

# Comparaison de méthodes de classification : application à l'identification d'objets ferromagnétiques sous-marins

## Comparison of Classification Methods : An Application in Underwater Ferromagnetic Object Identification

par André QUINQUIS\* et Emanuel RADOI\*\*

\* ENSIETA  
2, rue François Verny 29806 BREST Cédex 09  
Email : quinquis@ensieta.fr

\*\* Département Electronique et Informatique,  
Académie Technique Militaire, Bucarest, Roumanie

### *résumé et mots clés*

L'analyse temps-fréquence des signaux magnétiques, générés par des objets ferromagnétiques sous-marins, est utilisée afin de trouver un ensemble de paramètres discriminants pour leur classification. Après l'étape de sélection de caractéristiques, une étude étendue est menée pour comparer différentes structures de classifieurs, en fonction du taux moyen de bonne classification et de la capacité de généralisation. Les vertus de la logique floue, intégrée dans les classifieurs de la famille des K plus proches voisins (K-NN), et des classifieurs à base de réseaux de neurones sont mises en évidence au moyen d'un grand nombre de vecteurs de test et à une plage significative de variation pour le rapport S/B.

Signaux magnétiques sous-marins, analyse temps- fréquence, logique floue, réseaux de neurones, classification.

### *abstract and key words*

The time-frequency analysis of magnetic signals, generated by ferromagnetic objects, is used to extract a robust discriminant parameter set for their classification. After the feature selection phase, an extensive study is performed in order to validate the most appropriate classifier structure, in terms of the correct classification rate and the generalization ability. The K-NN family and the neural networks based classifiers are the candidates we are working with. The fuzzy-logic contribution and the neural networks capabilities are highlighted by means of a large number of test vectors and for a significative S/N ratio range.

Magnetic Underwater Signals, Time-Frequency Analysis, Fuzzy-Logic, Neural Networks, Classification.

## 1. introduction

La classification des objets statiques ou mobiles, dans l'espace libre ou dans le milieu sous-marin, à partir d'un signal ou d'une image contenant une information quant à leur structure ou à leur forme, constitue une préoccupation constante des spécialistes dans le domaine électronique. Les progrès tout à fait remarquables, enregistrés ces dernières années, notamment dans le développement des réseaux de neurones et des systèmes d'intelligence artificielle ont encore accentué cette tendance.

La plupart des systèmes de détection et de classification des objets qui évoluent dans le milieu sous-marin sont basés sur des sonars actifs ou passifs. Cependant, la signature acoustique de ces objets, notamment de ceux qui présentent une importance stratégique, est réduite constamment grâce à l'évolution de la technologie. Dans ces conditions, il est de plus en plus difficile pour les systèmes de détection et de classification acoustique d'accomplir leur mission. Ainsi, la mise au point d'une méthode alternative, basée sur un autre principe que les méthodes devenues déjà classiques, est aujourd'hui d'une importance prioritaire.

La plupart des objets sous-marins mobiles sont fabriqués de matériaux ferromagnétiques, ce qui fait qu'ils constituent « des

sources extrêmement efficaces de champ magnétique ambiant ». Ce champ peut être mesuré par des capteurs magnétiques (magnétomètres) immergés. L'intérêt et les potentialités de ces signaux sont soulignés dans le rapport LETI (CEA-Technologies Avancées) publié en 1994 : « Les informations *cachées* dans les signaux issus de magnétomètres, enregistrant les variations du champ magnétique terrestre et ses anomalies naturelles ou artificielles, sont pertinentes et complémentaires aux informations acoustiques obtenues à partir de sonars. Cette complémentarité offre des potentialités très intéressantes dans le cadre des applications de détection-localisation-identification en zones côtières ou stratégiques ».

Plus précisément, plusieurs études, comme par exemple celle de [1], montrent que tout objet ferromagnétique crée dans son voisinage une déformation des lignes de forces du champ magnétique terrestre qui, en l'absence de perturbations, est uniforme. Cette déformation dépend des composantes du moment magnétique de l'objet, qui, à leur tour, sont liées à la forme et aux dimensions de celui-ci. Il apparaît donc comme naturel d'essayer de reconnaître le type des objets ferromagnétiques à partir des mesures du champ induit, saisies par un capteur vectoriel ou scalaire immergé.

L'approche présentée dans cet article utilise l'analyse temps-fréquence des signaux magnétiques pour la *classification* des objets ferromagnétiques sous-marins. Elle a trois caractéristiques principales : efficacité, robustesse, rapidité. L'*efficacité* de la méthode développée est mesurée par l'excellent taux de bonne classification assuré et est donnée par la capacité discriminante des paramètres temps-fréquence extraits. Sa robustesse est prouvée par le comportement satisfaisant et régulier en présence de bruit. Enfin, sa *rapidité* est la conséquence directe de l'utilisation d'une transformation linéaire dans la phase de sélection des caractéristiques et des réseaux de neurones pour la phase de classification proprement dite.

Le schéma général de la chaîne de traitement est présenté au § 2. Le cadre physique du problème, les relations mathématiques de base qui modélisent les signaux magnétiques et une description très rapide des paramètres retenus font l'objet du § 3. L'analyse discriminante des paramètres extraits est réalisée au § 4 afin de mettre en évidence la contribution de chacun d'entre eux au pouvoir discriminant de l'ensemble. La même analyse est utilisée pour réduire la dimension de l'espace initial et préparer ainsi les données pour l'étape de classification. Le § 5 est dédié à l'analyse comparative des différentes structures possibles des classificateurs, y compris du point de vue de leur robustesse en présence de bruit. Les conclusions finales et les perspectives entrevues pour des développements futurs sont présentées au § 6.

## 2. la chaîne de traitement

Le schéma général de la chaîne de traitement est présenté sur la figure 1. Le fonctionnement du système est divisé en deux étapes : l'étape d'apprentissage et l'étape de test. Le processus d'apprentissage est réalisé à partir d'un certain nombre de signaux

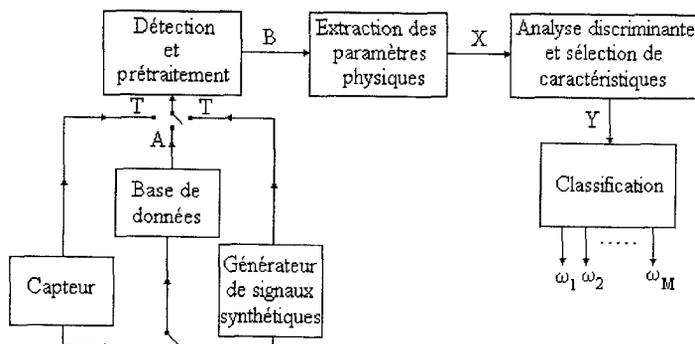


Figure 1. – Le schéma général de la chaîne de traitement.

(A) enregistrés dans une base de données et fournis par un capteur réel ou par un générateur de signaux synthétiques (nous avons utilisé cette dernière solution dans le cadre de cet article). Après avoir été entraîné, le système est testé en utilisant des signaux de test (T) générés de la même façon que les signaux d'apprentissage.

La détection des signaux magnétiques générés par des objets ferromagnétiques sous-marins fait depuis quelques années l'objet de nombreuses études et projets de recherche. La technique la plus récente utilise un réseau de magnétomètres pour résoudre ce problème [2]. Puisque les signaux détectés sont généralement fortement bruités, une opération de prétraitement (filtrage), comme par exemple celle décrite en [3], est mise en place afin d'améliorer le rapport signal/bruit. Dans le cadre de notre démarche, les problèmes de détection et de filtrage des signaux magnétiques ont été considérés résolus.

Dans la phase d'extraction des paramètres physiques, l'objectif est de mettre en évidence l'information cachée, liée à la structure particulière de chaque classe d'objets, au moyen d'une analyse complexe des signaux d'entrée dans le domaine temps-fréquence. Les paramètres extraits sont retenus en fonction de leur puissance discriminante et de leur robustesse à l'action du bruit. Le vecteur des paramètres retenus, X, représente les signaux magnétiques d'une manière spécifique, ce qui nous permet de le considérer comme une *signature magnétique* des objets sous-marins.

Bien que les données issues de la phase d'extraction des paramètres discriminants soient représentatives pour les différentes classes d'objets, elles ne sont pas encore adéquates pour être appliquées à l'entrée du classifieur à cause de leur dimension trop importante. Généralement, les algorithmes de classification sont d'autant plus efficaces que la dimension des vecteurs à classifier est faible. C'est la raison pour laquelle la dimension de l'espace initial est réduite de façon significative dans la phase de sélection des caractéristiques tout en gardant les propriétés discriminantes des paramètres extraits.

Enfin, dans la phase de classification, les vecteurs d'entrée, Y, sont assignés à une des M classes préétablies  $\omega_1, \omega_2, \dots, \omega_M$ . Cette dernière étape est donc l'étape décisionnelle. La justesse de la décision prise dépend des performances du classifieur choisi et de la qualité du processus d'apprentissage, mais aussi du niveau de bruit et des traitements précédents. Toutes les perturbations et les erreurs qui peuvent apparaître lors du déroulement de chaque étape se cumulent à ce stade du traitement, où la qualité du

processus est quantifiée à l'aide des différentes mesures, comme le taux moyen de bonne classification ou la matrice de confusion. Compte tenu de tous ces éléments nous avons consacré la partie la plus importante de notre article à la comparaison des performances de plusieurs méthodes de classification appliquées au problème qui nous concerne. Ainsi, nous avons comparé les types de classifieurs suivants : le plus proche prototype (NNP), les K plus proches voisins (K-NN), les K plus proches voisins dans la variante floue (Fuzzy K-NN), le perceptron multicouches (MLP), le réseau à fonction à base radiale (RBF) et le réseau de Kohonen dans la variante supervisée (LVQ). Ces classifieurs ont été comparés en fonction du taux moyen de bonne classification et de leur capacité de généralisation. Tous les classifieurs de type réseau de neurones ont été simulés en utilisant la toolbox de « Neural Networks », sous MATLAB.

### 3. extraction des paramètres physiques

#### 3.1. cadre du problème et signaux magnétiques

On supposera que l'objet à identifier suit une trajectoire rectiligne et parallèle à un axe de mesure, caractérisée par les paramètres de position  $(x, y, z)$  et la vitesse  $v$ . On supposera également que le capteur immergé est immobile ou du moins que son mouvement ne modifie pas l'information mesurée (fig. 2).

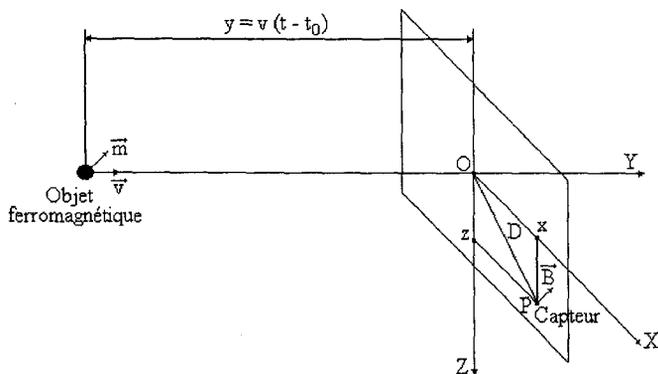


Figure 2. – Le schéma de mesure.

La distance  $D$  définit le plus proche point de la trajectoire (CPA - the Closest Point of Approach) par rapport au point de mesure P. Il peut être identifié par le système de mesure puisque l'amplitude du module du signal reçu devient maximale lorsque l'objet arrive en ce point.  $PO$  est alors orthogonal à l'axe  $OY$  et, par conséquent, la configuration du système de mesure est complètement déterminée par les coordonnées  $x$  et  $z$  du capteur par rapport au CPA (le point  $O$ ). Néanmoins, leurs valeurs sont inconnues puisque le capteur

est passif. La vitesse de l'objet est aussi supposée inconnue. En pratique, le rapport  $q = v/D$  est utilisé plutôt que la vitesse  $v$ .

En utilisant le développement du potentiel scalaire du champ magnétique en termes dipolaires, [1] démontre que les expressions des composantes de l'induction magnétique au point de mesure  $P$  sont données par l'équation :

$$[B] = 10^{-7} \cdot [W] \cdot [F] \tag{1}$$

où :

$$[B] = [B_x B_y B_z]^T \text{ le champ induit}$$

$$[F] = [f_0(u) f_1(u) f_2(u)]^T \text{ la projection sur le champ terrestre;}$$

$$[W] = \begin{bmatrix} -w_0 \cdot m_x & w_1 \cdot m_y & (w_3 + 2 \cdot w_0 - w_4) \cdot m_x \\ 2 \cdot w_0 \cdot m_y & w_1 \cdot m_x + w_2 \cdot m_z & -w_0 \cdot m_y \\ -w_0 \cdot m_z & w_2 \cdot m_y & w_3 \cdot m_x - (w_0 - w_4) \cdot m_z \end{bmatrix}$$

$m_x, m_y, m_z$  sont les composantes du moment magnétique;

$$\begin{aligned} w_0 &= (x^2 + z^2)^{-\frac{3}{2}}; & w_1 &= 3 \cdot x(x^2 + z^2); \\ w_2 &= 3 \cdot z \cdot (x^2 + z^2)^{-2}; & w_3 &= 3 \cdot x \cdot z(x^2 + z^2)^{-\frac{5}{2}}; \\ w_4 &= 3 \cdot z^2(x^2 + z^2)^{-\frac{5}{2}}; \end{aligned}$$

$f_0, f_1, f_2$  sont les fonctions d'Anderson [1];

$$u = \frac{y}{D} = \frac{v \cdot (t - t_0)}{D} = q \cdot (t - t_0)$$

Les domaines retenus pour les variations de  $x, y$  et  $q$  sont indiqués dans le tableau ci-dessous.

Tableau 1. – Les domaines de variations pour les paramètres de la trajectoire.

Paramètre	$x$	$z$	$q$
Domaine de variation	100 ... 600 m	100 ... 500 m	0,01 ... 0,25 s <sup>-1</sup>

Les valeurs des composantes du moment magnétique sont directement liées aux dimensions de l'objet selon les axes du système de coordonnées. Pour tout objet ferromagnétique mobile considéré, la dimension transversale selon l'axe  $OX$  est plus petite que les deux autres. Les différentes classes ont été alors définies à partir des rapports  $r_1 = m_y/m_x$  et  $r_2 = m_z/m_x$ . Par conséquent, le nombre de classes considérées sera dépendant de la discrétisation souhaitée du domaine de variation pour les deux rapports, qui s'étend habituellement entre 1 et 7. Nous avons travaillé avec un nombre total de 49 classes correspondant à toutes les combinaisons des valeurs entières comprises entre 1 et 7.

Dans chacune des classes, deux ensembles (d'apprentissage et de test), de 100 signaux chacun, ont été générés à partir de la relation (1), pour des valeurs de  $x, z$  et  $q$  tirées aléatoirement dans les intervalles de variation correspondants. Nous avons ainsi obtenu un nombre total de 4900 signaux dans l'ensemble d'apprentissage et un nombre équivalent pour l'ensemble de test.

### 3.2. description des paramètres extraits

Le premier objectif est de trouver un ensemble de paramètres discriminants et robustes au moyen des différentes représentations des signaux acquis. Ainsi, les signaux magnétiques générés ont été analysés dans les domaines temporel, fréquentiel et temporel-échelle, pour avoir une image conjointe concernant leurs propriétés. Dans chacun de ces domaines, nous avons testé tant la capacité discriminante que la robustesse au bruit des différents paramètres extraits et avons retenu seulement ceux qui apportaient une information significative quant à la répartition en classes et qui conservaient également un bon comportement pour des rapports signal/bruit faibles.

Une autre contrainte qui vient s'ajouter aux deux autres rappelées ci-dessus est que les paramètres discriminants doivent être indépendants de l'énergie globale des signaux puisque ceux-ci sont obtenus à la sortie d'un capteur passif.

#### A. Paramètres discriminants basés sur la variance des signaux

Trois paramètres sont calculés à partir des variances des composantes du champ magnétique induit  $B_i = (b_1^{(i)}, b_2^{(i)}, \dots, b_N^{(i)})$ ,  $i = 1 \dots 3$  comme étant :

$$v_i = \log \left( \frac{\hat{v}_i}{v_{\min}} \right), \quad i = 1 \dots 3 \quad (2)$$

où :  $\hat{v}_i = \frac{1}{N-1} \sum_{k=1}^N (b_k^{(i)} - \bar{B}_i)^2$  et  $v_{\min} = \min(\hat{v}_1, \hat{v}_2, \hat{v}_3)$ ,

l'indice  $k$  étant un indice temporel.

Ces paramètres décrivent la façon dont l'énergie est distribuée entre les composantes du signal magnétique. La robustesse au bruit apparaît alors comme une conséquence directe de leur nature énergétique.

#### B. Paramètres discriminants provenant du sous-espace signal de la matrice d'observation

Le comportement des paramètres extraits pour des rapports signal/bruit faibles est amélioré si l'information contenue par la matrice de corrélation ou la matrice d'observation est divisée en un sous-espace signal et un sous-espace bruit [4].

Ainsi, nous avons utilisé la décomposition en valeurs singulières de la matrice d'observation, définie pour un signal  $B = [b_1, b_2, \dots, b_N]$  comme étant :

$$T_p = \begin{bmatrix} b_p & \dots & b_1 \\ \vdots & \ddots & \vdots \\ b_N & \dots & b_{N-p} \end{bmatrix} \quad (3)$$

où  $p$  désigne la longueur de la fenêtre d'analyse. Si  $r$  représente la dimension du sous-espace signal, alors, en absence de bruit, la matrice  $T_p$  présente exactement  $r$  valeurs singulières non nulles. En présence de bruit, ses  $r$  plus grandes valeurs singulières correspondent au sous-espace signal, tandis que les autres correspondent au sous-espace bruit. Ainsi, pour diminuer l'influence du bruit, les paramètres discriminants retenus sont constitués par l'ensemble

des valeurs singulières  $\{\sigma_i\}_{i=1 \dots r}$ , correspondantes au sous-espace signal.

#### C. Paramètres discriminants obtenus par analyse en paquets d'ondelette

L'analyse en paquets d'ondelettes, développée par Coifman et Wickerhauser [5], généralise le concept d'analyse multirésolution et permet de travailler simultanément sur une collection de bases de décomposition. L'avantage essentiel apporté par cette technique est qu'on peut choisir ensuite parmi celles-ci la *meilleure base* de représentation de l'information au sens d'un certain critère. Le critère le plus souvent utilisé est celui de l'entropie minimale. Il s'agit d'une méthode très communément employée car elle ne suppose aucune restriction de l'ensemble des bases et trouve donc réellement la base de plus faible coût. Ce critère impose la minimisation de la quantité :

$$\gamma(B) = - \sum_i |b_i|^2 \cdot \log(|b_i|^2) \quad (4)$$

où  $B$  représente le signal  $B = \{b_i\}$  qui équivaut à la minimisation de l'entropie définie dans le sens de Shannon-Weaver et qui constitue de plus une mesure additive de l'information [5].

Une configuration d'entropie minimale est caractéristique d'un certain ordre, d'une certaine régularité. Ainsi, l'ensemble des valeurs de l'entropie, à partir du deuxième niveau de décomposition, a été pris en compte pour discriminer les signaux analysés.

Les coefficients de décomposition dans la meilleure base ainsi que la valeur de l'entropie seront retenus comme caractéristiques du signal analysé.

#### D. Matricage des paramètres discriminants

Bien que tous les paramètres décrits dans ce paragraphe aient un bon comportement tant du point de vue de la capacité discriminante que du point de vue de la robustesse au bruit, leur utilisation séparée ne permet pas d'obtenir les résultats souhaités en classification. Leur réunion a alors été considérée (fig. 3) afin de constituer le vecteur complet des paramètres physiques capable d'individualiser les classes d'objets définies au § 3.1.

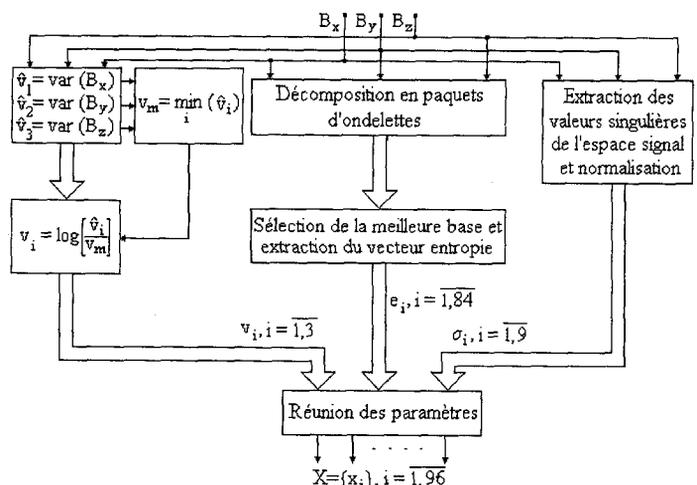


Figure 3. – Le schéma général pour l'extraction des paramètres discriminants.

## 4. analyse discriminante des paramètres extraits et sélection des caractéristiques

La sélection des caractéristiques est une étape clé dans le processus de la reconnaissance des formes, qui a une influence directe sur la complexité et les performances du classifieur. Elle consiste en une transformation linéaire ou non-linéaire de l'espace initial des paramètres dans un espace à dimension réduite, tout en gardant l'information essentielle sur les entités contenues dans l'espace initial. Cette transformation est à la fois *possible*, car la représentation initiale est d'habitude très redondante, et *nécessaire*, car une dimension trop importante des vecteurs d'entrée peut rendre inefficace ou déterminer une structure surdimensionnée du classifieur.

Les transformations linéaires sont les plus utilisées de par leur simplicité et leur efficacité en temps de calcul. Elles sont définies par la relation :

$$Y = K \cdot X \quad (5)$$

où  $K$  représente la matrice de la transformation de dimension  $(m, n)$ .  $X = (x_1 \ x_2 \ \dots \ x_n)$  et  $Y = (y_1 \ y_2 \ \dots \ y_m)^T$  désignent respectivement les vecteurs des paramètres dans l'espace initial et transformé.

Pour la classification supervisée, les transformations linéaires les plus connues sont celles qui reposent sur les critères de séparabilité et qui sont généralement référées sous le nom d'analyse *factorielle discriminante* (AFD) [6]. Ces transformations cherchent à maximiser la distance entre les différentes classes et, à la fois, à minimiser la distance entre les éléments appartenant à la même classe.

Cette démarche apparaît très naturelle, étant donné que la tâche du module qui suit, le classifieur, est justement de séparer au moyen de surfaces de décision les régions correspondantes aux différentes classes dans l'espace des vecteurs d'entrée. On comprend bien que la complexité de ces surfaces, et donc du classifieur qui doit les engendrer, est d'autant plus élevée que les classes sont mélangées, d'où l'importance tout à fait particulière de cette étape dans l'ensemble du traitement.

La séparabilité des classes est évaluée au moyen d'une fonction « critère », qui est définie à l'aide des matrices de dispersion (inertie) intra-classes et inter-classes. Pour une certaine classe  $\omega_i$ , la première matrice mesure la dispersion de ses éléments autour de son vecteur moyenne  $\mu_i$  et s'exprime naturellement par sa matrice de covariance  $\Sigma_i$ . Pour l'ensemble des  $M$  classes, supposées équiprobables, la matrice de dispersion intra-classes est obtenue en moyennant les matrices  $\Sigma_i$  :

$$S_W = \sum_{i=1}^M P(\omega_i) \cdot \Sigma_i = \frac{1}{M} \cdot \sum_{i=1}^M \Sigma_i \quad (6)$$

où  $P(\omega_i)$  est une loi équirépartie sur les  $\omega_i$ .

Les inerties interclasses mesurent les dispersions moyennes entre les  $M$  classes et sont définies sous la forme :

$$S_B = \sum_{i=1}^M P(\omega_i) \cdot (\mu_i - \mu_0) \cdot (\mu_i - \mu_0)^T = \frac{1}{M} \cdot \sum_{i=1}^M (\mu_i - \mu_0) \cdot (\mu_i - \mu_0)^T \quad (7)$$

$$\text{où : } \mu_0 = \sum_{i=1}^M P(\omega_i) \cdot \mu_i = \frac{1}{M} \cdot \sum_{i=1}^M \mu_i$$

Il est alors démontré [6] que si  $S_T = S_B + S_W$ , la matrice optimale  $K$  est formée des vecteurs propres de la matrice :

$$\Sigma = S_T^{-1} \cdot S_B \quad (8)$$

correspondant aux plus grandes valeurs propres.

Une propriété très intéressante de cette matrice [8] est que si  $M \leq n + 1$  (où  $M$  représente le nombre des classes et  $n$  la dimension de l'espace initial), le nombre de valeurs propres non nulles vaut toujours  $M - 1$ . Ceci signifie que si plusieurs paramètres sont utilisés pour discriminer les classes, leur puissance séparatrice est toujours obligée de se concentrer dans la même zone de l'espace généré par les vecteurs propres associés aux valeurs propres.

Ainsi, l'augmentation des valeurs propres significatives, obtenues par leur réunion, dépendra de la mesure dans laquelle leurs actions sont complémentaires. Si les différents paramètres apportent la même information quant aux caractéristiques des objets à classer, aucune amélioration ne sera obtenue par leur réunion.

Les valeurs propres, qui mesurent la capacité discriminante de chaque composante, sont toujours positives. [6] démontre aussi qu'elles sont également majorées par 1. Donc :

$$0 \leq \lambda_i \leq 1, \quad \forall i = \overline{1, n} \quad (9)$$

Ce résultat est particulièrement important puisqu'il permet d'utiliser les valeurs propres comme une mesure de la puissance discriminante des différents paramètres extraits.

Dans un premier temps nous avons appliqué successivement l'analyse factorielle discriminante, définie ci-dessus, aux trois sous-ensembles de paramètres physiques, extraits au § 3.2. Ensuite, la même procédure a été appliquée pour toutes les combinaisons de deux sous-ensembles et enfin les trois sous-ensembles à la fois.

Les résultats obtenus pour les signaux synthétiques idéaux (sans bruit) de la base d'apprentissage, sont présentés dans le tableau 2. Seules les 8 premières valeurs propres sont données pour les 5 derniers cas.

Il est à noter le bon comportement discriminant de chaque paramètre exprimé par l'importance des valeurs propres associées. Le renforcement des valeurs propres lors de leur réunion démontre clairement la complémentarité de leur action discriminante.

Pour avoir une idée plus précise sur la manière dont les classes sont séparées, nous avons représenté la projection des vecteurs de l'ensemble d'apprentissage dans l'espace transformé sur le plan engendré par les vecteurs propres correspondants aux deux plus grandes valeurs propres.

Les résultats sont satisfaisants dès que la réunion de tous les paramètres est considérée (annexe 2a). En effet, les classes sont parfaitement séparées, même si on considère uniquement la projection sur ce plan.

Tableau 2. – Les valeurs propres issues de l’analyse discriminante des paramètres sans bruit.

Paramètres physiques	Valeurs propres							
Variance (V)	<b>0.992</b>	0.722	0.283					
Valeurs singulières du sous-espace signal (VS)	<b>0.997</b>	<b>0.982</b>	<b>0.979</b>	0.016	0.01	0.007	0.005	0.001
Décomposition en paquets d’ondelettes (PO)	<b>0.994</b>	<b>0.950</b>	<b>0.897</b>	0.760	0.645	0.517	0.415	0.314
V + VS	<b>0.998</b>	<b>0.990</b>	<b>0.984</b>	<b>0.952</b>	0.450	0.199	0.015	0.001
V + PO	<b>0.999</b>	<b>0.999</b>	<b>0.897</b>	<b>0.970</b>	0.670	0.652	0.418	0.343
VS + PO	<b>0.999</b>	<b>0.997</b>	<b>0.992</b>	<b>0.944</b>	<b>0.811</b>	0.709	0.568	0.438
V + VS + PO	<b>0.999</b>	<b>0.999</b>	<b>0.998</b>	<b>0.991</b>	<b>0.984</b>	<b>0.913</b>	0.721	0.442

Tableau 3. – Les valeurs propres issues de l’analyse discriminante des paramètres pour  $S/N = 30$  dB.

Paramètres physiques	Valeurs propres							
Variance (V)	<b>0.992</b>	0.725	0.35					
Valeurs singulières du sous-espace signal (VS)	<b>0.996</b>	<b>0.978</b>	<b>0.976</b>	0.013	0.009	0.006	0.002	0.001
Décomposition en paquets d’ondelettes (PO)	<b>0.986</b>	<b>0.920</b>	<b>0.810</b>	0.670	0.567	0.469	0.353	0.252
V + VS	<b>0.997</b>	<b>0.987</b>	<b>0.981</b>	<b>0.939</b>	0.447	0.257	0.012	0.001
V + PO	<b>0.999</b>	<b>0.997</b>	<b>0.957</b>	<b>0.826</b>	0.591	0.569	0.354	0.298
VS + PO	<b>0.999</b>	<b>0.994</b>	<b>0.988</b>	<b>0.906</b>	<b>0.697</b>	0.593	0.450	0.358
V + VS + PO	<b>0.999</b>	<b>0.998</b>	<b>0.996</b>	<b>0.985</b>	<b>0.972</b>	<b>0.875</b>	0.598	0.364

Néanmoins, afin de renforcer la robustesse du système, nous avons décidé finalement de ne pas l’entraîner avec des signaux idéaux, mais avec des signaux faiblement bruités (rapport  $S/B = 30$  dB). Les valeurs propres obtenues dans ce cas sont données dans le tableau 3, et la même projection qu’avant a été effectuée (annexe 2b).

On peut remarquer une certaine dégradation dans l’aspect général du groupement des classes. Malgré cette impression visuelle, les classes sont encore parfaitement distinctes, car les régions qui apparaissent comme confondues dans ce plan sont séparées dans les autres dimensions. Ainsi, nous avons obtenu une classification parfaite pour  $S/N = 30$  dB, tout en gardant le même résultat pour les signaux idéaux.

Pour réaliser la sélection des caractéristiques, seuls les 6 premiers vecteurs propres correspondants aux 6 valeurs propres les plus importantes sont conservés (cf. tableau 3). Nous avons ainsi réalisé une réduction 19 : 1 de la dimension de l’espace initial des paramètres. La matrice K construite à l’aide de ces vecteurs reste dorénavant figée et elle servira à la sélection des caractéristiques de tout autre ensemble de vecteurs de test.

## 5. la classification

Le problème a trait à la classification supervisée, c’est-à-dire pour lequel « l’utilisateur connaît et le système utilise l’information concernant l’appartenance aux classes de vecteurs d’apprentissage » [10].

Mettre en œuvre un tel classifieur revient à trouver, pendant la phase d’apprentissage, et à utiliser, pendant la phase de test, les fonctions discriminantes  $g_1, \dots, g_M$ , telles que :

$$Y \in \omega_i \implies g_i(Y) = \max_{j=1, \dots, M} |g_j(Y)| \quad (10)$$

Le classifieur est *optimal*, au sens de la minimisation de l’erreur *a posteriori*, si :

$$g_i(Y) = f(Y|\omega_i) \cdot P(\omega_i) \quad (11)$$

Dans l’expression ci-dessus,  $P(\omega_i)$  désigne la probabilité *a priori* de la classe  $\omega_i$ , tandis que  $f(Y|\omega_i)$  désigne la densité de probabilité des vecteurs à classer conditionnellement à  $\omega_i$ .

Ce classifieur s'appelle *Bayésien* et il est intéressant seulement du point de vue théorique, comme référence, car généralement les expressions exactes de  $P(\omega_i)$  et de  $f(Y|\omega_i)$  ne sont pas connues.

Dans la plupart des cas, les classes sont considérées comme équiprobables, bien que, lorsqu'on dispose d'une certaine connaissance *a priori* sur leur structure, une modélisation beaucoup plus efficace de  $P(\omega_i)$  soit possible (par exemple la modélisation markovienne en traitement d'image). Trois autres approches sont alors souvent utilisées pour fournir  $f(Y|\omega_i)$ , ou de manière équivalente,  $g_i(Y)$  : l'approche paramétrique linéaire ou non linéaire et l'approche non paramétrique. Les classifieurs de type K-NN sont rappelés dans l'annexe 4, tandis que l'annexe 5 présente quelques types de réseaux de neurones utilisés dans notre étude.

Les performances des classifieurs décrits dans les annexes 4 et 5 ont été déterminées sur un ensemble représentatif de vecteurs pour chaque classe, généré de la manière présentée en § 3.1, pour un rapport S/B variant entre 5 et 30 dB, avec un pas de 1 dB. A partir de 30 dB, les classes sont parfaitement séparées, grâce aux paramètres extraits et aux traitements effectués; le taux de bonne classification est alors de 100 %. Le bruit est supposé blanc, gaussien et de moyenne nulle.

Une image suggestive de l'effet du bruit sur la séparation des classes est présentée dans l'annexe 3. L'exemple de 4 classes voisines, définies par les rapports  $r_1$  et  $r_2$  indiqués par la légende, a été considéré pour 6 valeurs du rapport S/B. On peut facilement remarquer le comportement régulier des nuages de points, dont le volume augmente proportionnellement au niveau du bruit. Par conséquent, même si un vecteur est mal classifié, il est classifié de toute façon dans une des classes voisines, et donc l'erreur n'est pas très conséquente, vu la manière dont les classes ont été définies et la signification physique des rapports  $r_1$  et  $r_2$ .

Nous avons comparé dans un premier temps les performances des méthodes de la famille K-NN. Une étude a été menée (table 4) pour trouver les valeurs optimales du nombre de voisins les plus proches tant pour la phase de « fuzzyfication » ( $K_F$ ), que pour la phase de traitement (K). Nous avons trouvé ainsi que la méthode K-NN fournit les meilleurs résultats pour  $K = 3$ . Ensuite, entre les 2 méthodes de « fuzzyfication » décrites dans l'annexe 4, celle donnée par la relation (A4.4), pour  $K_F = 5$ , s'est avérée la plus performante (seule cette méthode va être analysée par la suite sous le nom de « Fuzzy K-NN »).

Enfin, nous avons calculé, pour ces méthodes et pour le réseau LVQ, le taux moyen de bonne classification pour chaque valeur du rapport S/B. Les résultats sont présentés sur la figure 4.

On peut remarquer le bon comportement du réseau LVQ et du classifieur utilisant la variante floue de la méthode K-NN, ce qui met en évidence l'apport important de ce type de logique pour la modélisation des processus décisionnels. Néanmoins, ce dernier est pénalisé par le nombre important de calculs qui doivent être effectués lors de l'étape de classification. Pratiquement, seule la « fuzzyfication » des vecteurs d'entraînement est réalisée dans la phase d'apprentissage, le calcul des plus proches voisins et la « défuzzyfication » n'étant effectués qu'à la réception du signal. C'est la raison pour laquelle une classification rapide, en temps réel, n'est pas possible en utilisant cette méthode. Par opposition, un réseau de neurones est entraîné en amont de la décision, lorsque la durée du temps de calcul n'est pas critique, et par conséquent,

Tableau 4. – Le choix du nombre de voisins pour les méthodes K-NN.

Méthode	K	Taux moyen de bonne classification	
		S/B = 15 dB	S/B = 20 dB
K-NN	1	0.6247	0.8815
	<b>3</b>	<b>0.6486</b>	<b>0.8829</b>
	5	0.6351	0.8782
Fuzzy K-NN 1	1	0.6423	0.8754
	3	0.6604	0.8802
	<b>5</b>	<b>0.6812</b>	<b>0.8957</b>
	7	0.6763	0.8933
Fuzzy K-NN 2 ( $K_F = 5$ )	1	0.6725	0.8964
	<b>3</b>	<b>0.6982</b>	<b>0.9000</b>
	5	0.6901	0.8992

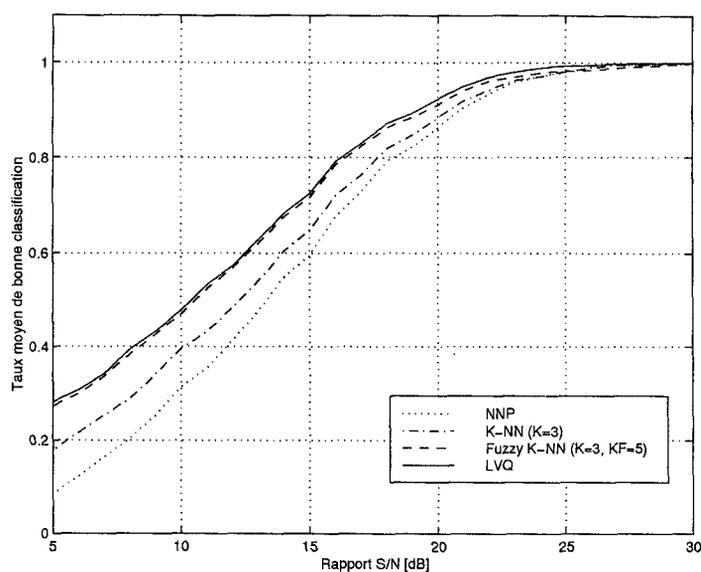


Figure 4. – Comparaison entre les performances des méthodes NNP, K-NN, Fuzzy K-NN et LVQ.

le temps de classification correspondra au temps de propagation par ses couches, ce qui équivaut à une classification en temps réel.

Nous avons comparé ensuite les performances des 3 réseaux de neurones décrits en annexe 5. Nous avons, dans les trois cas, utilisé des réseaux avec une seule couche cachée. Le nombre de neurones sur la couche d'entrée est fixé par la dimension de l'espace des caractéristiques, à savoir 6, dans tous les cas, tandis que le nombre de neurones sur la couche de sortie est égal au nombre de classes, à savoir 49, pour les réseaux RBF et LVQ, et 2 pour le réseau MLP, à cause de sa configuration particulière. Le nombre de neurones sur la couche cachée a été choisi égal à 98 (deux fois le nombre de

## Comparaison de méthodes de classification

classes), pour RBF et MLP, et 250 pour LVQ. Pour ce dernier les résultats ont déjà été présentés sur la figure 4. Son entraînement a été réalisé à partir d'un ensemble de vecteurs générés pour un rapport  $S/B = 30$  dB.

Afin d'assurer la capacité de généralisation maximale du réseau et d'éviter son surentraînement, nous avons également utilisé, pendant la phase d'apprentissage, 3 ensembles de test obtenus respectivement pour des rapports  $S/B = \infty$ ,  $S/B = 30$  dB et  $S/B = 20$  dB. Bien qu'il soit ensemble de test, l'ensemble des vecteurs non-bruités a été le premier à atteindre un taux moyen de bonne classification de 100 %. La même valeur a été également obtenue, après quelques époques, pour les ensembles d'apprentissage et de test correspondant à  $S/B = 30$  dB. Néanmoins, nous avons poursuivi l'entraînement jusqu'au moment où le taux moyen de bonne classification pour le dernier ensemble de test a cessé d'augmenter.

La variation du taux moyen de l'erreur de classification, pendant le processus d'apprentissage, pour les 4 ensembles de vecteurs est représentée sur la figure 5.

Pour le MLP nous avons adopté une configuration spéciale. Ainsi, nous avons seulement utilisé deux neurones, sur la couche de sortie, que nous avons munis d'une fonction d'activation linéaire. Les sorties souhaitées, indiquées au réseau, sont les valeurs des deux rapports qui définissent chaque classe (cf. § 3.1). Puisque, dans ce cas, le problème est plutôt un problème d'estimation que de classification, nous avons utilisé comme mesure de la qualité de la décision le rapport :

$$T_{BC} = 1 - \frac{\|D - O\|}{\|D - \bar{D}\|} \quad (12)$$

où  $O$  désigne le vecteur de sortie réel pour le vecteur d'entrée courant,  $D$  désigne le vecteur de sortie souhaité pour ce vecteur et  $\bar{D}$  désigne le vecteur de sortie souhaité le plus éloigné de  $D$ .

Il est facile de comprendre que si le vecteur est correctement classifié, alors  $T_{BC}$  est égal à 1, sinon la valeur de  $T_{BC}$  n'est pas nulle,

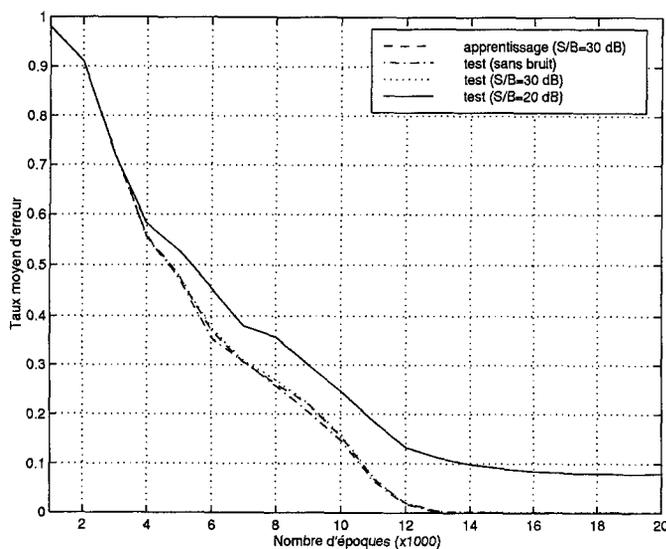


Figure 5. – Le taux moyen de l'erreur de classification pendant l'entraînement du réseau LVQ.

mais décroît en fonction de l'erreur d'estimation commise.  $T_{BC}$  est égal à 0, lorsque l'erreur d'estimation maximale est obtenue. Nous avons assimilé cette mesure de performance au taux moyen de bonne classification défini pour les autres classificateurs, bien qu'elle soit un indicateur de qualité tout à fait optimiste. Néanmoins, il paraît assez logique du point de vue physique d'associer des poids différents aux erreurs d'estimation différentes, puisque globalement on peut ainsi mieux apprécier le comportement du système.

Malgré ses qualités tout à fait remarquables, le réseau RBF n'est pas capable de travailler avec une base d'apprentissage importante. Cette limite provient du calcul global effectué par l'algorithme OLS dans la première étape. Pour pouvoir néanmoins utiliser ce réseau nous avons constitué une base d'apprentissage de taille réduite, en prenant seulement 6 vecteurs par classe, parmi les 100 initialement disponibles. Les 6 vecteurs sélectionnés pour chaque classe contiennent les valeurs maximales des coordonnées de l'espace transformé (une par vecteur), de telle manière qu'ils constituent une sorte d'enveloppe pour la classe considérée. Bien que la nouvelle base d'apprentissage soit beaucoup plus réduite, les résultats obtenus sont comparables à ceux que nous avons obtenus en utilisant les deux autres types de réseaux (fig. 6). Il s'agit donc en même temps d'une évaluation pessimiste des performances de la classification.

Entre les deux variantes, pessimiste (RBF) et optimiste (MLP), le réseau LVQ nous semble un bon compromis, vu les bons résultats obtenus et le fait que cet algorithme peut manipuler facilement un volume important de vecteurs dans la phase d'apprentissage. Des résultats plus détaillés concernant la classification avec ce type de réseau sont présentés sur la figure 7. Trois types de représentations ont été réalisés pour 4 valeurs du rapport S/B indiqués sur cette figure, à savoir : la répartition du taux de bonne classification (TBC) sur le nombre de classes, le TBC pour chaque classe et la matrice de confusion.

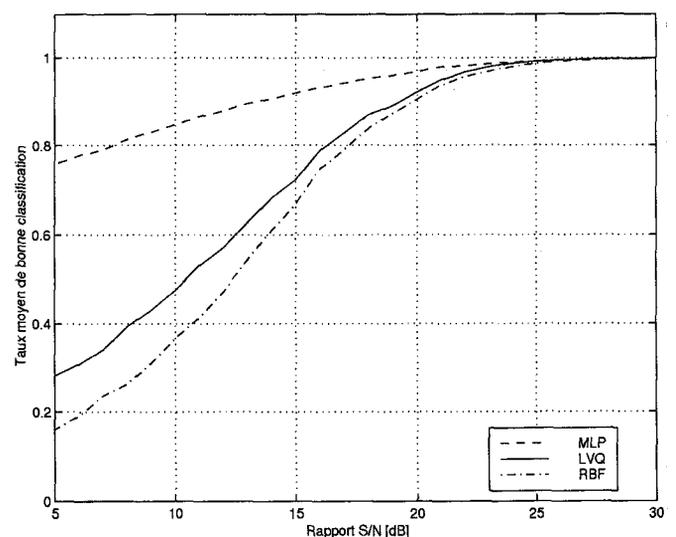


Figure 6. – Comparaison entre les performances des 3 types de réseaux de neurones.

## Comparaison de méthodes de classification

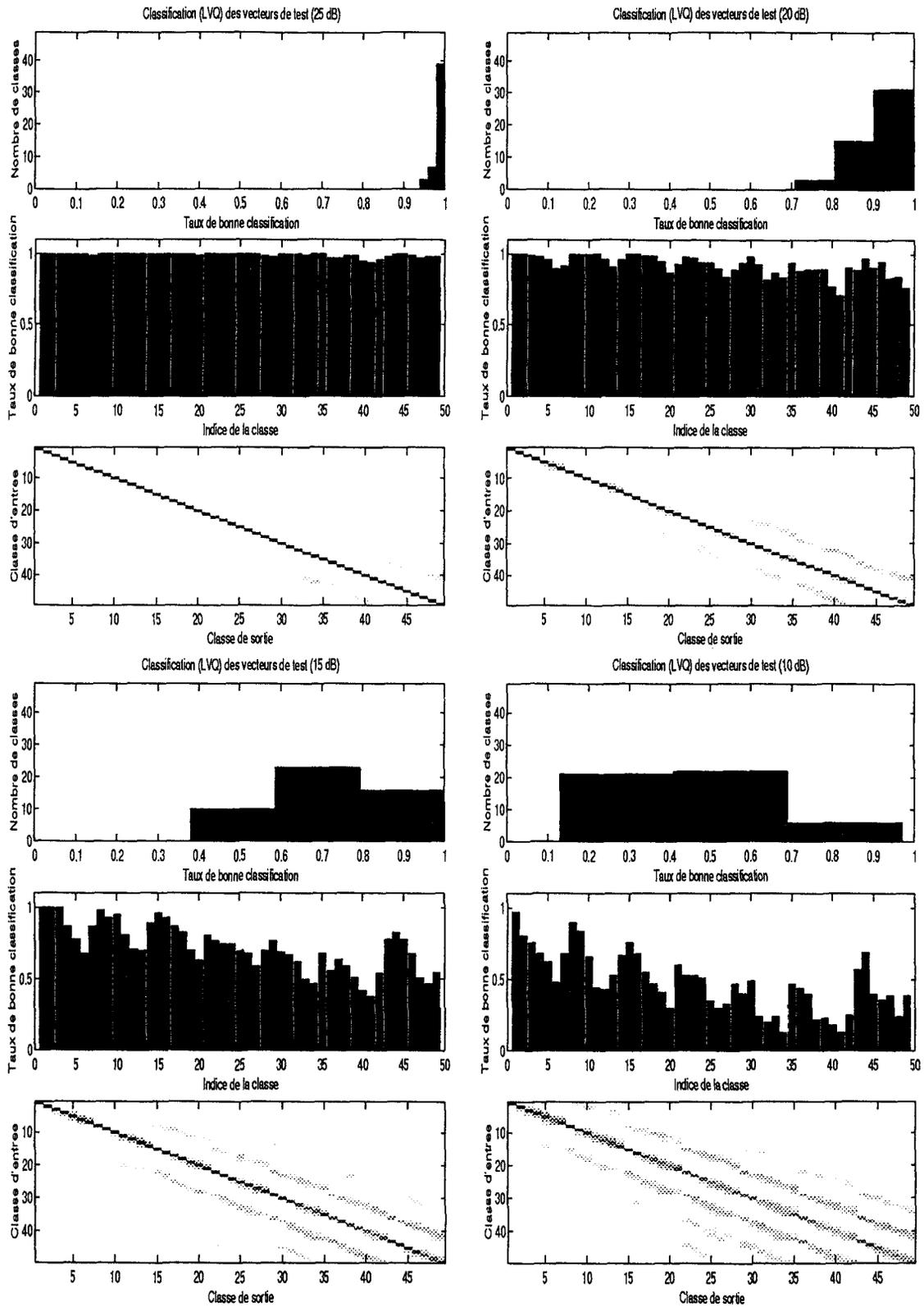


Figure 7. – Résultats de la classification utilisant le réseau LVQ.

La première représentation nous donne rapidement une idée sur la qualité de la classification. On peut, par exemple, remarquer que presque toutes les classes ont un TBC plus grand que 0.95 dans le premier cas, tandis que dans le dernier cas un TBC égal à 1 n'est réalisé par aucune classe. Le TBC pour chaque classe est donné par la deuxième représentation, dont les minima correspondent aux classes les plus proches (voir annexe 2). Enfin, la troisième représentation nous permet d'apprécier à la fois la qualité de la classification, car celle-ci est d'autant meilleure que la matrice de confusion est diagonale, et la mesure dans laquelle le jeu de données utilisé dans la phase d'apprentissage est représentatif pour la distribution des classes. Ainsi, comme on peut le remarquer tout de suite, les vecteurs de test mal classifiés ne sont pas attribués à des classes quelconques, mais aux classes qui se trouvent dans le voisinage de leur classe réelle. On aperçoit également, sur toutes ces figures, la dégradation progressive et contrôlée, proportionnelle au niveau du bruit, des performances du classifieur, qui montre ainsi sa robustesse.

## 6. conclusions

L'originalité de l'étude porte sur : l'utilisation des signaux magnétiques pour classier les objets ferromagnétiques sous-marins, l'identification d'un ensemble de paramètres physiques fortement discriminant et robuste en présence de bruit, l'analyse comparative des différentes structures de classifieurs et leur utilisation (la variante de « fuzzyfication » la plus adaptée et les valeurs optimales des paramètres associés, la mesure de la qualité de la classification pour MLP, la manière d'entraînement des classifieurs LVQ et RBF).

La méthode décrite permet de classier les objets ferromagnétiques sous-marins, en fonction des composantes de leur moment magnétique. Trois sous-ensembles de paramètres physiques ont été extraits comme étant à la fois suffisamment discriminants et robustes à l'action du bruit.

L'ensemble des paramètres extraits a été soumis à une analyse de données sous la forme de l'analyse factorielle discriminante. Celle-ci nous a permis de quantifier l'apport de chaque paramètre à la capacité discriminante totale et finalement de réduire sensiblement la dimension de l'espace qui les contient.

Plusieurs structures de classifieurs ont été comparés, en fonction du taux moyen de bonne classification et de la capacité de généralisation. Les classifieurs de type K-NN, provenant directement des méthodes d'estimation non paramétriques de la densité de probabilité, ont un comportement assez bon, surtout lorsque la logique floue est intégrée à leur principe de fonctionnement. Leur grand désavantage réside dans le temps de calcul important requis dans la phase décisionnelle, ce qui rend ces classifieurs inadéquats à la classification en temps réel.

Les classifieurs de type réseau de neurones permettent l'obtention de performances généralement supérieures aux méthodes K-NN, grâce à leur capacité de généralisation, et réalisent en même temps la classification en temps réel, puisque le processus d'apprentissage s'est déroulé bien avant la phase décisionnelle. Si les

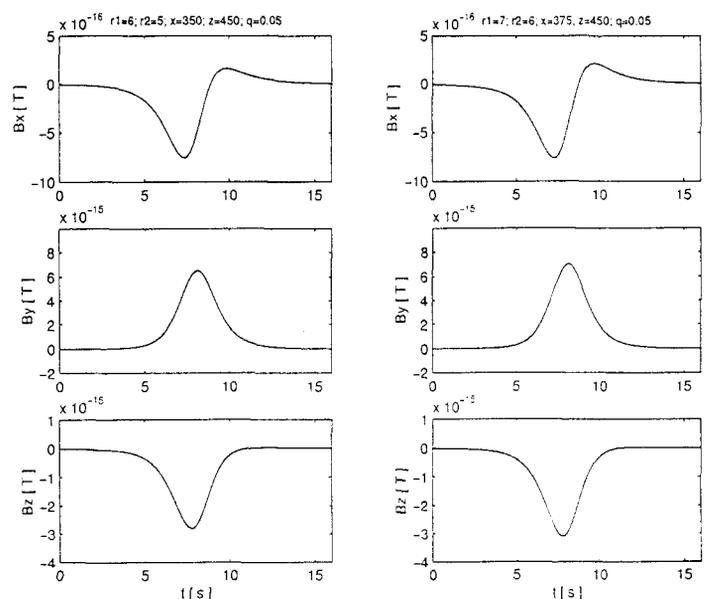
réseaux sont correctement entraînés, alors leurs performances sont à peu près équivalentes. Le choix du type de réseau dépend alors des contraintes du problème concret. C'est la raison pour laquelle nous avons essayé d'identifier dans une première étape la configuration de chaque réseau le mieux adapté aux contraintes de notre problème et ensuite de choisir parmi ces réseaux celui qui nous semblait avoir réalisé le meilleur compromis. Par conséquent, notre comparaison n'a pas concerné tant leurs performances absolues (auquel cas il aurait fallu considérer des structures identiques) que leur capacité de résoudre un problème concret et d'être intégrés dans le système conçu.

Le système présenté assure une classification parfaite pour un rapport  $S/B \geq 30$  dB et une décroissance acceptable du taux moyen de bonne classification avec le niveau du bruit, pour  $S/B < 30$  dB. Si on considère la valeur 0,7 comme limite minimale acceptable du taux moyen de bonne classification, alors la robustesse du système est assurée jusqu'à 15 dB. Si cette valeur est baissée à 0,5 alors la robustesse du système peut atteindre 10 dB.

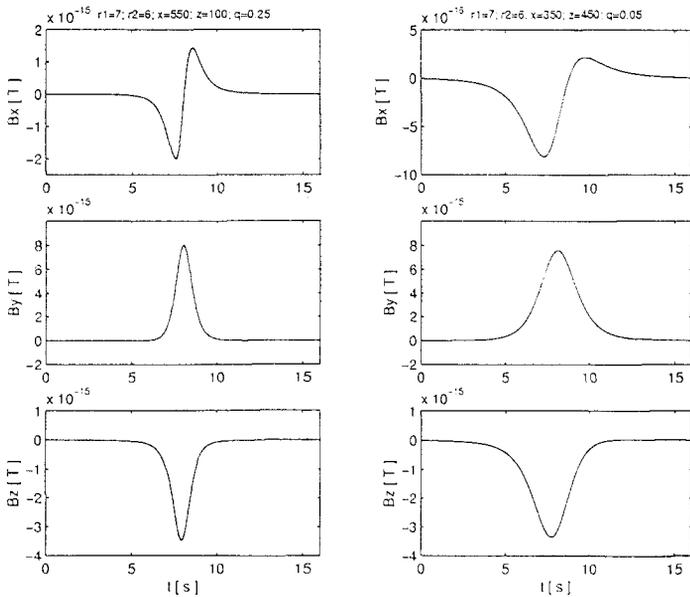
Nous envisageons par la suite de tester le comportement du système en présence d'un bruit magnétique réel, non stationnaire, et d'étudier de plus près les possibilités des méthodes non linéaires pour la sélection des caractéristiques, ainsi que l'influence des différentes distances non-euclidiennes dans le cadre de ce problème.

## 7. annexes

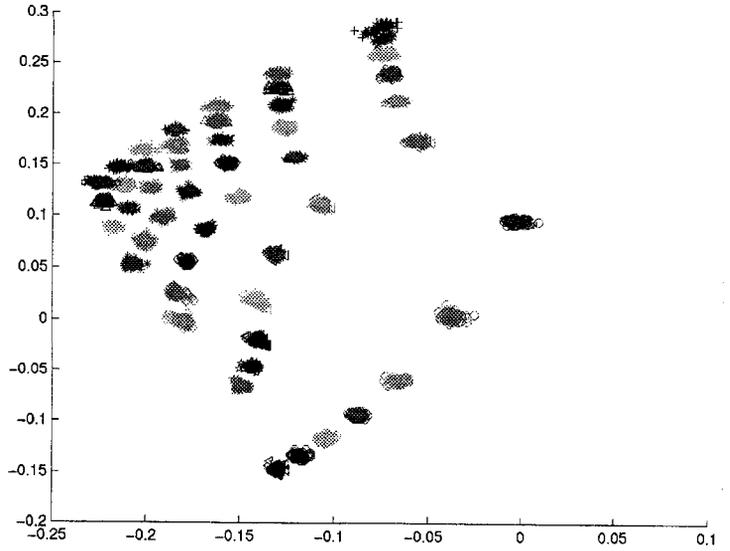
### ANNEXE 1



a) Signaux magnétiques générés par deux objets appartenant à deux classes différentes.

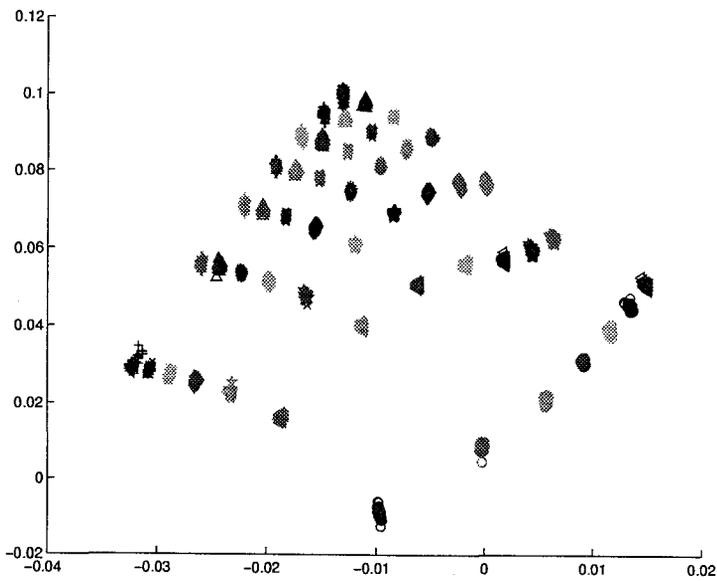


b) Signaux magnétiques générés par deux objets appartenant à la même classe.



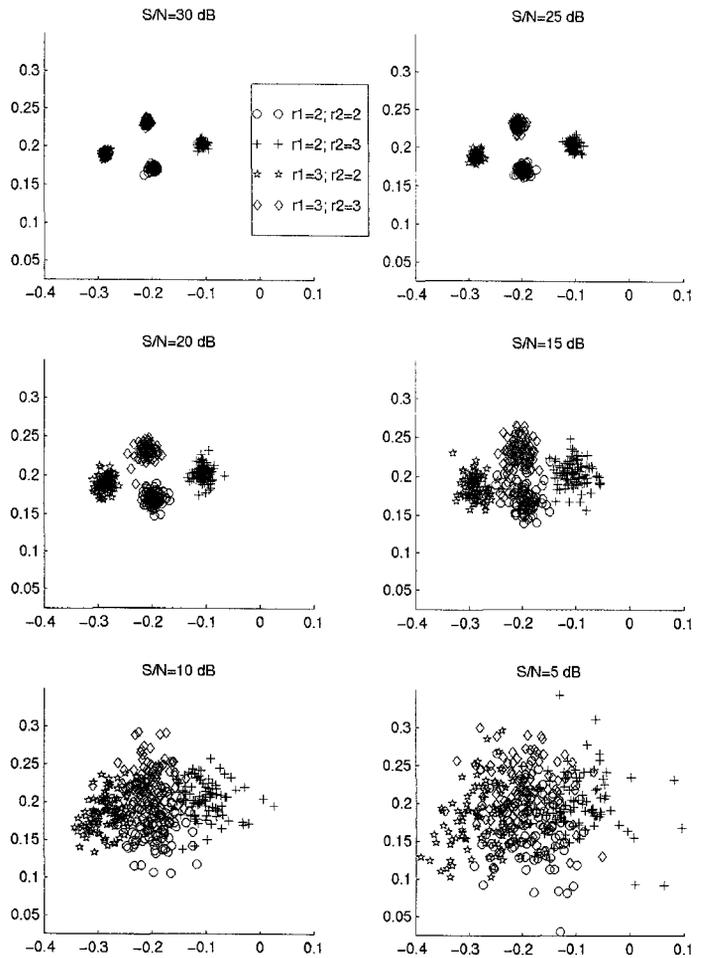
b) L'image bidimensionnelle des 49 classes pour les paramètres réunis et un rapport  $S/B = 30$  dB

**ANNEXE 2**



a) L'image bidimensionnelle des 49 classes pour les paramètres réunis, sans bruit.

**ANNEXE 3**



**ANNEXE 4**

Le principe des méthodes d'estimation non-paramétriques est de réaliser une partition de l'espace des  $N$  vecteurs d'entrée en un nombre suffisamment grand de régions, afin que la densité de probabilité ne varie pas à l'intérieur de celles-ci. Si on impose que le volume de chaque région soit le même,  $V_N$ , alors on obtient l'estimation de Parzen. Si, par contre, on impose que le nombre de vecteurs de chaque région soit le même,  $K_N$ , alors on obtient la méthode  $K$ -NN [7].

La règle de décision est très simple et peut être facilement généralisée au cas de plusieurs classes. Ainsi, si  $V_K(Y)$  représente le voisinage d'ordre  $K$  du vecteur  $Y$  et :

$$K_j(Y) = \text{card}\{Y_i | Y_i \in \omega_i, Y_i \in V_K(Y)\} \quad (A4.1)$$

alors :

$$K_i(Y) = \max_{j=1, \dots, M} [K_j(Y)] \Rightarrow Y \in \omega_i \quad (A4.2)$$

La règle de décision consiste alors à classifier le vecteur inconnu  $Y$  dans la classe qui a le nombre le plus important d'éléments parmi ses  $K$  plus proches voisins.

Si seulement un représentant  $P_i$  de chaque classe (par exemple le vecteur moyenne) est pris en compte, la classification sera réalisée en fonction du représentant le plus proche (NNP).

Les performances de la méthode  $K$ -NN peuvent être sensiblement améliorées si on utilise sa variante floue. La notion centrale dans la théorie de la logique floue est celle de « coefficient d'appartenance » d'un élément (vecteur) à un ensemble (classe). Ce coefficient peut prendre toute valeur entre 0 et 1, à condition que la somme des coefficients d'appartenance à toutes les classes vaille 1, alors que, dans la logique classique, l'appartenance à une certaine classe doit être précisée à coup sûr.

Cette manière, beaucoup moins contraignante, de définir l'appartenance est mieux adaptée à la réalité physique, car le passage d'une forme à l'autre est souvent continu.

Généralement, les algorithmes de classification, qui utilisent la logique floue, comprennent trois étapes : la « fuzzyfication », le traitement proprement-dit et la « défuzzyfication ». La première étape consiste à associer des coefficients d'appartenance aux classes pour les vecteurs d'apprentissage. Dans le cas de la méthode  $K$ -NN floue deux variantes de « fuzzyfication » sont généralement utilisées :

$$u_{ji} = \frac{1}{\sum_{i=1}^M \frac{1}{\|Y_i - P_j\|^{2/\delta}}} \quad (A4.3)$$

$$u_{ji} = \frac{K_j^{(i)}}{K_F} \quad (A4.4)$$

$P_i$  et  $P_j$  sont les prototypes des classes  $i$  et  $j$  tandis que  $K_j^{(i)}$  mesure le nombre de voisins de la classe  $j$  et  $K_F$  est le nombre de voisins les plus proches de  $Y_i$ .

Dans les relations ci-dessus  $\delta$  représente une constante convenablement choisie (typiquement  $\delta = 1$ ) et  $u_{ji}$  représente le coefficient d'appartenance à la classe  $\omega_j$  du vecteur d'apprentissage  $Y_i$ .

La première variante est basée sur les distances du vecteur  $Y_i$  au prototype de chaque classe (sa moyenne le plus souvent), tandis que la deuxième, quant à elle, mesure le poids des plus proches voisins appartenant à la classe  $\omega_j$  par rapport au nombre total,  $K_F$ , de voisins les plus proches de  $Y_i$ .

La « fuzzyfication » peut être assimilée à une phase d'apprentissage. Dans la phase suivante, le traitement, les coefficients d'appartenance du vecteur inconnu  $Y$  aux classes  $\{\omega_j\}$ ,  $j = 1 \dots M$ , sont calculés par la relation :

$$u_j(X) = \frac{\sum_{l=1}^K u_{jl} \cdot \frac{1}{\|X - X_l\|^{2/\delta}}}{\sum_{l=1}^K \frac{1}{\|X - X_l\|^{2/\delta}}} \quad (A4.5)$$

où  $K$  représente le nombre de voisins les plus proches considérés pour  $Y$ .

Enfin, dans la dernière étape, une décision est prise quant à la classe d'appartenance du vecteur inconnu, selon la règle :

$$u_i(Y) = \max_j \{u_j(Y)\} \Rightarrow Y \in \omega_i \quad (A4.6)$$

Ceci équivaut à un processus de « défuzzyfication », car, après avoir appliqué cette règle, le vecteur d'entrée est assigné définitivement à une certaine classe. Néanmoins, ses coefficients d'appartenance aux classes peuvent être conservés, comme une mesure de confiance pour les résultats de la classification. Ceci différencie cette méthode d'autres méthodes floues, comme par exemple le fuzzy-perceptron. Ainsi, bien que ce dernier utilise pendant l'entraînement les coefficients d'appartenance aux classes des vecteurs d'apprentissage, il ne fournit pas explicitement, à la sortie, une mesure de confiance pour la décision prise.

**ANNEXE 5**

Les réseaux de neurones artificiels sont couramment utilisés pour des tâches de classification. Classiquement, les réseaux sont utilisés comme mémoires associatives [11]. Leur rôle peut être de reconstruire une information bruitée (*auto-association*) ou d'associer deux types d'information différents (*hétéro-association*). Il y a une grande variété de réseaux de neurones tant pour la classification supervisée que pour la classification non supervisée. Ils sont tous construits à partir d'un élément de base, le neurone artificiel, qui calcule généralement un produit scalaire ou une fonction « distance ».

*A. Le perceptron multicouches*

Le neurone élémentaire de ce type de réseau calcule le produit scalaire entre le vecteur d'entrée ( $X$ ) et son vecteur poids ( $W$ ). Ensuite, la sortie est déterminée à l'aide d'une fonction non linéaire dérivable (sigmoïde, tanh, etc.). Le schéma d'un neurone simple est représenté sur la figure A5.1.

Les neurones sont organisés en une couche d'entrée, une couche de sortie et une ou deux couches cachées. Toutes les sorties des neurones d'une couche sont connectées aux entrées de tous les neurones de la couche suivante, comme indiqué sur la figure A5.2. La connexion entre deux neurones,  $i$  et  $j$ , est pondérée par un poids  $w_{ij}$ . Tous les poids sont initialement fixés d'une manière aléatoire

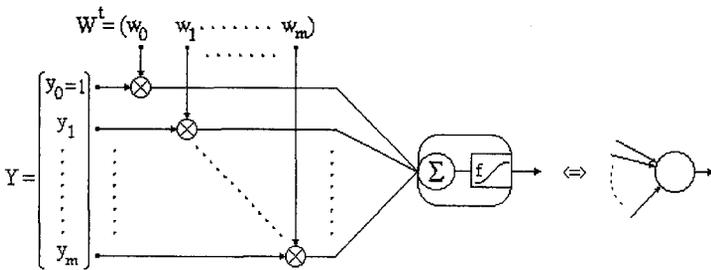


Figure A5.1. – Le schéma d'un neurone du MLP et sa représentation.

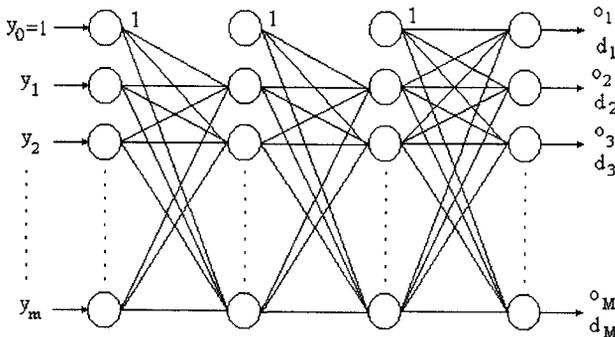


Figure A5.2. – Le schéma général d'un MLP.

ou plus élaborée (par exemple en utilisant la règle de Nguyen-Widrow [12]) et ils atteignent des valeurs stables à la fin du processus d'apprentissage. L'apprentissage consiste à présenter au réseau, à plusieurs reprises (appelées « époques », toutes les paires :

- vecteur d'apprentissage  $Y = (y_0, y_1, \dots, y_m)^T$ ;
- sortie souhaitée  $D = (d_0, d_1, \dots, d_M)^T$ ;

et à modifier les poids au sens du gradient descendant afin de minimiser l'erreur quadratique entre les sorties réelles  $\{o_j\}$  et les sorties souhaitées  $\{d_j\}$ . On obtient ainsi la règle d'apprentissage *delta généralisée* ou l'algorithme de *rétropropagation*, démontré pour la première fois par Rumelhart, en 1986. Plus précisément, si la fonction d'activation du neurone est la fonction *sigmoïde*, représentée sur la figure A5.3 (la constante  $c$  vaut typiquement 3), alors, selon [11], la règle de mise à jour des poids au cours de l'apprentissage s'écrit :

$$\begin{cases} W_{ij}(t+1) = W_{ij}(t) + \eta \cdot \delta_j(t) \cdot o_i(t) \\ \delta_j(t) = c \cdot o_j(t) \cdot [1 - o_j(t)] \cdot [d_j - o_j(t)] & \text{pour les neurones de sortie} \\ \delta_j(t) = c \cdot o_j(t) \cdot [1 - o_j(t)] \cdot \sum_l \delta_l(t) \cdot w_{jl}(t) & \text{pour les autres neurones} \end{cases} \quad (A5.1)$$

où  $\eta$  désigne la vitesse d'apprentissage (typiquement 0.01).

Malgré son importance théorique, le MLP est pratiquement inutilisable dans sa forme classique à cause du fait qu'il converge trop lentement et demeure souvent piégé dans un minimum local. De nombreuses améliorations ont alors été proposées afin de rendre robuste l'algorithme. Des solutions heuristiques, comme l'addition d'un « *momentum term* » ou l'utilisation d'une *vitesse d'apprentissage adaptative*, ou bien mathématiques, comme celle

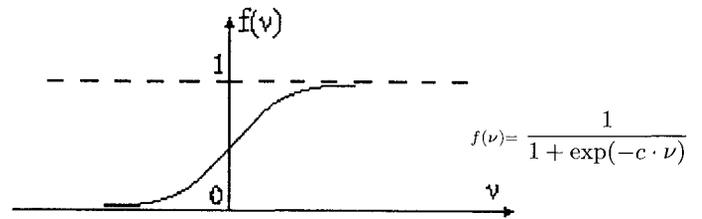


Figure A5.3. – La fonction sigmoïde

de *Levenberg-Marquardt* [13], sont parmi les plus connues. Néanmoins, bien qu'il soit intensivement utilisé dans des domaines très variés, son exploitation est parfois alourdie, car il n'est pas encore possible de déterminer automatiquement certains de ses paramètres fondamentaux.

### B. Le réseau à fonction à base radiale

Les difficultés du MLP sont dues principalement à la saturation de sa fonction d'activation. Comme plusieurs auteurs l'ont déjà remarqué, ceci correspond à une attitude rigide, peu flexible de son « raisonnement », ce qui explique qu'il soit assez souvent piégé dans des minima locaux, malgré toutes les améliorations apportées, et qu'il apprenne très lentement. Certaines techniques d'optimisation (algorithmes génétiques, recuit simulé) pourraient être envisagées, mais le prix à payer est l'augmentation de la complexité des calculs.

Ces inconvénients peuvent être en principe éliminés par l'utilisation d'une fonction d'activation non-saturante. Cette idée apparaît sous la forme des nouveaux concepts comme « représentation scalaire distribuée » en [15] ou « neurone intelligent » en [16], mais la structure de base reste celle du MLP.

Un nouveau type de réseau qui utilise cette idée et constitue une alternative viable au MLP est le réseau à fonction à base radiale (RBF). Les RBF (fig. A5.4) sont des réseaux à deux couches; la première (la couche cachée) contient des neurones avec une fonction d'activation de type « Gaussien » :

$$\Phi(\nu) = \exp(-\nu^2/\beta) \quad (A5.2)$$

où  $\beta$  est une constante positive.  $\nu$  représente la distance entre le vecteur d'entrée,  $Y$ , et le vecteur poids du neurone,  $C$ , c'est-à-dire :  $\nu = \|Y - C\|$ .

Les neurones de la couche cachée réalisent ainsi une partition de l'espace des vecteurs d'entrée en plusieurs régions de « responsabilité ». Le centre de chaque région est constitué par le vecteur poids du neurone associé et le rayon est déterminé par la valeur de  $\beta$ . La séparation des classes ne se fait donc plus par des hyperplans (comme dans le cas du MLP), mais par des hypersphères. Les neurones de la deuxième couche ont une fonction d'activation linéaire. On peut donc regarder le réseau RBF comme une combinaison entre un réseau de type Kohonen (la couche cachée) et un réseau de type MADALINE (la couche de sortie).

## Comparaison de méthodes de classification

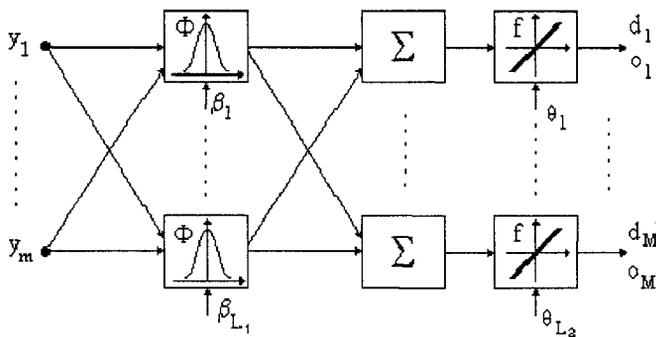


Figure 7. – Le schéma bloc du réseau RBF.

Il est facile de constater que les sorties du réseau sont alors données par :

$$o_j = f \left( \theta_j + \sum_{i=1}^N w_{ij} \Phi(\|X - C_i\|) \right) \quad (A5.3)$$

$$= f \left( \theta_j + \sum_{i=1}^n w_{ij} \exp \left( -\frac{\|X - C_i\|^2}{\beta_i} \right) \right)$$

L'algorithme d'apprentissage sera de type Kohonen pour la première couche et de type Widrow-Hoff pour la deuxième. Néanmoins, une méthode mieux adaptée pour déterminer d'une manière optimale les poids (ou les centres)  $\{C_i\}$  est développée dans [17] sous le nom d'*OLS* (Orthogonal Least Squares Learning Algorithm). Dans la procédure utilisée, les poids  $\{C_i\}$  sont calculés successivement afin de minimiser la variance de chaque sortie autour de la valeur souhaitée.

Les neurones de la couche de sortie combinent les neurones responsables de plusieurs régions de l'espace des vecteurs d'entrée pour former chaque classe. Ainsi, une fois les  $\{C_i\}$  fixés, il reste à déterminer seulement les poids de la deuxième couche  $w_{ij}$ , qui est un problème linéaire, d'où la rapidité de l'algorithme d'apprentissage pour ce type de réseau.

### C. Le réseau LVQ

La structure d'un réseau LVQ [18] ressemble à la structure du réseau RBF, mais le principe de fonctionnement pour la couche cachée est différent. Ainsi, les neurones appartenant à cette couche, appelés « neurones compétiteurs », sont entraînés par la règle de Kohonen modifiée. Chaque neurone calcule la distance entre le vecteur d'entrée et son vecteur poids. Le neurone qui est le plus proche du vecteur d'entrée est déclaré « vainqueur » et sa sortie passe à 1, tandis que les sorties des autres neurones restent à 0. Son poids est alors modifié selon la règle de Kohonen :

$$\Delta W_i = \eta \cdot (Y - W_i) \quad (A5.4)$$

Il est facile de comprendre qu'à la suite de cette mise à jour, le neurone vainqueur sera encore plus proche de ce vecteur d'entrée, donc il devient de plus en plus représentatif pour lui et pour ses voisins. L'espace des vecteurs d'entrée est ainsi partagé entre les neurones compétiteurs, sous la forme de plusieurs sous-ensembles (clusters), dont ils deviennent les représentants.

La règle de Kohonen modifiée prend en compte, pendant le processus d'entraînement, l'information sur l'appartenance aux classes des vecteurs d'apprentissage. Ainsi, la relation (A5.4) est appliquée telle quelle, si et seulement si le neurone « vainqueur » représente une sous-classe de la classe du vecteur d'apprentissage courant. Sinon, la relation (A5.4) est appliquée avec le signe moins, ce qui a pour effet d'éloigner ce neurone du vecteur d'entrée et de réduire ses chances de gagner à nouveau, à la prochaine époque.

Par conséquent, des groupes de neurones sont répartis vers les classes définies de façon supervisée par la règle de Kohonen modifiée. Ensuite, à l'intérieur de chaque classe, les neurones appartenant à chaque groupe entrent en compétition pour former des sous-classes. Plus le nombre de neurones est important, plus l'information contenue par une classe est mieux représentée en sortie du classifieur.

## BIBLIOGRAPHIE

- [1] A. Quinquis : *Classification des signaux magnétiques TBF sous contrainte de décision*. Thèse de doctorat d'Université de Bretagne Occidentale, 1989.
- [2] R. Blanpain : *Réseaux de magnétomètres. Traitement du signal et de l'information*. ENSIEG/SEE, Grenoble, 1994.
- [3] A. Quinquis, S. Rossignol : *Noise Reduction, with a Noise Reference, of Underwater Magnetic Signals*. Digital Signal Processing, p. 240-248, Vol. 6, Nr. 4, 1996.
- [4] L. Marple : *Digital Spectral Analysis with Applications*. Prentice-Hall, 1987.
- [5] R.R. Coifman, M.V. Wickerhauser : *Entropy Based Algorithms for Best Basis Selection*. IEEE Trans. Inform. Theory IT, vol. 38, p. 713-718, 1992.
- [6] G. Saporta : *Probabilités, analyse des données et statistique*. Technip, Paris, 1990.
- [7] V. Neagoe : *Pattern Recognition Theory*. Roumanian Academy Edition House, Bucarest, 1994.
- [8] E. Backer : *Computer-Assisted Reasoning in Cluster Analysis*. Prentice Hall, 1995.
- [9] J.-P. Cocquerez et al. : *Analyse d'images : filtrage et segmentation*. Masson, 1995.
- [10] S. Banks : *Signal Processing, Image Processing and Pattern Recognition*. Prentice Hall, 1990.
- [11] B. Kosko : *Neural Networks for Signal Processing*. Prentice Hall, 1990.
- [12] D. Nguyen, B. Widrow : *Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights*. International Joint Conference of Neural Networks, vol.3, p. 21-26, 1990.
- [13] R. Fletcher : *Practical Methods of Optimisation*. Wiley, 1987.
- [14] J.F. Mari, A. Napoli : *Aspects de la classification*. Rapport de recherche 2909. INRIA, 1996.
- [15] G. Burel : *Réseaux de neurones en traitement d'images*. Thèse de doctorat d'Université de Bretagne Occidentale, 1991.
- [16] B.W. Dahanayake et al. : *Quick Learner : Smart Neural Nets*. Proceedings of ICSPAT, Boston, p. 1398-1402, 1995.
- [17] S. Chen et al. : *Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks*. IEEE Trans. on NN, Vol. 2, No. 2, p. 302 - 309, 1991.
- [18] T. Kohonen : *Self-Organisation and Associative Memory*. Springer-Verlag, 1984.

Manuscrit reçu le 28 mai 1997.

### LES AUTEURS

#### André QUINQUIS



André Quinquis a reçu son diplôme de doctorat en électronique en 1989 à l'Université de Brest; Depuis 1992, il a rejoint l'Ecole Nationale Supérieure des Ingénieurs des Etudes et Techniques d'Armement (EN-SIETA), où il enseigne le traitement du signal et développe une activité de recherche orientée vers la théorie de la décision dans des environnements non stationnaires à partir des méthodes temps-échelle et temps-fréquence. Depuis 1996, il est auteur de 4 ouvrages de traitement du signal publiés chez HERMES.

#### Emanuel RADOI



Emanuel RADOI est diplômé de l'Académie Technique militaire de Bucarest (Roumanie). En 1997, il a obtenu un DEA d'électronique à l'Université de BREST. Il est assistant à l'ATM de Bucarest et prépare une thèse doctorat. Ses principaux thèmes de recherche sont le traitement du signal, le traitement haute résolution et la reconnaissance des formes.