ . A chaque pas d'apprentissage, on calcule l'erreur entre la sortie  $y = Wx + b$  et le vecteur cible  $t$  pour servir dans la règle d'apprentissage.

### 3. Algorithme de rétro-propagation du gradient

L'algorithme d'apprentissage par rétro-propagation du gradient de l'erreur est un algorithme itératif de premier ordre qui a pour objectif de trouver les poids des connexions minimisant l'erreur quadratique moyenne (2.21), commise par le réseau sur l'ensemble d'apprentissage.

Cet algorithme d'apprentissage du réseau MLP non bouclé, peut se résumer comme suit:

1. initialiser l'ensemble des vecteurs de poids  $W_{ji}(0)$ ,  $W_{mj}(0)$  et le paramètre d'apprentissage  $\eta$  à des valeurs aléatoires,
2. appliquer le vecteur  $x_i(k)$  en entrée du réseau,
3. calculer la sortie  $y_m(k)$  donnée par la relation (2.9),
4. calculer l'erreur  $e_m(k) = yd(k) - y_m(k)$
5. calculer l'ensemble des nouveaux poids  $W_{mj}(k+1)$ ,  $W_{ji}(k+1)$
6. incrémenter  $k \rightarrow k+1$  et aller à l'étape 2.

L'utilisation de l'algorithme du gradient peut conduire à des blocages en cas d'un minimum local. Ainsi son efficacité dépend, en effet, d'un grand nombre de paramètres que doit fixer l'utilisateur : le pas du gradient, le paramètre d'apprentissage  $\eta$ , les paramètres des fonctions d'activations, l'initialisation des poids et les biais, l'architecture du réseau, le nombre de neurones par couche, etc.

En revanche, l'algorithme du gradient est très efficace loin du minimum local, lorsque la norme du gradient a une grande valeur. Toutefois, la convergence devient particulièrement lente lorsque le vecteur paramètre  $W$ , s'approche d'un minimum : on a alors intérêt à utiliser un algorithme du (quasi) second ordre (Personnaz et Rivals, 2003)

#### 3.1 La première étape

Dans une première étape, le réseau propage l'information, de la couche d'entrée à la première couche cachée et ensuite à la couche suivante jusqu'à la couche de la sortie.

Fixons les notations suivantes :

$N$  : nombre d'entrées du réseau.

$N_h$  : nombre d'unités sur la couche cachée du réseau.

$N_{output}$  : nombre d'unités sur la couche de sortie.

L'entrée nette de chaque unité cachée est donnée pour chaque p stimulus par la relation :

$$net_{pj} = \sum_i w_{ji} \cdot X_i + \theta_j \quad (3.8)$$

La sortie de chaque unité cachée est une fonction de la somme pondérée des composantes du vecteur d'entrée x et du seuil  $\theta$  :

$$o_j = f(\sum_{i=1}^{N_{input}} W_{ji} \cdot X_i + \theta_j) \quad (3.9)$$

où f est en général soit une fonction sigmoïde soit la fonction tangente hyperbolique.

Dans le cas d'un réseau muni d'une seule couche cachée, la sortie  $o_i$  du neurone k de la couche de sortie du réseau s'exprime en fonction de l'information venant de la couche cachée par :

$$o_k = f(\sum_{i=1}^{N_h} W_{kj} \cdot O_j + \theta_k) \quad \text{ou} \quad o_k = \sum_{i=1}^{N_h} W_{ij} \cdot O_j + \theta_k \quad K = 1, \dots, N_{output}$$

La fonction totale réalisée par le réseau sur la sortie i s'écrit ainsi :

$$O_k = f(\sum_{i=1}^{N_h} W_{kj} f(\sum_{i=1}^{N_{input}} W_{ji} \cdot X_i + \theta_j) + \theta_k) \quad (3.10)$$

Les fonctions réalisées par les neurones d'une même couche sont du même type, seuls les vecteurs poids et les seuils changent d'une unité à l'autre.

En comparant la réponse courante du chaque neurone de la sortie  $O_k$  avec la réponse désirée  $d_{pk}$  et en tenant compte de l'état d'activation on évalue une erreur  $\delta_{pk}$  :

$$\delta_{pk} = (d_{pk} - o_{pk}) \cdot f'(net_{pk}) \quad (3.11)$$

où f' est la dérivée de la fonction d'activation du neurone.

### 3.2 La deuxième étape

Pendant la deuxième étape l'erreur entre la sortie courante et la réponse désirée se propage de la sortie à l'entrée en générant la modification de l'intensité des connexions. La règle d'apprentissage pour les connexions entre la dernière couche cachée et la sortie est :

$$\Delta_p w_{kj} = \eta \cdot \delta_{pk} \cdot o_{pj} \quad (3.12)$$

où  $\eta$  est la constante d'apprentissage, qui peut avoir une valeur appartenant à l'intervalle (0,1).

Pour les neurones cachés (indexés par  $j$ ) l'erreur  $\delta_{pj}$  se calcule avec la relation suivante :

$$\delta_{pj} = \left( \sum_k \delta_{pk} \cdot w_{kj} \right) \cdot f'(net_{pj}) \quad (3.12)$$

Suivant, on modifie l'intensité des connexions entre l'entrée est la couche cachée avec une règle d'apprentissage (qui peut être la même)

$$\Delta_p w_{ji} = \eta \cdot \delta_j \cdot o_{pi} \quad (3.13)$$

S'il y a plusieurs couches cachées, pour chaque unité de la couche cachée l'erreur s'évalue avec la relation (3.13). Les erreurs se propagent successivement, de couche à couche, de la sortie vers l'entrée.

Dans un réseau de neurones peuvent coexister aussi des connexions avec une intensité fixe. S'il y a des unités de sortie aussi dans les couches cachées, celles-ci ont deux types des erreurs. Il y a une erreur obtenue en comparant la sortie courante avec la réponse désirée et une erreur déterminée par la propagation des erreurs des unités de la sortie couplées avec l'unité cachée.

L'algorithme de rétro-propagation de l'erreur minimise le carré de l'erreur dans chaque itération. On peut démontrer que l'algorithme de rétro-propagation de l'erreur implémente un gradient d'erreur carrée descendant dans l'espace des poids :

$$w(t + 1) = w(t) - \eta \cdot \frac{dE}{dw} \quad (3.14)$$

Où  $\frac{dE}{dw}$  est le gradient aléatoire inconnu d'erreur de paire  $(x_i, t_i)$

#### 4. Application des réseaux de neurones multicouches

Les RN sont des approximateur universels.

Jusqu'au présent la rétro- propagation de l'erreur a été utilisée dans un très grand nombre des applications, comme par exemple :

- la détection et la classification des signaux radar
- filtrage adaptatif de chenal de communication
- analyse de courbes, classification de formes (EEG, diagrammes) ;
- la reconnaissance de chiffres et autres symboles écrits ;
- la reconnaissance de la parole (des phonèmes) ;
- la lecture des textes (Guyon et al.) Le succès de la méthode a permis de fournir une des premières applications industrielles des réseaux de neurones ;
- machine a prononcer l'anglais. Sejnovski et Rosenberg ont entraîné un réseau de neurones, baptisé NET-TALK à prononcer l'anglais à partir d'un texte écrit. Le

réseau a été composé de 203 unités d'entrée, d'une couche cachée de 80 neurones et d'une couche de sortie de 30 unités. Ils ont montré que l'histoire d'apprentissage d'un réseau de neurones ressemble à l'apprentissage humain. Le réseau a commencé par être incapable de prononcer de sons cohérents, puis il a babillé et finalement, après un temps assez long, il a prononcé les mots correctement, intelligiblement

- le modelage d'un système inconnu et le contrôle d'un système dynamique ;
- le contrôle d'un bras mobile du contrôle d'un véhicule autonome ;
- analyse financière
- aide à la prise de décision.

## 5 Réseaux de neurones linéaires adaptatifs

### 5.1. Architecture

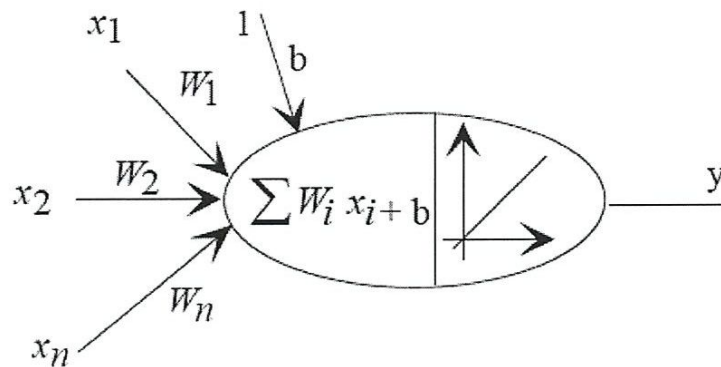


Figure.3.1 : Réseaux de neurones linéaires adaptatifs

Le neurone de type Adaline réalise une somme pondérée des signaux qu'il reçoit, à laquelle il ajoute un biais. Cette activation passe à travers une fonction de transfert linéaire.

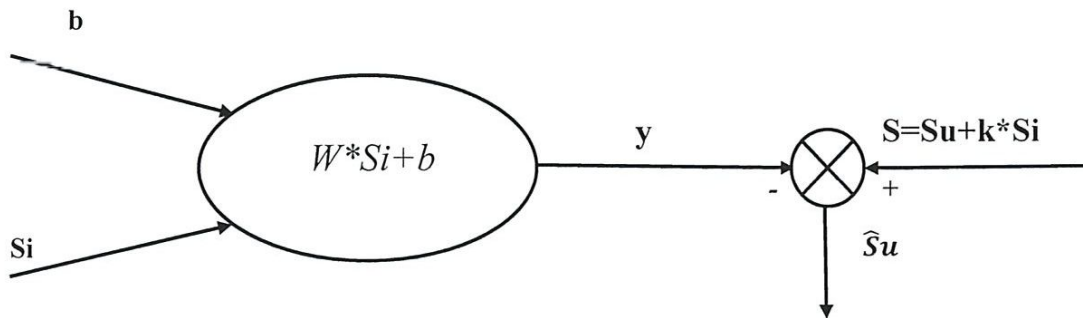
6-Partie Application :

6-1 Elimination d'interférence par réseaux de neurones (type Adaline) :

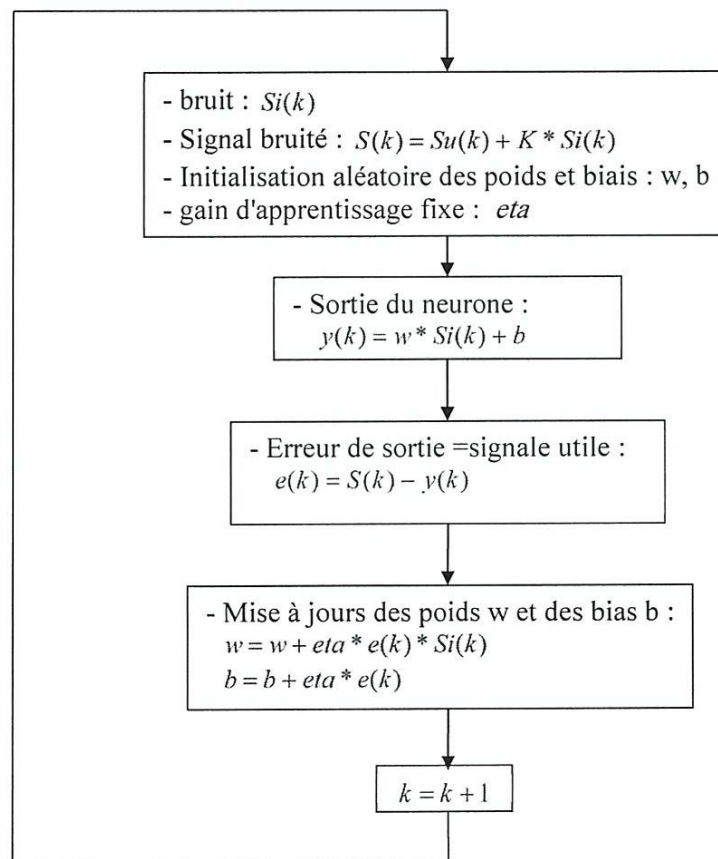
Le principe consiste à présenter à l'entrée du neurone linéaire le bruit  $p$  afin de reconstituer le signal bruité.

Comme le neurone est linéaire, il cherchera la partie du signal cible Proportionnelle au bruit  $p$  : c'est l'interférence.

L'erreur entre la cible et la sortie du neurone constituera le signal utile  $x$  que l'on cherche à extraire.



6-1-1 Algorithme :



a- Cas monodimensionnel :

a.1- Signal bruitée :

- Résultat du programme d'élimination d'une interférence :

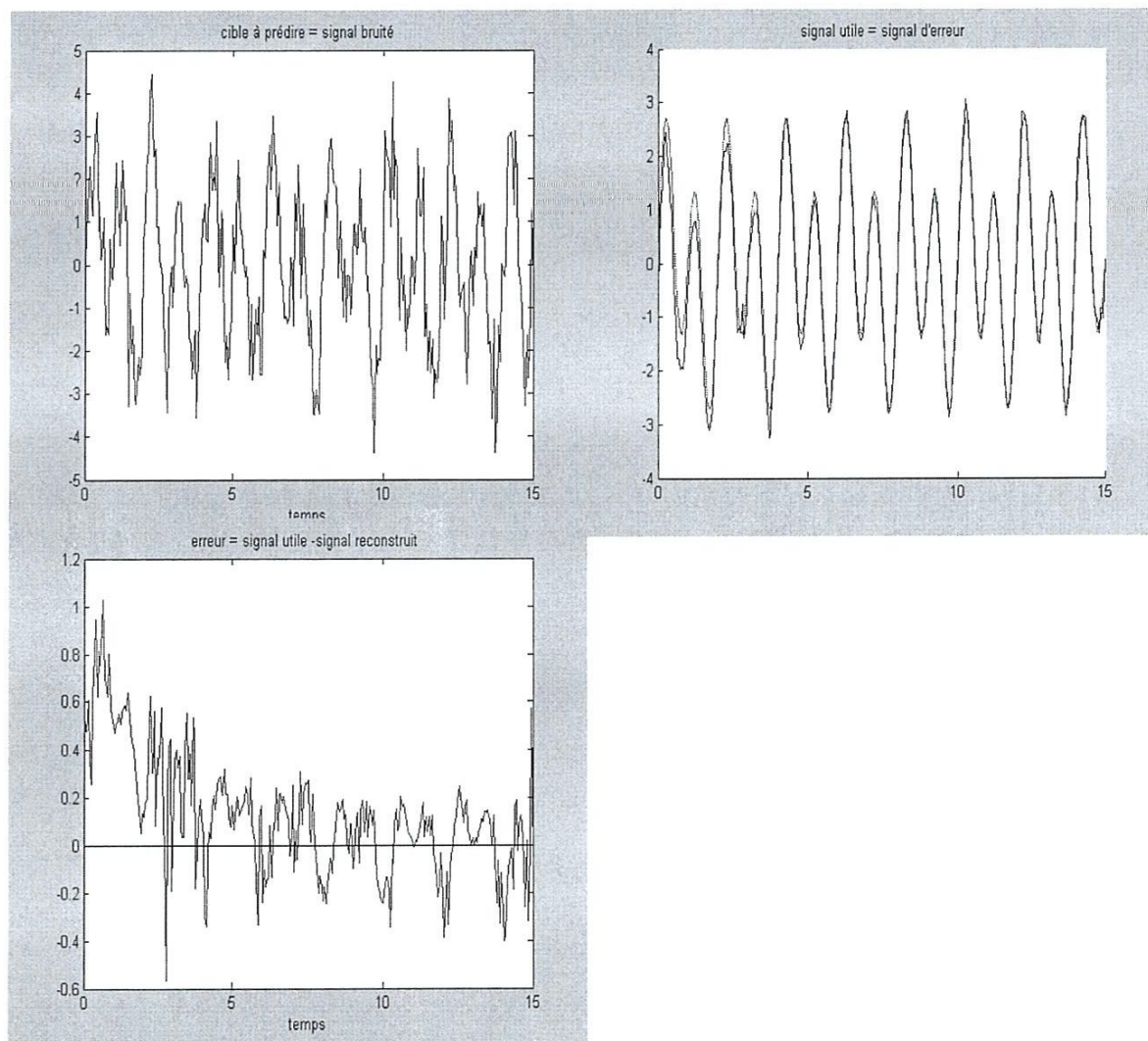


Figure 3.2 : Elimination d'une interférence pour Signal bruitée (Signal monodimensionnel)

b- Cas bidimensionnel (image à niveau de gris) :

b.1- Image bruitée :

- Résultat du programme d'élimination d'une interférence :



Figure 3.3 : Elimination d'une interférence pour une image bruitée (Signal bidimensionnel).

## 7. Conclusion :

l'utilisation des réseaux de neurone dans notre cas un neurone linéaire (type adaline) nous a permis d'extraire notre signal (cas d'un signal monodimensionnel ou cas d'une image signal bidimensionnel) utile d'un signal bruité et cela en atténuant l'effet du bruit, les résultats obtenu paraissent satisfaisantes



## *Conclusion general*

**Conclusion général :**

Dans ce projet nous avons présenté les notions de base sur généralité d'interférence. Après un historique sur l'apparition d'annulation d'interférence, et la présentation de la différente méthode de filtrage ainsi que leurs types (bruit, écho). Nous avons terminé par les différents domaines d'applications d'élimination d'une interférence.

Et après on a introduit les notions de base sur les réseaux de neurones artificiels. Après un historique sur l'apparition des RNA, et la présentation des modèles biologique et mathématique du neurone, les différentes architectures des réseaux de neurones ainsi que leurs types d'apprentissage sont présentés. Les RNA les plus utilisés sont aussi exposés. Nous avons terminé par les différentes applications des réseaux de neurones artificiels.

Nous nous sommes concentrés principalement sur deux architectures neuronales qui sont concernées par notre travail de recherche et qui sont généralement les plus utilisées pour la Suppression d'interférence : le Réseaux de neurones linéaires adaptatifs " type Adalin " et la rétropropagation de l'erreur.

L'utilisation des réseaux de neurone dans notre cas un neurone linéaire (type adaline) nous a permis d'extraire notre signal (cas d'un signal monodimensionnel ou cas d'une image signal bidimensionnel ) utile d'un signal bruité et cela en atténuant l'effet du bruit, les résultats obtenu paressent satisfaisantes

## *Références bibliographiques*

## Références bibliographiques

- [1] W. C. Jakes, « Microwave Mobile Communication », Piscataway, NJ : IEEE Press, 1983.
- [2] S. Guedaiem, “Détection et suivi de cibles par radar automobile à onde continue modulée en fréquence (FMCW),” Thèse de docteur, de l’université Paul Sabatier-Toulouse III, Spécialité Traitement du Signal, 1997.
- [3] Barton, D. K., and Leonov, S. A., Radar Technology Encyclopedia (Electronic edition), Artech House, INC. 1998.
- [4] Mahafza, B. R., “Radar systems analysis and design using Matlab,” PhD, Colsa Corporation; Huntsville, Alabama; Chapman & Hall/CRC 2000.
- [5] Ong, K. P., “Signal Processing for Airborne Bistatic Radar,” PhD Thesis, The university of Edinburgh, June 2003
- [6] Barkat, M., Signal Detection and Estimation, Artech House, 1991.
- [7] M. BELLANGER, ‘Traitement numérique du signal Théorie et pratique’, Science sup, 8<sup>ème</sup> édition, Dunod, 200
- [9] <https://fr.wikipedia.org/wiki/Interf%C3%A9rence>
- [10] Mathieu Pouliquen, « Introduction au filtrage adaptative et à l’égalisation », Cours de traitement numérique de signal.
- [11] S.Haykin – Adaptive Filter theory. 4 th ed. , Prentice Hall.
- [12] D. Hebb, THE ORGANIZATION OF BEHAVIOR, Wiley, New York, 1949
- [13] F. Rosenblatt, THE PERCEPTRON: A PERCEIVING AND RECOGNIZING AUTOMATON, Technical report, Cornell Aeronautical Lab, 1957.
- [14] M.L. Minsky and S.A. Papert, PERCEPTRONS, Cambridge, MIT Press, 1969.
- [15] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities”, *Proc. Nat. Acad. Sci.*, 2554–2558, 1982
- [16] J. Moody and C. J. Darken, “Fast learning in networks of locally-tuned processing units”, *Neural Computation*, 281–294, 1989.
- [17] Micheli-Tzanakou Evangelia, SUPERVISED AND UNSUPERVISED PATTERN RECOGNITION FEATURE EXTRACTION AND COMPUTATIONAL INTELLIGENCE, CRC Press LLC, 2000
- [18] S. Haykin. NEURAL NETWORKS: A COMPREHENSIVE FOUNDATION, 2nd Edition, Prentice Hall, 1999.

## Références bibliographiques

---

- [19] G. Dreyfus, J. M. Martinez, M. Samuelides, M. B. Gordon, F. Badran, S. Thiria, L. Héroult, RESEAUX DE NEURONES : METHODOLOGIE ET APPLICATIONS, Eyrolles, 2002.
- [20] Ossar, Y., (1998). Réseaux d'ondelettes et réseaux de neurones pour la modélisation statique et dynamique de processus. Thèse de Doctorat, Université de Paris 6, Paris.
- [21] Dreyfus, G., Martinez, J.M., Samuelides, M., Gordon, M.B., Badran, F., Thiria, S., Héroult(2002). Réseaux de neurones: méthodologie et applications, Eyrolles, Paris
- [22] M.H. Hassoun, FUNDAMENTALS OF ARTIFICIAL NEURAL NETWORKS, The MIT Press, 1995.
- [23] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition", *Neural Computation*, 541–551, 1989.
- [24] B. Prasad, S.R. M. Prasanna, SPEECH AUDIO IMAGE AND BIOMEDICAL SIGNAL PROCESSING USING NEURAL NETWORKS, Springer, 2008.
- [25] B. Dubuisson, DIAGNOSTIC ET RECONNAISSANCE DES FORMES, Hermes Sciences Publication, 1990.
- [26] L. Personnaz, I. Rivals, RESEAUX DE NEURONES FORMELS POUR LA MODELISATION LA COMMANDE ET LA CLASSIFICATION, CNRS Editions, 2003.
- [27] J. P. Haton, RECONNAISSANCE AUTOMATIQUE DE LA PAROLE: DU SIGNAL A SON INTERPRETATION, Dunod, 2006.
- [28] R. Begg, J. Kamruzzaman, R. Sarker, NEURAL NETWORKS IN HEALTHCARE: POTENTIALS AND CHALLENGES, IDEA Group Publishing, 2006
- [29] A. Wismuller, A. Meyer-Base, O. Lange, D. Auer, M. F. Reiser, D. Sumners, "Model-free functional MRI analysis based on unsupervised clustering", *Journal of Biomedical Informatics*, 10–18, 2004.
- [30] M. Buscema, M. Maszetti di Pietralata, V. Salvemini, M. Intragaligi, M. Indrimi, "Application of artificial neural networks to eating disorders", *Substance Use & Misuse*, 765–791, 1998.
- [31] S. Berdia, J. T. Metz, "An artificial neural network stimulating performance of normal subjects and schizophrenics on the Wisconsin card sorting test", *Artificial Intelligence in Medicine*, 123–138, 1998.

### Références bibliographiques

---

- [32] P. S. Heckerling, B. S. Gerber, T. G. Tapeç, R. S. Wigton, "Use of genetic algorithms for neural networks to predict community-acquired pneumonia", *Artificial Intelligence in Medicine*, 71–84, 2004.
- [33] D. B. Henson, S. E. Spenceley, D. R. Bull, "Artificial neural network analysis of noisy visual field data in glaucoma", *Artificial Intelligence in Medicine*, 10, 99– 113, 1997.
- [34] T.G. Barbounis, J.B. Theocharis, "Locally recurrent neural networks for long term wind speed and power prediction", *Neurocomputing*, 466–496, 2006.
- [35] S. Srivastava, M. Sing, M. Hanmandlu, A.N. Jha, "New fuzzy wavelet neural networks for system identification and control", *Applied Soft Computing*, 1–17, 2005.