ELSEVIER

Signal Processing 82 (2002) 547–561

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Neural methods for antenna array signal processing: a review ☆

K.-L. Du[a,*], A.K.Y. Lai[b], K.K.M. Cheng[b], M.N.S. Swamy[a]

[a] Department of Electrical and Computer Engineering, Concordia University, 1455 de Maisonneuve Bvd West, Montreal, Canada H3G 1M8

[b] Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong, China

Received 27 September 2000; received in revised form 5 June 2001

Abstract

The neural method is a powerful nonlinear adaptive approach in various signal-processing scenarios. It is especially suitable for real-time application and hardware implementation. In this paper, we review its application in antenna array signal processing. This paper also serves as a tutorial to the neural method for antenna array signal processing. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Software radio; Beamforming; Smart antenna; Direction-of-arrival; Multi-layer perceptron; Backpropagation; Hopfield network; Radial basis function network; Principal component analysis; Cross-relation network; Antenna array signal processing; Fuzzy neural network

1. Introduction

Artificial intelligence methods such as the neural method [37] and the genetic algorithm method [32] have evolved into powerful tools in various fields in the past few decades. Due to their strong numerical approximation capabilities, they are widely used in identification and optimization. The research in antenna arrays is very active due to its military and commercial applications, and there is a vast literature available [31]. Research in antenna array signal processing (AASP) has been mainly focused on the direction-of-arrival (DoA) estimation and beamforming. The DoA problem is considered as a mapping from the space of the sensor output to the space of DoA, while the beamforming function is an inversion of the DoA estimation function.

Some powerful and high-resolution methods are available, such as MUSIC [83] and ESPRIT [79] for DoA estimation, and minimum-variance distortionless response (MVDR), recursive least square (RLS) for beamforming [31]. MUSIC [83] has the advantages of high resolution for signals with small angular separation and good performance under low signal-to-noise ratios (SNR), but it suffers from its high sensitivity to the structure of the covariance matrix. These beamforming methods require the knowledge of the DoA of the sources. Once the DoA of the sources is available, the beamforming algorithms can be used to track those interested sources in real time, and null out the other sources as interference by controlling the beampattern of an antenna array in an adaptive way.

Conventional methods are typically linear algebra-based methods, requiring computationally intensive matrix inversion, and cannot meet real-time requirement. They also require calibrated antennas with

☆ This work is partially supported by NSERC of Canada, and partially supported by ITF of Hong Kong.

* Corresponding author. Tel.: +1-514-8483123; fax: +1-514-8482802.

E-mail addresses: kldu@ece.concordia.ca (K.-L. Du), albertlai@cybernation.com.hk (A.K.Y. Lai), kkcheng@ee.cuhk.edu.hk (K.K.M. Cheng), swamy@ece.concordia.ca (M.N.S. Swamy).

PII: S0165-1684(01)00185-2

uniform features, and are sensitive to the manufacturing fault and other physical uncertainties. These techniques make use of the first and second moment information of the data, and the higher-order statistics (cumulant) arising from the correlation between signal sources and imperfect array geometry is missed, thus reducing the performance. Cumulant is useful to describe non-Gaussian signals, and it is also insensitive to additive Gaussian noise.

Neural networks, using simple addition, multiplication, division, and threshold operations in the basic processing element, can be readily implemented in analog VLSI or optical hardware [8,28], or be implemented on special purpose massively parallel hardware. The neural method is typically used in two steps: training and recalling. The network is first trained with known input–output pattern pairs. Although a large training pattern set is required for network training, it can be implemented off-line. After training, it can be used directly to replace the complex system dynamics.

The neural method possesses such advantages as general-purpose nature, massive parallelism and suitable for simple VLSI implementations, nonlinear property, adaptive learning capability, generalization capability, and strong fault-tolerant capability and insensitivity to uncertainty. Some neural network types and suitable layered networks can actually eliminate the need for weight training entirely. In this paper, we review the neural method and its application in AASP.

2. AASP neural models

2.1. Antenna array signal model

Most of the available AASP methods assume that the signal sources are in the far field, and hence the received signals are planar wavefront.

Consider an array of L omnidirectional elements, surrounded by M point sources. $(r_i, \varphi_i, \theta_i)$ is the position of the ith source in the spherical coordinate system, or simply expressed as vector $\mathbf{r}_i$, as shown in Fig. 1. For narrowband signals, the signal induced on the reference element due to the ith source can be expressed as $s_i = m_i(t)\exp(\mathrm{j}2\pi f_0 t)$, where $m_i(t)$ is the complex modulating function. The total measured signal on the lth element is

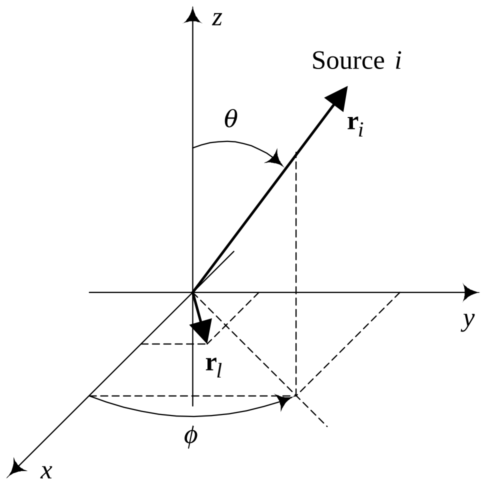

Fig. 1. Coordinate system for the signal model.

$$x_l = \sum_{i=1}^{M} s_i \exp(\mathrm{j}2\pi f_0 \tau_l(\phi_i, \theta_i)) + n_l(t)$$
$$= \sum_{i=1}^{M} d_{il} s_i + n_l(t), \quad (1)$$

where $d_{il} = \exp(\mathrm{j}2\pi f_0 \tau_l(\phi_i, \theta_i))$ is the steering vector of the ith source on the lth element, $\tau_l(\varphi_i, \theta_i) = (\mathbf{r}_l \cdot \hat{\mathbf{r}}_i)/c, \mathbf{r}_l$ is the position vector of the lth element, $\hat{\mathbf{r}}_i$ is the unit position vector of the ith source, c is the speed of wave propagation, and $n_l(t)$ is the random noise component on the lth element, which is assumed to be a zero-mean white Gaussian with variance σ_{n}^2.

For all the sources and array elements, we can write the signal model in the following matrix form:

$$\mathbf{x} = \mathbf{D}\mathbf{s} + \mathbf{n}, \quad (2)$$

where $\mathbf{x} = (x_1, \ldots, x_L)^{\mathrm{T}}$, $\mathbf{s} = (s_1, \ldots, s_M)^{\mathrm{T}}$, $\mathbf{D} = [\mathbf{d}_1 \ldots \mathbf{d}_M]$, $\mathbf{d}_i = (d_{i1}, \ldots, d_{iL})^{\mathrm{T}}, \mathbf{d}_i$ is the steering vector associated with the ith source, $\mathbf{n} = (n_1, \ldots, n_L)^{\mathrm{T}}$, and superscript T denotes the transpose of a matrix.

The structure of a narrowband beamformer is shown in Fig. 2. The beamformer output is given as

$$\mathbf{y}(t) = \mathbf{W}^{\mathrm{H}}\mathbf{x}(t), \quad (3)$$

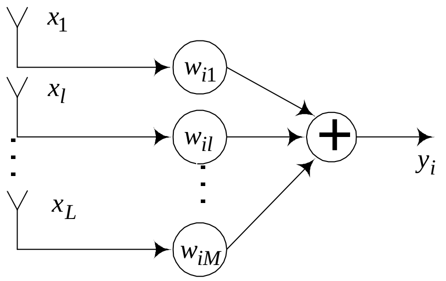

Fig. 2. Structure of the narrowband beamformer.

where $\mathbf{W} = [\mathbf{w}_1 \dots \mathbf{w}_M]$ is an $L \times M$ weight matrix and superscript H denotes the complex conjugate transpose.

If the components of $\mathbf{x}(t)$ can be modeled as a zero-mean stationary process, the mean output power of the beamformer is given by

$$P(\mathbf{W}) = E[\mathbf{y}(t)\mathbf{y}^*(t)] = \mathbf{W}^{\mathrm{H}}\mathbf{R}\mathbf{W}, \tag{4}$$

where $E[\cdot]$ denotes the expectation operator and $\mathbf{R}$ is the array correlation matrix

$$\mathbf{R} = E[\mathbf{x}(t)\mathbf{x}^{\mathrm{H}}(t)] = \mathbf{D}\mathbf{S}\mathbf{D}^{\mathrm{H}} + \sigma_{\mathrm{n}}^2\mathbf{I} \tag{5}$$

and $\mathbf{S} = E[\mathbf{s}\mathbf{s}^{\mathrm{H}}]$, an $M \times M$ matrix denotes the source correlation. For uncorrelated sources, $\mathbf{S}$ is a diagonal matrix.

Eq. (5) can be eigendecomposed as $\mathbf{R} = \boldsymbol{\Sigma}\boldsymbol{\Lambda}\boldsymbol{\Sigma}^{\mathrm{H}}$, $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \dots, \lambda_M, \lambda_{M+1}, \dots, \lambda_L)$, with M signal eigenvalues and L–M noise eigenvalues, and $\boldsymbol{\Sigma}$ is composed of L eigenvectors.

According to the optimization objectives such as maximizing the output SNR or minimizing output interference, there are many AASP methods available [31].

2.2. DoA estimation and beamforming

The AASP mainly involves two classes of problems, DoA estimation and beamforming. Some systems need only DoA estimation to detect the signals, such as the radar or sonar systems, while others such as the mobile communication systems need beamforming to acquire the signals.

Beamforming methods are generally classified into two categories: beamforming based on the DoA estimated by a calibrated array, and beamforming based on a known training signal transmitted by the user. There are also some blind beamforming methods, which do not require the knowledge of the DoA or training sequence [96].

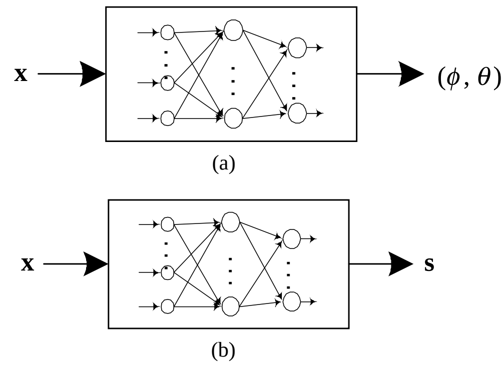

Fig. 3. AASP neural models: (a) DoA estimation; (b) beamforming.

Many successful methods together with their merits and demerits are well expounded in [31]. Generally, these methods cannot meet the requirements of real time and multi-source tracking requirements. The neural method is a typical adaptive method, and has proved to be a powerful general-purpose method.

The neural models for the two classes of problems are shown in Fig. 3. An antenna array serves as a nonlinear mapping from signal sources to array measurement, while the two classes of problems are both inversion problems, trying to solve for the signal sources from the array measurements. The DoA problem aims to get the DoA of signals from the measurement of the array output, while beamforming technique tries to recover the original signal of the desired source. For an antenna array system, a neural network is first trained, which then performs the DoA estimation or beamforming. The block diagram is shown in Fig. 4.

A typical architecture of a neural processor is displayed in Fig. 5. The network computation structure is composed of input preprocessing for antenna measurement, a neural network to perform the inversion, and output postprocessing. Input preprocessing is used to remove redundant or irrelevant information to reduce the size of the network, and hence to reduce the dimensionality of the signal parameter space. Postprocessing the output node information yields the desired information. For example, for the DoA problem, the

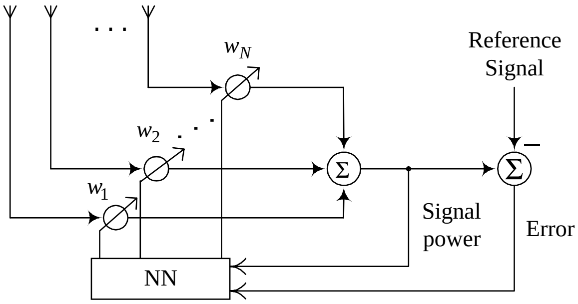

Fig. 4. Block diagram of an antenna array system.

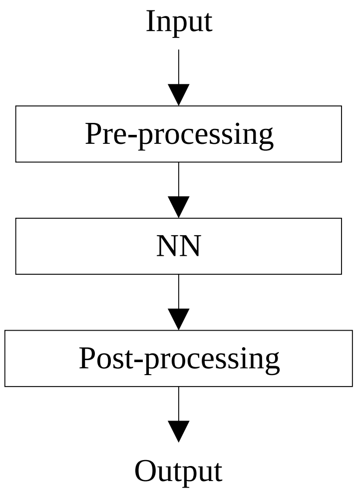

Fig. 5. Neural processor architecture.

initial phase contains no information about the DoA and can be eliminated at the preprocessing phase. The input of the network can also be normalized since the signal gain does not affect the detection of the DoA.

3. Neural networks and their AASP applications

Any neural network with universal approximation capability can be used for AASP. This section is a review of the applications of the neural method in AASP. Currently, most applications have been concerned with the DoA problem, and some with beamforming.

Real-time processing requires that the network should have a constant processing delay regardless of the number of inputs, and a minimum number of layers. As the number of input nodes increases, the size of the network layers should grow at the same rate without additional layers.

Neural networks can be divided into feedforward and recurrent classes according to their connectivity. Powerful learning algorithm is one of the main strengths of the neural approach. Recurrent networks cannot guarantee stability, and one needs to construct a Lyapunov function to test its stability. Recurrent networks cannot be trained by the standard back-propagation (BP) rule. The training of the network is typically accomplished using examples. Training can roughly be divided into supervised, unsupervised, and reinforcement learning. Supervised learning is based on a direct comparison between the actual output of a network and the desired output. Reinforcement learning is a special case of supervised learning where the exact desired output is unknown. It is based only on the information of whether or not the actual output is close to estimation. Unsupervised learning is solely based on the correlations among the input data. No information on the correct output is available for learning. A detailed review of the various learning algorithms can be found in [14].

The evolutionary training approach is attractive since it can handle the global search problem better in a vast, complex, multi-modal, and nondifferentiable surface. It is not dependent on the gradient information of the error (or fitness) function, and thus is particularly appealing when this information is unavailable or very costly to obtain or estimate. Evolutionary algorithms are widely used to train neural networks, and are generally much less sensitive to the initial conditions. They always search for a globally optimal solution, while a gradient-descent algorithm can only find a local optimum in a neighborhood of the initial solution [98].

3.1. *Multilayer perceptron method*

The multilayer perceptron (MLP) has strong classification capabilities, and is a universal approximator with a three-layer topology [17,43]. The architecture of the MLP is shown in Fig. 6, where $g(\cdot)$ represents the sigmoid relation. The MLP is very efficient for function approximation in high dimensional spaces [6]. The convergence rate of the MLP is independent of the input space dimensionality, while the rate of the

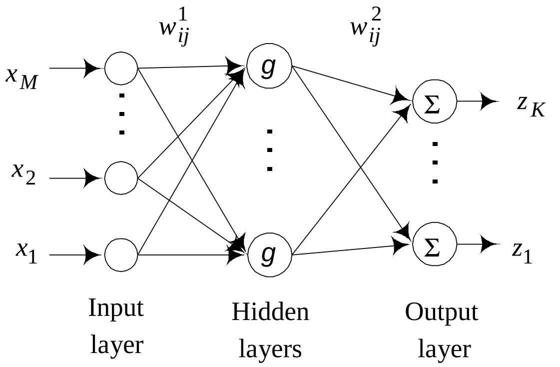

Fig. 6. MLP architecture.

error convergence of polynomial approximators decreases with the input dimensionality. The MLP with the BP learning rule is one of the most widely used networks. The BP is a supervised gradient-descent technique wherein the squared error between the actual output of the network and the desired output is minimized. It is prone to local minima.

The BP is a degenerate form of the extended Kalman filter (EKF) [81], which is an optimum filter for a linear system resulting from the linearization of a nonlinear system. It attempts to estimate the state of a system that can be modeled as a linear system driven by an additive white Gaussian noise. The weights of the MLP are the states that the Kalman filter tries to estimate, and the network output is the measurement acting as the input to the Kalman filter. It has been shown in [81] by simulation that the EKF requires orders of number of floating point operations to achieve the same accuracy as compared with the BP or the momentum algorithm [81]. There are some fast versions of the BP algorithm, such as the recursive BP algorithm [75].

The MLP is purely static and is incapable of processing the time information. One can add a time window over the data to act as the memory for the past. Local recurrent globally feedforward networks can be used specifically in time series modeling [92].

For a given problem, there is a set of training vectors $\mathbf{X}$, such that for every vector $\mathbf{x} \in \mathbf{X}$ there is a corresponding desired output vector $\mathbf{d} \in \mathbf{D}$, where $\mathbf{D}$ is the set of the desired output. The error E_p is defined as

$$E_p = \frac{1}{2} ||\mathbf{d}_p - \mathbf{z}_p||^2, \tag{6}$$

where subscript p represents the pth desired output, $||\cdot||$ denotes the norm in the Euclidean space. The total error is defined as $E_T = \sum_{p=1}^{P} E_p$, where P is the cardinality of $\mathbf{X}$. The BP algorithm is defined as

$$\Delta w(t) = -\eta \frac{\partial E_p}{\partial w} + \alpha \Delta w(t-1), \tag{7}$$

where η is the learning rate, α is the momentum factor, and w represents any single weight in the network. When $\alpha \neq 0$, it is called the momentum method; for $\alpha = 0$, it is the BP algorithm.

Some of the researchers have classified the signal DoA into prescribed sectors. In [39,94] the authors have implemented the MLP with the BP rule to classify the signal DoA into a number of signal sectors with training input taken directly from the array elements, and the output is a binary code vector. In [11,12] the authors have used the MLP with the Levenberg–Marquart (LM) learning algorithm to track a full azimuthal sector with only three array elements. One hidden layer of the MLP is composed of Adaline neurons. The LM algorithm is much faster but requires more memory than the BP rule. The authors in [12] have used two parallel MLP's and a subsequent Kohonen network [52]. The system classifies the DoA into one of the ten 36° intervals covering the full 360° azimuth range. In [11] the MLP has been used to classify the DoA into three sectors.

A complex EKF algorithm to replace the BP rule has been proposed in [77]. The authors have used third-order cumulants of the received signals evaluated at different combinations of DoA as training input to improve the system. An MLP with simulated annealing rules [50] has been presented in [99] to solve the DoA problem. In [62] the authors have applied a simplified MLP with the BP rule to solve the DoA problem. The MLP with the BP rule has been used in [56] for low angle tracking, which is a multipath signal environment for radar systems. A convex constrainted algorithm (CR) has been presented in [89] to replace the BP rule, and has been applied to the DoA problem. The CR is based on nonlinear optimization and convexity analysis. The training algorithm incorporates both the first- and second-order optimality conditions in each iterative search step to prevent the algorithm from a premature termination due to the low gradient values. The simulation result is better than that of the LM method. In [90], the authors have applied the MLP

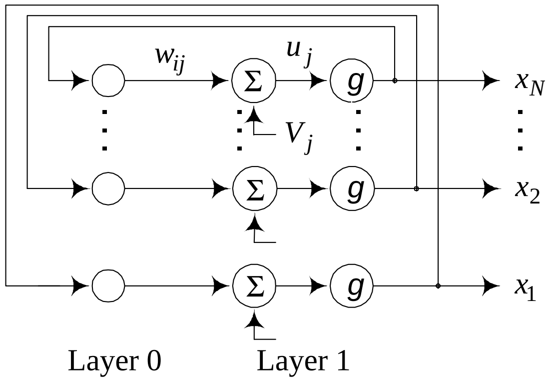

Fig. 7. Hopfield network.

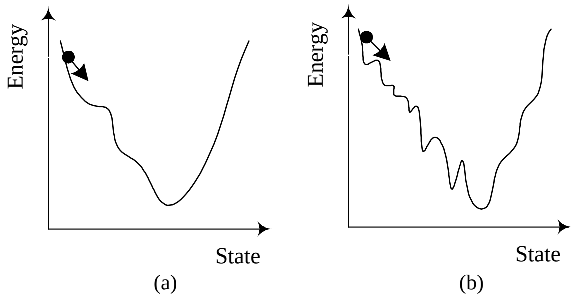

Fig. 8. Change in the energy landscape through gain β': (a) low gain β smoothes the surface; (b) high gain β reveals all details in the surface.

with the BP rule to directly map the noise eigenvectors into the spectra minima to improve the MUSIC algorithm [83].

The MLP with the fast convergent block recursive least-squares (BRLS) learning rule has been applied to beamforming in [19]. In [67], a beamformer using the MLP with the BP rule is compared with the conventional beamformer using the LMS algorithm for HF data transition. The MLP with the EKF rule has been used in [74] to separate signals of simultaneous multiusers. In [51], the authors have applied the MLP with the BP rule to the case of two-microphone beamforming. Replacing the activation function of the MLP by a polynomial leads to the finite-order Volterra filter with which the global optimum can be obtained. In [82] a biology-inspired neuron model has been explored with hardware emulator to solve the real-time adaptive beamforming with a good fault-tolerant capability.

3.2. Hopfield method

The Hopfield network [45] is a two-layered fully interconnected recurrent neural network. The structure of the network is shown in Fig. 7. The input layer only collects signals from the feedback of the output layer. Due to recurrence, it remembers cues from the past and does not appreciably complicate the training procedure. This network is considered as a stable dynamic system in which the forward and backward paths will cause the processing of the network to converge to a stable fixed point.

The energy minimization ability of the Hopfield network is used to solve optimization problems. In this approach, the synaptic weight matrix and the neuron bias values are set such that the global energy minima state of the network corresponds to the optimum solution of the optimization problem. It is widely used in quadratic programming optimization problems. The cellular neural network (CNN) is a generalization of the Hopfield network, and can be used to solve a more generalized optimization problem [15]. An important part of designing a neural optimization approach is the incorporation of constraints. The common method is to add a penalized energy item of constraints, and then to use the Lagrange multiplier method.

The basic Hopfield network implements a gradient descent procedure, and always gets trapped to the nearest local minima of the initial random state. There are two methods to avoid this. One can inject a small amount of noise to the output of the neurons and/or to the synaptic weight values during the update step of the algorithm. This perturbation will make the system escape from the local minima. The second strategy is to change the sigmoid gain parameter β, by starting from a low gain and gradually increasing the gain, as shown in Fig. 8. This is analogous to the cooling process of simulated annealing [50].

The high interconnection in physical topology makes this network especially suitable for analog implementation. The Hopfield model has inspired many efforts at analog VLSI implementation and computational experiment because of the deterministic formulation instead of the stochastic one. The analog implementations have a network time constant and a convergence time, which are of the order of a few nanoseconds [44]. A drawback of this network is the necessity to update the complete set of network coefficients caused by the signal change.

The continuous Hopfield network model is given as

$$\mathrm{d}u_i/\mathrm{d}t = \sum_{j=1}^{N} w_{ij}x_i + V_i, \tag{8.1}$$

$$x_i = g(u_i), \tag{8.2}$$

where w_{ij} is a connection synaptic weight between a pair of neurons i and j with the assumption of $w_{ij}=w_{ji}$ and $w_{ii}=0$, N is number of neurons, $u_i(t)$ is the calculated net input value of ith neuron, $x_i(t)$ is the output activation value of ith neuron, V_i is an external input to the neuron, and $g(\cdot)$ is the neuron transfer function, typically taking the form of sigmoid functions

$$g(u_i, \beta) = 1/[1 + \exp(-\beta u_i)], \tag{9}$$

where β is used as the gain of this transfer function.

The discrete differential equation takes the form

$$u_i(t+1) = \sum_{j=1}^{N} w_{ij}x_j(t) + V_i, \tag{10.1}$$

$$x_i(t+1) = g(u_i(t+1)). \tag{10.2}$$

Each neuron in the second layer sums the weighted inputs from other neurons to calculate its new internal activation u_i, then applies an activation function to u_i and broadcasts the result along the connections to other neurons. If the connection weights are chosen carefully, this network is stable and converges to neuron activation value x^* which minimizes the energy level in a quadratic form

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}x_i^* x_j^* - \sum_{i=1}^{N} V_i x_i^*. \tag{11}$$

The Hopfield network cannot compute complex values. The AASP problems are, however, in essence complex-valued optimization problems, and thus one needs to separate the real and the imaginary parts of the complex computation.

In [46] the authors have tried to implement an iterated descent (ID) procedure to the Hopfield network to solve the DoA problem. The ID procedure combines the benefits of the analog Hopfield network with the ability of the Boltzmann machine (BM) and the stochastic network (SN) to dislodge the network from the local minimum. Hardware implementation is also explored on the transputer array. In [33], the Hopfield network was used in the DoA problem and was extended to the wideband case. Averaging interconnection strengths and bias terms over several received data snapshots from the sensor array was used as a smoothing technique.

The Hopfield network has been applied in [84] to solve the multi-target multi-sensor passive tracking problem, and a multiple elastic model (MEM) has been proposed to obtain global optimum to this combinatorial optimization problem. A recurrent Hopfield-like network has been used in [40] for real-time DoA tracking and polarization of any incident sources for polarized EM waves using blind separation of sources. A random search algorithm with a Hopfield network has been employed in [2] to track desired signal and suppressing interfering sources. In [93] the DoA problem has been solved by employing the Hirose complex-valued network [41], which is a complex version of the Hopfield model.

In [10,100] the Hopfield network has been applied to the problem of MVDR beamforming, and an analog switched-capacitor implementation has been proposed based on the programmable capacitor array technique. A Hopfield network-based neural beamformer combined with chaotic CDMA has been proposed in [49]. A symbolic chaotic signal is used as the spreading sequence for the spreading spectrum communication. The chaotic signal has the property of low auto-/cross-correlation, which has been used to remove the periodic restriction on conventional spreading code of CDMA and thus make truly random signals possible. The method is suitable for hardware implementation. A neural beamformer, composed of a Hopfield network and Kohonen-type organization direction associate network (DAN), has been described in [88]. The DAN uses a MLP with the BP rule to learn associate direction with beamformer weight pattern. In [35] the authors have applied the Hopfield network to interference cancellation [78]. A modified Hopfield network has been proposed in [38] to compute the matrix inversion in the C-CAB beamforming method presented in [96].

3.3. *Radial-basis function network method*

The radial-basis function network (RBFN) is a three-layered feedforward network, and is shown

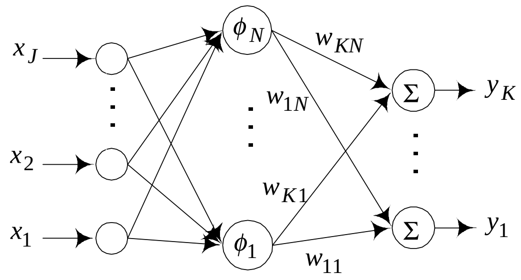

Fig. 9. Architecture of the RBFN.

in Fig. 9. It has universal approximation, optimization, and regularization capabilities [76]. It has been proved that the RBFN can theoretically approximate any continuous function [72,76].

The RBFN is a type of receptive field network, or a localized network. The localized approximation method provides the strongest output when the input is near a node centroid. It has a faster learning speed as compared to global methods, such as the MLP with the BP rule, and only part of the input space needs to be trained. It has been reported that the RBFN requires orders-of-magnitude less iterations for convergence than the popular MLP with the BP rule using the sigmoid activation function [53]. When a new sample is fed to the network, the network modifies weights only in the vicinity of the sample point, and maintains constant weights in the other regions. The basis function covers only a very small local zone, and the extension of its neighborhood is determined by the variance of units, which usually incurs the requirement of more network resources. These disadvantages can be eliminated or reduced by progressive learning. The RBFN is insensitive to the order of the appearance of the adjusted signals, and hence more suitable for on-line or sequent adaptive adjustment [27]. Another well-known receptive field network is the cerebellar model articulation controller (CMAC) [3].

The nonlinear relation at the nodes of the second layer is a radial-basis function (RBF), which possesses excellent mathematical properties. The connection from the second layer to the third layer is linear. The RBFN can achieve a global optimal solution using the linear optimization method. If the radial-basis function is suitably selected and the node centroids are fixed, the hidden layer performs a fixed nonlinear transform, and the output layer becomes a linear combiner mapping the nonlinearity into a new space. The adjustable weight can be determined in the LMS error sense.

For input space $\mathbf{X}$, the network output

$$\mathbf{y} = \mathbf{\Phi W}, \tag{12}$$

where $\mathbf{W} = [\mathbf{w}_1 \mathbf{w}_2 \ldots \mathbf{w}_K]$, $\mathbf{w}_i = (w_{i1} w_{i2} \ldots w_{iN})^{\mathrm{T}}$, $\mathbf{\Phi} = [\Phi_{mn}]_{M \times N}$, $\Phi_{mn} = \phi(||\mathbf{x}_m - \mathbf{c}_n||)$, M is the number of input samples, and N is the number of basis centers. The basis function $\phi(\cdot)$ is typically selected as Gaussian.

The choice of the vector centroid of the basis function is critical to the approximation accuracy. The centroid can be placed on some or all the training examples, or determined by clustering or randomly selecting untrained samples. Clustering, such as the K-means [91], is an unsupervised learning algorithm. In [16] the authors have proposed to adjust the vector center by continuously adapting the cluster center using a linear learning rule to track the clusters

$$\mathbf{c}_i(k) = \alpha \mathbf{c}_i(k-n) + (1-\alpha)\mathbf{x}(k), \tag{13}$$

where $\mathbf{c}_i(k)$ is the radial-basis function center corresponding to sequence i at time k, $\mathbf{x}(k)$ is the current observant value, $\mathbf{c}_i(k-n)$ is the last value of function center, and α is the adaptation gain between 0 and 1. The network parameter adaptation strategies have been dealt in [37].

In [22,59,60,95] a covariance matrix of measured signals has been used as the input to the RBFN to eliminate the initial phase at the preprocessing stage. The input vector was normalized to reduce the effect of signal gain. The DoA problem has been solved in [59], and the method has been extended in [60,95] to a multipath environment. In [68,69,86,87], each input vector of the RBFN consists of the sines and cosines of the phase differences between the measured signals of the elements and the reference element. This treatment is to deal with network training difficulty associated with 2π periodic basis discontinuity. The combined basis function is also referred to as the periodic basis function (PBF) [66]. The sines and cosines are also to limit the boundary of the input. We can also use the mod(2π) operator to limit the input space and its discontinuity. Two training schemes, adaptive RBF (ARBF) [55] and linear RBF, have been studied. The ARBF adds basis functions as needed and can move the center of the basis functions when they are very

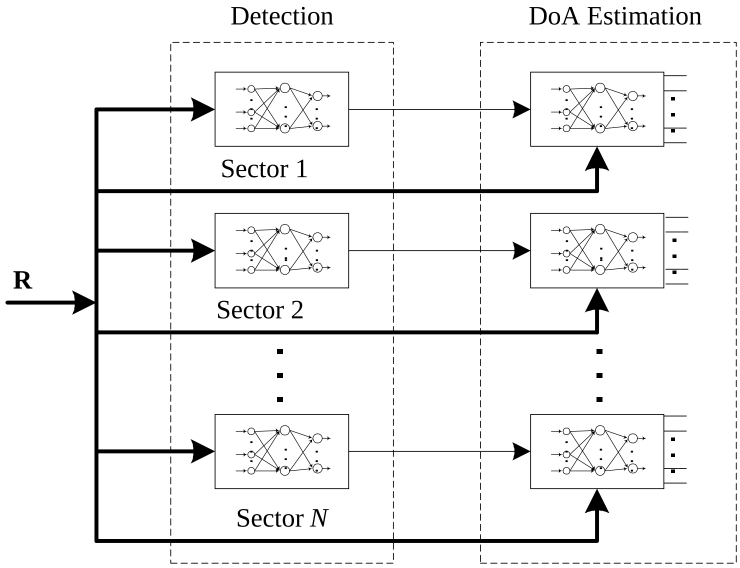

Fig. 10. Structure of the coarse-fine strategy.

close to each other, whereas the linear RBF method places Gaussian RBF's at all training points. The experimental result for single source in case of uniform linear arrays has shown that the neural method is more accurate than the classic monopulse direction finding method even in the case of model inaccuracy [86]. This input mode, however, is only valid for single source DoA estimation. For multisource case, choosing multiple output nodes as classifier for several sectors, and determining the actual DoA by interpolation has been suggested in [86].

An RBFN-based experimental system using a cylindrical eight-element phased array antenna for the DoA problem has been described in [24]. In [34] the RBFN has been used to predict the DoA of the desired signal source, which is then used as the input to the generalized sidelobe canceler (GSC) beamformer. This processing ensures the on-line processing. In [54], the performances of the MLP with the BP rule, the Hopfield network, and the RBFN in the DoA problem have been compared through an example. Simulation results have shown that all these methods perform better than the linear algebra-based techniques, and that the RBFN technique is the fastest. For DoA estimation, it has been demonstrated by simulation that the RBFN outperforms the MUSIC both in accuracy and speed [4,22,24,60,95].

Training a single neural network to detect the DoA of multiple sources requires an exhaustive combination of source angle separation. It is a prohibitive task for more than three sources. Also, the resulting network size is huge, and the performance delay is intolerable. Most of the DoA estimation algorithms also require the knowledge of the number of sources. The coarse-fine strategy proposed in [26] is appealing. The structure is shown in Fig. 10.

The strategy performs both detection and DoA estimation for multiple-source tracking. The field of view is divided into spatial angular sectors, which are in turn assigned to the networks. In the detection stage, a number of RBFNs are trained to detect whether there is signal(s) in a specified sector. Each of these RBFNs has one output node, whose output is binary, serving as a gate to control the second stage computation. Each RBFN at the DoA estimation stage is trained for a specified range of DoA. When a network at the first stage detects the existence of signal(s) in its sector, the corresponding following network is activated to perform DoA estimation. By division of the field of view, each RBFN tracks a smaller number of sources within a smaller angular sector, and thus the size of the training set for each network and the number of distinct locations of possible sources are significantly reduced. Rather than designing a network with the number of output nodes equal to number of sources, one need only specify the minimum angular resolution to be achieved. For a sector of width θ_W and minimum angular resolution of $\Delta\theta_{\min}$, the number of output nodes is given by $J = \theta_W/\Delta\theta_{\min}$. The J output nodes uniformly divide the sector into J bins with the angular separation $\Delta\theta_{\min}$. The output nodes are trained to produce values between 0 and 1. An output of 1 indicates that the source exactly on the bin, and 0 means no source. A value between 0 and 1 represents a source between two bin angles. This method possesses the capability of locating sources that are greater in number than the number of the array elements.

The approximation capability of the RBFN by learning the optimum Wiener beamforming solution and using the trained network to replace Wiener beamformer for real-time tracking has been demonstrated in [20,23,25]. A new basis function, $\phi(r) = 1/(1+r^2/c^2)$, where r represents a kind of distance from the basis center and c is similar to the standard deviation of the Gaussian function, has been applied in [20]. This basis function is suitable for DSP hardware implementation. In [13] the authors have explored the use of the RBFN to learn the beampattern of a circular array, and then perform the RBFN as a beamformer. A comparison with the linear constrained least-squared method (LCLSM [65]) shows that the RBFN can predict reasonably satisfactorily with a good generalization capability.

3.4. Principal component analysis based neural method

The principal component analysis (PCA) is a well-known statistical criterion. This criterion turns out to be closely related to the Hebbian learning rule. The PCA is directly related to the Karhunen–Loeve transform (KLT) and singular value decomposition (SVD). Neural PCA has been introduced by Oja [70], and it allows the removal of the second-order correlation among given random processes. PCA-based neural networks, including symmetric PCA network [70], cross-correlation asymmetric PCA network [18] and Herault–Jutten network [48], can be used for signal processing.

The PCA can be used as a solution to the maximization technique when the weight $\mathbf{W}$ in (3) is the eigenvector corresponding to the largest eigenvalue of the correlation $\mathbf{R}$. On the other hand, the minor component analysis (MCA), which is a variant of the PCA, has been shown to provide an alternative solution to total least-squares (TLS) technique [30]. The solution corresponds to the eigenvector of the smallest eigenvalue of $\mathbf{R}$.

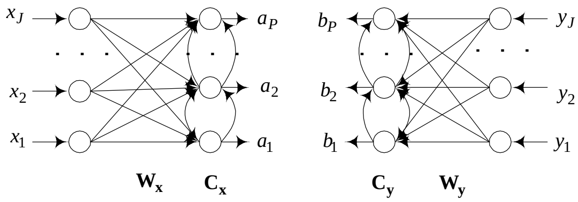

Fig. 11. Cross-correlation ACPA network.

The cross-correlation asymmetric PCA network consists of two sets of neurons laterally connected mutually. The topology of the network is shown in Fig. 11. The two sets of neurons are with cross-coupled Hebbian learning rules orthogonal to each other. This model is extracting the SVD of the cross-correlation matrix of two stochastic signals. The exponential convergence has been demonstrated by simulation.

In Fig. 11, $\mathbf{x}$ and $\mathbf{y}$ are input signals to the network, $\underline{\mathbf{W}} = [\underline{\mathbf{w}}_1, \underline{\mathbf{w}}_2, \ldots, \underline{\mathbf{w}}_P]$ and $\bar{\mathbf{W}} = [\bar{\mathbf{w}}_1, \bar{\mathbf{w}}_2, \ldots, \bar{\mathbf{w}}_P]$ are the weights of network connection, the lateral connection weights $\bar{\mathbf{C}}$ and $\underline{\mathbf{C}}$. The network has the following relation:

$$\mathbf{a} = \underline{\mathbf{W}}^{\mathrm{T}}\mathbf{x}, \tag{14.1}$$

$$\mathbf{b} = \bar{\mathbf{W}}^{\mathrm{T}}\mathbf{y}. \tag{14.2}$$

Thus

$$\mathbf{a}\mathbf{b}^{\mathrm{T}} = \underline{\mathbf{W}}^{\mathrm{T}}\mathbf{x}\mathbf{y}^{\mathrm{T}}\bar{\mathbf{W}}. \tag{15}$$

The learning rules are as follows:

$$\bar{\mathbf{w}}_p(k+1) = \bar{\mathbf{w}}_p(k) + \beta[\mathbf{y}(k) - \bar{\mathbf{w}}_p(k)b_p(k)]a'_p(k), \tag{16.1}$$

$$\underline{\mathbf{w}}_p(k+1) = \underline{\mathbf{w}}_p(k) + \beta[\mathbf{x}(k) - \underline{\mathbf{w}}_p(k)a_p(k)]b'_p(k), \tag{16.2}$$

$$\bar{c}_{pi}(k+1) = \bar{c}_{pi}(k) + \beta[b_i(k) - \bar{c}_{pi}(k)b_p(k)]a'_p(k), \tag{16.3}$$

$$\underline{c}_{pi}(k+1) = \underline{c}_{pi}(k) + \beta[a_i(k) - \underline{c}_{pi}(k)a_p(k)]b'_p(k), \tag{16.4}$$

where

$$a'_p = a_p - \sum_{i<p} \underline{c}_{pi} a_i, \quad a_i = \underline{\mathbf{w}}_i^{\mathrm{T}} \mathbf{x}, \quad i \leqslant p, \tag{17.1}$$

$$b'_p = b_p - \sum_{i<p} \bar{c}_{pi} b_i, \quad b_i = \bar{\mathbf{w}}_i^{\mathrm{T}} \mathbf{y}, \quad i \leqslant p \tag{17.2}$$

and β is the learning rate. In the above algorithm, $\underline{\mathbf{W}}$ and $\bar{\mathbf{W}}$ will approximate the left and right singular vectors of $\mathbf{R}_{\mathrm{yx}}$ respectively, as $k \to \infty$. When $\mathbf{y} = \mathbf{x}$, it reduces to the conventional symmetric PCA [70].

A cross-correlation asymmetric PCA network-based beamformer which makes use of the cyclostationary property of signals to perform blind beamforming has been proposed in [42]. It is an SVD problem of the correlation matrix $\mathbf{R}_{\mathbf{xu}}$, where $\mathbf{x}$ and $\mathbf{u}$ are the array data and its time-frequency translated version. Taking $\mathbf{x}$ and $\mathbf{u}$ as inputs to the network, the network extracts and separates the desired signals simultaneously. In [21], a simple, fast, sub-optimal blind cyclostationary beamforming algorithm, inspired from the cross-correlation asymmetric PCA network [18], has been advanced. It is a gradient decent-based method, and has a computational complexity of $\mathrm{O}(L)$ complex multiplications for each iteration. However, it has the disadvantages of the gradient-decent method, namely, erratic and slow convergence, and the trial-and-error property.

In [57] the authors have combined a symmetrically balanced beamforming array with the Herault–Jutten [48] network for separating independent broadband sound sources and their multipath delays. The proposed method has two advantages: no penalty for very long impulse responses caused by long delays, and no training signals needed for equalization. The proposed VLSI scheme makes use of analog VLSI Herault–Jutten neural chips available. The MCA method has been applied in [29] to beamforming, treating the beamforming weights as the weights of a neuron. The validity of the MCA-based neural method has been demonstrated by simulation.

Most of the AASP methods are based on the subspace concept and require the eigendecomposition of the input correlation matrix. For the purpose of improving the MUSIC algorithm, many efforts have been made to compute the noise space as quickly as possible to meet the real-time requirement. An MCA or PCA algorithm has been proposed in [5] to extract the noise or signal subspace, respectively. By transforming the eigenvector problem in complex form into that in real form, one can perform the extraction of the noise (or signal) subspace using the anti-Hebbian (or Oja) algorithm [71], respectively, which is then followed by one-dimensional minimization using the Newton algorithm. This produces an iterative procedure for real-time DoA estimation and tracking. This algorithm has demonstrated the capability of tracking two time-changing sources by simulation. A four-layered recursive network, based on analog circuit architecture, has been proposed in [61] to compute the noise space. In [9] the authors have described the unitary decomposition artificial neural network (UNIDANN), which is a normalized version of the dynamic equation presented in [61]. The UNIDANN can perform the unitary eigendecomposition of a Hermitian positive definite synaptic weight matrix. The neural output will converge to the principal eigenvectors of the synaptic weight matrix. Due to the introduction of an optimal time-varying weighing in the recursive equation and its underlying analog circuit structure, the UNIDANN exhibits a fast rate of convergence, more accurate final results, and numerical stability compared to the neural network presented in [61].

3.5. *Fuzzy neural method*

Fuzzy logic feedback control is popular in control systems. The inputs to the system are the error and the change in error of the feedback loop, while the output is the control action. The date flow in a fuzzy logic system involves fuzzification, rule base evaluation, and defuzzification [7]. The fuzzy neural network (FNN) is a neural network structure constructed from fuzzy reasoning. The FNN is to acquire fuzzy rules and tune membership functions based on the learning ability of the neural network, and is a synergy of the two paradigms, which captures the merits of both the fuzzy logic and the neural networks. Expert knowledge is expressed by using the fuzzy IF-THEN rules, and is put into the network as a priori knowledge, which can increase the learning speed and estimation accuracy. The advantages of the fuzzy neural method are faster convergence speed with a smaller network size as compared to the ordinary neural networks.

A typical FNN structure includes an input layer, an output layer, and several hidden rule nodes. The FNN employs the same network topologies as the ordinary neural networks, such as the multi-layer perceptron-structured FNN [36], RBFN-structured FNN [1,97], and recurrent FNN [58]. Fundamentals in fuzzy neural synergisms for modeling and control have been reviewed in [80].

When a signal source is not far enough compared to the array aperture, the wavefront is spherical. In [63,64] the authors have attempted to solve the near-field DoA problem using the fuzzy neural method. The fuzzy neural method has been applied in [63] to the near-field real-time DoA tracking with any array configuration. In [64] the authors has presented a fuzzy neural method for the 2-D DoA estimation in a coherent multipath environment, in which the sensor output is preprocessed by a reference-point scheme. In the multipath environment, the proposed 2-D DoA estimator substantially outperforms the spatially smoothing 2-D MUSIC algorithm [101].

In [85] the authors have applied the six-layered self-constructing neural fuzzy inference network (SONFIN) [47] to the DoA problem. The SONFIN is a feedforward multi-layer network that integrates the basic elements and functions of a traditional fuzzy system into a connectionist structure. It can find the proper fuzzy logic rules dynamically, and can find its optimal structure and parameters automatically. It always produces an economical network size, and its learning speed and modeling ability are superior to the ordinary neural networks. The input to the network is the phase difference, which experiences the same input preprocessing as in [86], and thus the method is useful only for single source case. Simulation for a linear array has shown that for the same accuracy in DoA estimation the SONFIN has a much better performance than the RBFN with regard to the convergence rate, requirement on the number of tunable parameters, and insensitivity to noise. In [73], a fuzzy logic-based high-resolution DoA estimation method has been addressed, and the most appealing feature of the proposed method is its low computational burden.

4. Conclusions

We have briefly surveyed the application of the neural method in AASP. Due to its general-purpose nature and proven excellent properties, the neural method provides a powerful means to solve the AASP problems. The neural method outperforms the conventional linear algebra-based method in both the accuracy and speed, although each specific application has its own limitation. It can adjust adaptively with the plant, and can provide a real-time solution with a minimal hardware cost. It is a robust solution for uncertain and perturbed systems. It is especially suitable for hardware implementation. Among the neural models, the Hopfield network is suitable for hardware implementation and can converge to the result in the same order of time as the hardware time constant; the RBFN method is much faster than the MLP method, and is receiving more attention in recent years; the PCA-based neural method deserves attention since it provides a general method for treating the computationally intensive SVD or eigendecomposition, which is common in linear algebra-based methods; the FNN method is a synergy of the neural method and fuzzy logic, which captures the merits of both methods. It can achieve a faster convergence speed with smaller network size, and represents a direction of research interests.

Acknowledgements

The authors are most grateful to the anonymous referees for their relentless efforts to help to improve this paper.

References

[1] D.F. Akhmetov, Y. Dote, S.J. Ovaska, Fuzzy neural network with general parameter adaptation for modeling of nonlinear time-series, IEEE Trans. Neural Networks 12 (1) (2001) 148–152.
[2] M. Alberti, A Hopfield network based adaptation algorithm for phased antenna arrays, Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, Ermioni, Greece, September 1994, pp. 555–564.
[3] J.S. Albus, A new approach to manipulator control: cerebellar model articulation control (CMAC), Trans. ASME, J. Dyn. Systems Meas. Control 97 (1975) 220–227.
[4] J. Alspector, J.D. Norgard, J. Parish, Spatio-temporal equalization for wireless communication, IEEE RAWCON'98 Proceedings, Colorado Springs, CO, USA, August 1998, pp. 165–168.
[5] L. Badidi, L. Radouane, A neural network approach for DoA estimation and tracking, Proceedings of the 10th IEEE Workshop on Statistical Signal and Array Processing, Pocono Manor, PA, USA, August 2000, pp. 434–438.

[6] A. Barron, Complexity regularization with applications to artificial neural networks, in: Roussas (Ed.), Nonparametric Functional Estimation and Related Topics, 1991, pp. 561–576.

[7] B.K. Bose, Fuzzy logic and neural networks in power electronics and drives, IEEE Ind. Appl. Mag. 6 (3) (2000) 57–63.

[8] G.D. Boyd, Optically excited synapse for neural networks, Appl. Opt. 26 (1987) 2712–2719.

[9] S.H. Chang, T.Y. Lee, W.H. Fang, High-resolution bearing estimation via UNItary decomposition artificial neural network (UNIDANN), IEICE Trans. Fund. ELEC. COM. COM. E81A (11) (1998) 2455–2462.

[10] P.R. Chang, W.H. Yang, K.K. Chan, A neural network approach to MVDR beamforming problem, IEEE Trans. Antennas and Propagation 40 (3) (1992) 313–322.

[11] E. Charpentier, J.J. Laurin, Sectorial direction finding antenna array with a MLP beamformer, Proceedings of the IEEE International Symposium on Antennas and Propagation, Vol. 4, Montreal, Canada, July 1997, pp. 2270–2273.

[12] E. Charpentier, J.J. Laurin, An implementation of direction-finding antenna for mobile communications using a neural network, IEEE Trans. Antennas and Propagation 47 (7) (1999) 1152–1159.

[13] D. Chatterjee, S. Chakrabarti, K. Shanmugan, G.E. Prescott, Performance simulation studies for a class of "smart", conformal array antenna architectures, Proceedings of the IEEE International Conference on Phased Array Systems and Techniques, Dana Point, CA, USA, May 2000, pp. 251–254.

[14] Y. Chauvin, D.E. Rumelhart (Eds.), Backpropagation: Theory, Architectures, and Applications, Erlbaum, Hillsdale, NJ, 1995.

[15] L.O. Chua, L. Yang, Cellular neural network: Part I: theory; Part II: applications, IEEE Trans. Circuits Systems 35 (1988) 1257–1290.

[16] C.F.N. Cowan, Non-linear adaptive equalization, Proceedings of the IEE International Conference on Signal Processing, Loughborough, UK, September 1991, pp. 1–5.

[17] G. Cybenko, Approximation by superposition of a sigmoid function, Math. Control, Signals Systems 12 (1989) 303–314.

[18] K.I. Diamantaras, S.-Y. Kung, Cross-correlation neural network models, IEEE Trans. Signal Process. 42 (11) (1994) 3218–3323.

[19] E.D. Di Claudio, R. Parisi, G. Orlandi, Application of the block recursive least squares algorithm to adaptive neural beamforming, Proceedings of the IEEE Workshop on Neural Networks for Signal Processing VII, Amelia Island, FL, USA, September 1997, pp. 560–567.

[20] K.-L. Du, K.K.M. Cheng, M.N.S. Swamy, A fast neural beamformer for antenna arrays, Submitted for presentation at the IEEE International Conference on Communications, New York City, NY, April–May 2002.

[21] K.-L. Du, M.N.S. Swamy, An iterative blind cyclostationary beamforming algorithm, Submitted for presentation at the IEEE International Conference on Communications, New York City, NY, 28 April–2 May, 2002.

[22] A.H. El Zooghby, C.G. Christodoulous, M. Georgiopoulos, Performance of radial-basis function with antenna arrays, IEEE Trans. Antennas and Propagation 45 (11) (1997) 1611–1617.

[23] A.H. El Zooghby, C.G. Christodoulous, M. Georgiopoulos, Neural network-based adaptive beamforming for one- and two-dimensional antenna arrays, IEEE Trans. Antennas and Propagation 46 (12) (1998) 1891–1893.

[24] A.H. El Zooghby, H.L. Southall, C.G. Christodoulous, Experimental validation of a neural network direction finder, Proceedings of the IEEE International Symposium on Antennas and Propagation, Vol. 3, Orlando, FL, USA, July 1999, pp. 1592–1595.

[25] A.H. El Zooghby, C.G. Christodoulou, M. Georgiopoulos, A neural-network-based linearly constrained minimum variance beamformer, Microwave Opt. Technol. Lett. 21 (6) (1999) 451–455.

[26] A.H. El Zooghby, C.G. Christodoulou, M. Georgiopoulos, A neural network-based smart antenna for multiple source tracking, IEEE Trans. Antennas and Propagation 48 (5) (2000) 768–776.

[27] S. Fabri, V. Kadirkamanathan, Dynamic structure neural networks for stable adaptive control of nonlinear systems, IEEE Trans. Neural Networks 7 (5) (1996) 1151–1167.

[28] N.H. Fahat, D. Psaltis, A. Prata, E. Paek, Optical implementation of the Hopfield model, Appl. Opt. 24 (1985) 1469–1475.

[29] S. Fiori, F. Piazza, Neural MCA for robust beamforming, Proceedings of the IEEE International Symposium on Circuits and Systems, Geneva, Switzerland, Vol. 3, May 2000, pp. 614–617.

[30] K. Gao, M.O. Ahmed, M.N. Swamy, A constrained anti-Hebbian learning algorithm for total least-square estimation with applications to adaptive FIR and IIR filtering, IEEE Trans. Circuits Systems II 41 (11) (1994) 718–729.

[31] L.C. Godara, Application of antenna arrays to mobile communications, Part II: beam-forming and direction-of-arrival considerations, Proc. IEEE 85 (8) (1997) 1195–1245.

[32] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[33] D. Goryn, M. Kaveh, Neural networks for narrowband and wideband direction finding, Proceedings of the ICASSP-88, New York, NY, USA, April 1988, pp. 2164–2167.

[34] A.M.G. Guerreiro, A.D.D. Neto, F.A. Lisboa, A joint operation of a neural network and a GSC structure applied to linear and planar Arrays, Proceedings of the SBMO/IEEE International Conference on Microwave and Optoelectronics, Vol. 2, Rio de Janeiro, Brazil, August 1999, pp. 493–498.

[35] A.M. Haimovich, B. Even-Or, Interference cancellation using a Hopfield neural network, Proceedings of the IEEE International Symposium on Circuits and Systems, New Orleans, LA, USA, May 1990, pp. 503–506.

[36] Y. Hayashi, J.J. Buckley, E. Czogala, Fuzzy neural network with fuzzy signals and weights, Int. J. Intell. Systems 8 (1993) 527–537.

[37] S. Haykin, Neural Network: A Comprehensive Foundation, Macmillan, New York, 1994.

[38] Z. He, Y. Chen, Robust blind beamforming using neural network, IEE Proc. Radar, Sonar, Navig. 147 (1) (2000) 41–46.

[39] P.C.J. Hill, P.D. Wells, Antenna beamforming for EW using adaptive layered networks, Proceedings of the IEE Colloquium on Signal Processing in Electronic Warfare, London, UK, January 1994, pp. 2/1–2/5.

[40] M. Hirari, M. Hayakawa, Direction of arrival estimation using blind separation of sources, Radio Sci. 34 (3) (1999) 693–701.

[41] A. Hirose, Dynamics of fully complex-valued neural networks, Electron. Lett. 28 (16) (1992) 1492–1494.

[42] Y.S. Hwu, M.D. Srinath, A neural network approach to design of smart antennas for wireless communication systems, Proceedings of the 31th Asilomar Conference on Signals, Systems & Computers, 1998, pp. 145–148.

[43] K.M. Homi, K.M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366.

[44] J.J. Hopfield, Neurons with graded response have collective computational properties like those of two-state neurons, Proc. Nat. Acad. Sci. 81 (1984) 3088–3092.

[45] J.J. Hopfield, D.W. Tank, Neural computation of decisions in optimization problems, Biol. Cybern. 52 (1985) 141–152.

[46] S. Jha, T. Durrani, Direction of arrival estimation using artificial neural networks, IEEE Trans. Systems Man. Cybernet. 21 (5) (1991) 1192–1201.

[47] C.F. Juang, C.T. Lin, An on-line self-constructing neural fuzzy inference network and its application, IEEE Trans. Fuzzy Systems 6 (1) (1998) 12–32.

[48] C. Jutten, J. Herault, Blind separation of sources, Part I: an adaptive algorithm based on a neuromimetric architecture, Signal Process. 24 (1) (1991) 1–10.

[49] H. Leung, B. Quach, T. Lo, J. Litva, A chaotic neural beamformer for wireless communication, Proceedings of the IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing, Victorial, BC, Canada, May 1995, pp. 538–541.

[50] S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680.

[51] W.G. Knecht, Nonlinear noise filtering and beamforming using the perceptron and its volterra approximation, IEEE Trans. Speech Audio Process. Part 1 2 (1) (1994) 55–62.

[52] T. Kohonen, The self-organizing map, Proc. IEEE 78 (1990) 1464–1480.

[53] B. Kosko (Ed.), Neural Networks for Signal Processing, Prentice-Hall, Englewood Cliffs, NJ, 1992.

[54] C.S. Lee, Nonlinear adaptive techniques for DOA estimation—a comparative analysis, IEEE Proceedings on Electronic Technology Directions to the Year 2000, Adelaide, SA, Australia, May 1995, pp. 72–77.

[55] S. Lee, R.M. Kil, Bidirectional continuous associator based on Gaussian potential function network, Proceedings of the IEEE IJCNN, San Diego, CA, Vol. 1, 1989, pp. 45–53.

[56] Y.C. Liao, Radar tracking with neural network, Proceedings of the IEEE International Workshop on Intelligent Signal Processing and Communication Systems, Taipei, Taiwan, March 1992, pp. 567–575.

[57] S. Li, T.J. Sejnowshi, Adaptive separation of mixed broadband sound sources with delays by a beamforming Herault–Jutten network, IEEE J. Oceanic Eng. 20 (1) (1995) 73–79.

[58] F.J. Lin, R.J. Wai, Hybrid control using recurrent fuzzy neural network for linear-induction motor servo drive, IEEE Trans. Fuzzy Systems 9 (1) (2001) 102–115.

[59] T. Lo, H. Leung, J. Litva, Radial basis function neural network for direction-of-arrivals estimation, IEEE Signal Process. Lett. 1 (2) (1994) 45–47.

[60] T.K.Y. Lo, H. Leung, J. Litva, Artificial neural network for AOA estimation in a multipath environment over the sea, IEEE J. Oceanic Eng. 19 (4) (1994) 555–562.

[61] F.-L. Luo, Y.-D. Li, Real-time neural computation of the noise subspace for the MUSIC algorithm, Proceedings of the IEEE ICASSP, Minneapolis, MN, Vol. 1, 1993, pp. 485–488.

[62] L. Ma, Training simplified neural network for high resolution direction finding of correlated source, Proceedings of the ISSAP, Australia, 1990.

[63] C.W. Ma, C.C. Teng, Near-field direction finding with a fuzzy neural network, Proceedings of the Fuzzy Systems Symposium on Asian Soft Computing in Intelligence Systems and Information Processing, 1996, pp. 230–235.

[64] C.W. Ma, C.C. Teng, Fuzzy neural network method approach for 2-D direction finding in multipath environments, IEE Proc. Radar, Sonar, Navig. 146 (2) (1999) 78–83.

[65] S.R. Nagesh, T.S. Vedavathy, A procedure for synthesizing a specified sidelobe topography using an arbitrary array, IEEE Trans. Antennas and Propagation 43 (7) (1995) 742–745.

[66] F.J. Narcowich, Generalized hermite interpolation and positive definite kernels on a remannian manifold, J. Math. Anal. Comput. 190 (1995) 165–193.

[67] M. Naumovski, R.A. Carrasco, Neural network beamformer for narrowband HF transition, Proceedings of the IEE Colloquium on HF Antennas and Propagation, London, UK, November 1995, pp. 5/1–5/8.

[68] T.H. O'Donnell, M.J. Simmers, H.L. Southall, Lt T. Klemas, Neural beamforming for signal detection and location, Proceedings of the IEEE MILCOM'94, Vol. 2, Fort Monmouth, NJ, USA, October 1994, pp. 326–330.

[69] T.H. O'Donnell, H.L. Southall, J.A. Simmers, T.J. Klemas, Neural network antenna array processing for detection and direction finding, Proceedings of the IEEE International Radar Conference, Vol. 2, Alexandria, VA, USA, May 1995, pp. 213–218.

[70] E. Oja, A simplified neuron model as a principal component analyzer, J. Math. Biol. 15 (1982) 267–273.

[71] E. Oja, Principal components, minor components, and linear neural networks, Neural Networks 5 (1992) 929–935.

[72] J. Park, I.W. Sanberg, Universal approximation using radial basis function networks, Neural Comput. 3 (1991) 246–257.

[73] A. Perez-Neira, M.A. Lagunas, J. Bas, Fuzzy logic for robust detection in wireless communications, Proceedings of the Eighth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications: Waves of the Year 2000 (PIMRC '97), Vol. 3, Helsinki, Finland, September 1997, pp. 1145–1149.

[74] A. Perez-Neira, M.A. Lagunas, Multiuser array beamforming based on a neural network mapping, Proceedings of the ICASSP-94, Vol. 4, Adelaide, SA, Australia, April 1994, pp. 9–12.

[75] F.J. Pineda, Generalization of back propagation to recurrent neural networks, Phys. Rev. Lett. 59 (1987) 2229–2232.

[76] T. Poggio, F. Girosi, Networks for approximation and learning, Proc. IEEE 78 (9) (1990) 1481–1497.

[77] K.D. Rao, Narrowband direction finding using complex EKF trained multilayered neural network, Proceedings of the ICSP'96, Beijing, China, October 1996, pp. 1377–1380.

[78] R. Rastogi, Array signal processing with interconnected neuron-like elements, Proceedings of the IEEE ICASSP'87, Dallas, TX, USA, 1987, pp. 2328–2331.

[79] R. Roy, T. Kailath, ESPRIT—estimation of signal parameters via rotational invariance techniques, IEEE Trans. Acoust. Speech Signal Process. 37 (1989) 984–995.

[80] J.S. Roger Jang, C.I. Sun, Neuro-fuzzy modelling and control, Proc. IEEE 83 (1995) 378–406.

[81] D.W. Ruck, S.K. Rogers, M. Kabrisky, P.S. Maybeck, M.E. Oxley, Comparative analysis of backpropagation and the extended Kalman filter for training multiperceptrons, IEEE Trans. Pattern Anal. Mach. Intell. 14 (6) (1992) 686–691.

[82] M. Scholles, B.J. Hosticka, M. Schwarz, Real-time application of biology-inspired neural networks using an emulator with dedicated communication hardware, Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '95), Vol. 1, Seattle, WA, USA, April–May 1995, pp. 267–270.

[83] R.O. Schmidt, Multiple emitter location and signal parameter estimation, IEEE Trans. Antennas Propagation 34 (1986) 276–280.

[84] S. Shams, Neural network optimization for multi-target multi-sensor passive tracking, Proc. IEEE 84 (10) (1996) 1442–1457.

[85] C.-S. Shieh, C.-T. Lin, Direction of arrival estimation based on phase differences using neural fuzzy network, IEEE Trans. Antennas and Propagation 48 (7) (2000) 1115–1123.

[86] H.L. Southall, J.A. Simmers, T.H. O'Donnel, Direction finding in phased arrays with a neural network beamformer, IEEE Trans. Antennas and Propagation 43 (12) (1995) 1369–1374.

[87] H.L. Southall, Array antenna applications for neural beamforming, Proceedings of the IEEE 30th Asilomar Conference, Vol. 1, Pacific Grove, CA, USA, November 1996, pp. 40–44.

[88] S.L. Speidel, Neural adaptive sensory processing for undersea sonar, IEEE J. Oceanic Eng. 17 (4) (1992) 341–350.

[89] K.-A. Toh, C.-Y. Lee, Efficient network training for DOA estimation, Proceedings of the IEEE International Conference On Systems, Man and Cybernetics, Vol. 5, Tokyo, Japan, October 1999, pp. 383–388.

[90] D. Torrieri, K. Bakhru, Simplification of the MUSIC algorithm using a neural network, Proceedings of the IEEE Military Communication Conference MILCOM, Piscataway, NJ, Vol. 3, 1996, pp. 873–876.

[91] J.T. Tou, R.C. Gonzalez, Pattern Recognition Principles, Addison-Wesley, Reading, MA, 1976.

[92] A.C. Tsoi, A.D. Back, Locally recurrent globally feedforward networks: a critical review of architectures, IEEE Trans. Neural Networks 5 (2) (1994) 229–239.

[93] W.-H. Wang, K.-K. Chan, P.-R. Chang, Complex-valued neural network for direction of arrival estimation, IEEE Electron. Lett. 30 (7) (1994) 574–575.

[94] P.D. Wells, P.C.J. Hill, Adaptive antenna beamforming using artificial neural networks, Proceedings of the IEEE Workshop on Neural Network Applications and Tools, Liverpool, UK, September 1993, pp. 1–5.

[95] T. Wong, T. Luo, H. Leung, J. Litva, E. Bosse, Low-angle radar tracking using radial basis function neural network, IEE Proc.—F 140 (5) (1993) 323–328.

[96] Q. Wu, K.M. Wong, Blind adaptive beamforming for cyclostationary signals, IEEE Trans. Signal Process. 44 (11) (1996) 2757–2767.

[97] S. Wu, M.J. Er, Dynamic fuzzy neural networks—a novel approach to fuzzy-neural system modeling, IEEE Trans. Systems Man. Cybern. Part B 30 (2) (2000) 358–364.

[98] X. Yao, Evolving artificial neural networks, Proc. IEEE 87 (9) (1999) 1423–1447.

[99] J. Yang, Q. Wu, J.P. Reilly, A neural network for direction finding in array processing, Proceedings of the ICASSP-92, Vol. 2, San Francisco, CA, USA, March 1992, pp. 313–316.

[100] W.H. Yang, P.R. Chang, Programmable switched-capacitor neural network for MVDR beamforming, IEEE J. Oceanic Eng. 21 (1) (1996) 77–84.

[101] C.C. Yeh, J.H. Lee, Y.M. Chen, Estimating two-dimensional angles of arrival in coherent source environment, IEEE Trans. Acoust. Speech Signal Process. 37 (1989) 153–155.