

# Application des Réseaux Neuronaux pour la Modélisation et l'Optimisation de Réseaux d'Antennes Imprimées

L. MERAD, S.M. MERIAH et F.T. BENDIMERAD

Laboratoire de Télécommunications, Département d'Electronique  
Faculté des Sciences de l'Ingénieur, Université Abou-Bekr Belkaïd -Tlemcen  
BP 230, Pôle Chetouane, 13000 Tlemcen  
Email : l\_merad@mail.univ-tlemcen.dz

**Résumé** – Dans cet article notre contribution principale est le développement d'un modèle de synthèse de réseaux d'antennes imprimées. L'approche utilisée est basée sur les réseaux de neurones capable de modéliser et d'optimiser le système réseau d'antennes, compte tenu de critères généraux prédéterminés, et permet de réduire à l'extrême le temps de calcul lors de la phase d'utilisation.

**Mots clés** – antenne imprimée, réseau d'antennes, modélisation, optimisation, réseau de neurones

## I. INTRODUCTION

Dans le domaine des réseaux d'antennes imprimées, plusieurs méthodes de synthèse sont rencontrées à savoir les méthodes déterministes et stochastiques [1, 2]. Compte tenu de la diversité des buts recherchés par les utilisateurs, on ne trouvera pas une méthode générale de synthèse applicable à tous les cas, mais plutôt un nombre important de méthodes propres à chaque type de problème.

Cette diversité de solutions peut être exploitée pour constituer une base de données utile pour une approche générale de synthèse d'un réseau d'antennes imprimées.

Dans cet article, nous présentons les résultats concernant la modélisation et l'optimisation par les réseaux de neurones, par action sur le paramètre amplitude de l'alimentation.

Le réseau de neurones permet d'établir lors de la phase d'apprentissage des relations analytiques importantes pour l'étape de modélisation – optimisation du réseau d'antennes. Une grande flexibilité entre les caractéristiques du réseau d'antennes : amplitude d'alimentation, domaine d'ondulation, niveau de lobes secondaires, ... est ainsi introduite, puisqu'il n'y a aucune restriction quant au nombre de paramètres du système en entrée et en sortie.

## II. FORMULATION

Les réseaux de neurones artificiels (RNA) sont des modèles mathématiques inspirés de la structure et du comportement des neurones biologiques. Ils sont composés d'unités interconnectées que l'on appelle neurones formels ou artificiels capables de réaliser certaines fonctions particulières et bien précises [3].

Les RNA permettent d'approcher des relations non linéaires à des degrés de complexité importants. Les cellules d'entrées sont destinées à recueillir l'information qui est transformée par les cellules cachées jusqu'aux cellules de sortie. Ces réseaux possèdent une ou plusieurs couches cachées (figure.1). Généralement on utilise dans ce type de réseaux une fonction d'activation sigmoïde.

$$g(x) = \frac{1}{1 + \exp(-x)} \quad (1)$$

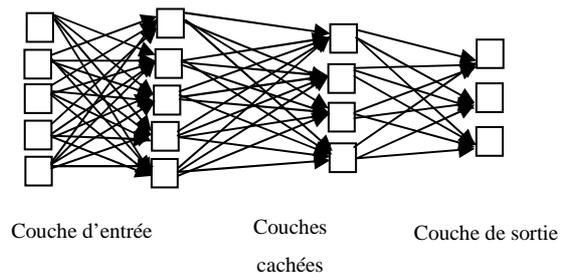


Fig .1. Réseaux multicouches

L'apprentissage dans ce type de réseau, consiste en un entraînement. On présente au réseau des entrées et on lui demande de modifier sa pondération de telle sorte que l'on retrouve la sortie correspondante.

L'algorithme consiste dans un premier temps à propager vers l'avant les entrées jusqu'à obtenir une sortie calculée par le réseau. La seconde étape compare la sortie calculée à la sortie réelle connue. On modifie alors les poids synaptiques de sorte qu'à la prochaine itération, l'erreur commise entre la sortie calculée et connue soit minimisée. On rétro-

propage alors l'erreur commise vers l'arrière jusqu'à la couche d'entrée tout en modifiant la pondération.

L'expression de calcul des nouvelles valeurs de poids synaptiques reliant les neurones est donnée par la relation suivante [4] :

$$W_{ij}(k+1) = W_{ij}(k) + \lambda D_i P_j \quad (2)$$

avec  $\lambda$  : pas d'apprentissage

$P_j$  : l'entrée du neurone  $j$ .

$W_{ij}$ : poids associé à la connexion du neurone  $i$  vers le neurone  $j$ .

$D_i$  : dérivée d'erreur du neurone  $i$ .

### III. RESULTATS

Dans notre application, nous avons utilisé les différentes fonctions de distribution d'amplitude d'alimentation (rectangulaire, triangulaire, Poisson, Tuckey, Cauchy, Gaussienne, Tchebycheff, ect ... ) [5], pour créer un fichier d'apprentissage nécessaire à l'élaboration du réseau de neurones.

La construction du réseau de neurones se fait par un processus itératif sur les échantillons d'une base de données contenu dans le fichier d'apprentissage. Chaque itération permet de minimiser l'erreur quadratique moyenne entre les sorties du RNA et les échantillons donnés.

#### A.Phase d'apprentissage

Après plusieurs essais, un réseau multicouches à été retenu avec la topologie suivante :

- Un neurone dans la couche d'entrée représentant le niveau maximum de lobes secondaires désiré  $NLS_{lim}$ .
- 20 neurones dans la couche cachée
- 10 neurones dans la couche de sortie représentant la loi d'amplitude pour un réseau symétrique à 20 éléments rayonnants.

Une fois que l'architecture du réseau a été décidée, la phase d'apprentissage permet de calculer les poids synaptiques menant à chaque neurone formel. Elle utilise l'algorithme de Quasi-Newton [6]. Cet algorithme consiste à présenter au réseau des exemples d'apprentissage, c'est-à-dire des jeux d'activités des neurones d'entrée ainsi que ceux des activités des neurones de sortie. On examine l'écart entre la sortie du réseau et la sortie souhaitée et on modifie les poids synaptiques des connexions jusqu'à ce que le réseau produise une sortie très proche de celle souhaitée. L'apprentissage par le logiciel Matlab est supervisé [6]. Les fonctions tangente hyperbolique, et linéaire sont affectées respectivement à la couche cachée et à la couche de sortie.

L'objectif essentiel ici, est de trouver le meilleur apprentissage qui permet de donner un bon modèle. Pour cela, plusieurs essais sont nécessaires, en

agissant sur les paramètres influant sur l'apprentissage. Ces paramètres sont les suivants :

- le nombre de neurones dans la couche cachée,
- les fonctions d'activation,
- le pas d'apprentissage.

La phase d'apprentissage est illustrée par la figure 2. Cette figure représente l'évolution de l'erreur quadratique moyenne entre les sorties du RNA et les échantillons donnés en fonction du nombre d'epochs, l'erreur finale obtenue est de 0.0129451.

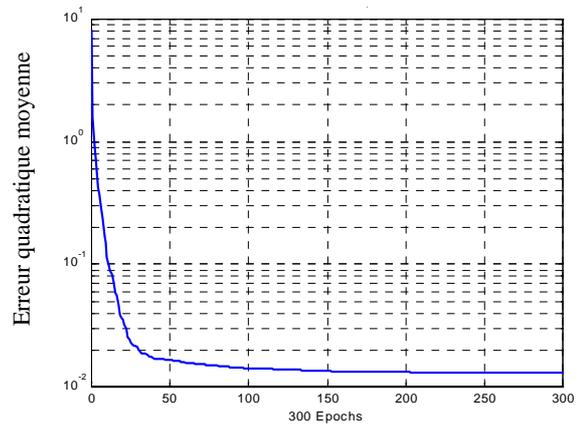


Fig .2. Phase d'apprentissage du réseau de neurones

#### B.Phase d'utilisation

Une fois l'apprentissage terminé, nous avons généré plusieurs niveaux maximaux de lobes secondaires afin de valider le modèle obtenu par le réseau de neurones. On a constaté que le réseau a pu reconnaître la plupart des exemples de tests avec un taux de reconnaissance de 90%, le taux de reconnaissance étant le rapport entre les exemples reconnus par le réseau et la totalité des exemples présentés au modèle construit par le RNA.

Les figures 3 et 4 présentent les résultats de synthèse de diagramme de rayonnement d'un réseau symétrique à 20 éléments pour les entrées de test respectives ( $NLS_{lim} = -33.1 \text{ dB}$  et  $NLS_{lim} = -17.2 \text{ dB}$ ).

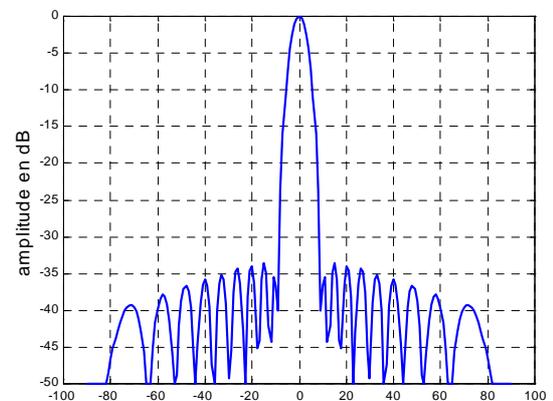


Fig.3. Diagramme d'un réseau symétrique à 20 éléments avec  $NLS_{lim} = -33.1 \text{ dB}$

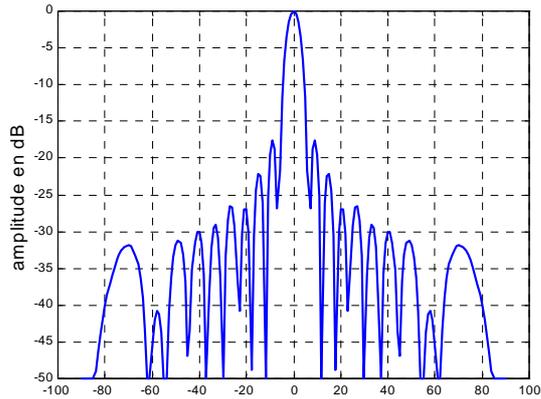


Fig.4. Diagramme d'un réseau symétrique à 20 éléments avec  $NLS_{lim} = -17.2 \text{ dB}$

#### IV. CONCLUSION

Au cours de cette étude, nous nous sommes intéressés aux réseaux d'antennes imprimées, particulièrement à leur modélisation et optimisation par les réseaux de neurones, par action sur le paramètre amplitude de l'alimentation. L'approche neuronale réduit à l'extrême le temps de calcul lors de la phase d'utilisation ou de généralisation. La précision du modèle construit dépend du nombre d'exemples concrets contenus dans la base de donnée de l'apprentissage. Cependant, les réseaux de neurones multicouches présentent l'inconvénient de la lenteur due à la phase d'apprentissage, et l'absence d'une règle générale pour définir l'architecture du réseau.

#### REFERENCES BIBLIOGRAPHIQUES

- [1] I.J. BAHL, P. BHARTIA, " Microstrip antennas ", *Artech House, USA, 1980.*
- [2] Y. CHUNYANG, G. DEYUAN, W. WENBING, " Nonuniform linear antenna array optimization – genetic algorithm approach ", *ISAE'97 proceeding, 1997.*
- [3] B. MULLER, J. REINHARDT, M. T STRCKLAND, " Neural networks an introduction ", *Springer-verlang, Berlin Heidelberg, 1995.*
- [4] Y. HAMMOU, M. ZERIKAT, A. BELAIDI, "Application des réseaux de neurones pour l'identification d'un moteur à courant continu en vue de son contrôle ", *COMAEI'96, Tlemcen, 1996.*
- [5] MIODRAG, MIKAVICA, ALEKSANDAR, NESIC, " CAD for linear and planar antenna arrays of various radiating elements ", *Artech House, Inc, 1992.*
- [6] H. DEMUTH, M. BEALE, « Neural network toolbox for use with Matlab », *Users guide, mathworks, 1997.*