

# Fuzzy neural network approach for 2-D direction finding in multipath environments

C.W.Ma and C.C.Teng

**Abstract:** A fuzzy neural network (FNN) approach for estimating the two-dimensional (2-D) direction of a radiating source in coherent multipath environments via a 2-D passive sensor array is presented. Outputs of the array are preprocessed by a judiciously constructed reference-point preprocessing scheme to produce proper inputs for an FNN. The FNN then uses these preprocessed outputs to produce the estimate of the source direction. Since the preprocessed outputs preserve enough information about the source direction and the FNN maps the preprocessed outputs into the source direction with high accuracy, the FNN approach exhibits good estimation performance. By computer simulations, it is found that the FNN approach drastically outperforms the spatial smoothing 2-D MUSIC algorithm.

## 1 Introduction

Finding the direction of a radiating source is an important task in sensor array signal processing. Applications of such a technique are versatile, including radar, sonar, multiuser mobile communication systems, etc. Researchers have published some remarkable works in the past twenty years. Among these works, the MUSIC algorithm [1], ESPRIT algorithm [2], weighted subspace fitting [3], MODE algorithm [4], etc., have received a lot of attention. These algorithms are generally based on an eigen-decomposition of the array's output covariance matrix. As a result, the computational cost is expensive when the array size is large. The high computational cost makes these algorithms difficult to implement for a real-time environment. Recently, some researchers have become interested in using a fuzzy system or neural network, or both, to deal with the direction finding problem. The advantages of using the fuzzy system and neural network include a low computational cost and the ability to deal with complex signal environments. In [5], fuzzy systems are applied to an automatic target detection and tracking sonar system from an application-oriented point of view. In [6], a fuzzy neural network (FNN) based near-field moving target tracking system was successfully developed. In [7], a neural network beamformer was shown to be efficient in direction finding using phased arrays. In [8] and [9], a radial basis function neural network is used for direction finding in a multipath environment over the sea. All these works consider the one-dimensional direction finding problem. Herein, we extend the 'fuzzy-neural' approach to the two-dimensional 2-D direction finding problem.

In this paper, an FNN approach is presented for estimating the 2-D direction of a far field radiating source using a 2-D sensor array in multipath environments. The multipath

environment may be found in multiuser mobile communication systems, underwater sonar systems, and low-angle radar tracking systems, etc. [10–12]. The multipath severely degrades the performance of conventional adaptive array processing. However, the proposed FNN approach performs very well and is superior to the well known spatial smoothing 2-D MUSIC algorithm, which is computationally expensive. The FNN approach preprocesses the array outputs and uses an FNN to construct a mapping that maps from the preprocessed outputs into the 2-D direction estimates. Regarding the pre-processing, it should preserve information about the target direction as much as possible and remove all redundant factors, such as the power and phase of the signal and the power of the noise. Moreover, it should be simple to implement in real-time applications. With these guidelines in mind, we develop a so-called reference-point preprocessing scheme. We first choose some reference points in the space that the target may travel to. Then we calculate some judiciously defined 'distances' between array output covariance matrices of the reference points and the array output covariance matrix of the target. Since information about the target direction is hidden in the array output covariance matrix, it is also hidden in the distances. Consequently, there is a mapping from these distances to the target direction. The FNN is then used to construct such a mapping. Note that theoretically, many kinds of mapping mechanisms can be used for this mapping task. We use the FNN because it is a rule based network. The rules can be combined, eliminated, or added to reduce the size of the network, to increase the mapping accuracy, or to implement an expert's experience [13–15]. Simulation results show that the FNN maps these distances to the target direction with high accuracy. Simulations also show that the FNN approach drastically outperforms the well known spatial smoothing 2-D MUSIC algorithm [16].

## 2 Problem formulation

Consider a far-field narrowband stationary radiating source observed by a 2-D planar array in a multipath environment, as shown in Fig. 1. In order to simplify the complexity of

© IEE, 1999

*IEE Proceedings* online no. 19990258

DOI: 10.1049/ip-rsn:19990258

Paper first received 26th January and in revised form 23rd October 1998

The authors are with Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu, Taiwan, Republic of China

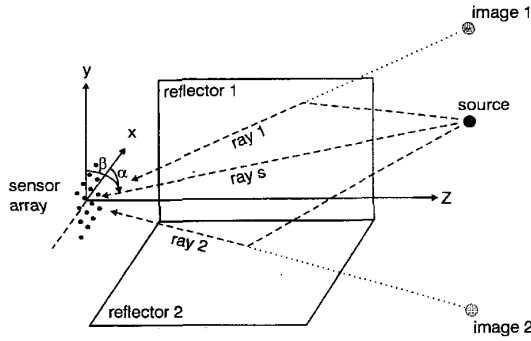


Fig. 1 Multipath environment

the problem without loss of generality, we consider two reflectors. The radiated signal travels to the array along three paths. One is the direct path, which is denoted as rays. The others are reflected paths, which are denoted as ray<sup>1</sup> and ray<sup>2</sup>. The signal that comes from ray<sup>1</sup> can be treated as having been radiated by image<sup>1</sup>. Similarly, the signal that comes from ray<sup>2</sup> can be treated as having been radiated by image<sup>2</sup>.

The data observed at the array can be expressed as

$$\mathbf{x}(t) = \mathbf{A} \cdot s(t) + \mathbf{A}_1 \cdot s(t) \cdot \rho_1 + \mathbf{A}_2 \cdot s(t) \cdot \rho_2 + \mathbf{n}(t) \quad (1)$$

where  $\mathbf{x}(t) \in C^{P \times 1}$  is the data vector,  $\mathbf{n}(t) \in C^{P \times 1}$  is the noise vector,  $\mathbf{A} \in C^{P \times 1}$ ,  $\mathbf{A}_1 \in C^{P \times 1}$  and  $\mathbf{A}_2 \in C^{P \times 1}$  are the array steering vectors at the source direction, the direction of image<sup>1</sup>, and the direction of image<sup>2</sup>, respectively. Moreover,  $s(t)$  is a complex scalar denoting the signal radiated by the source, and  $\rho_1$  and  $\rho_2$  are complex scalars representing the reflection coefficients of reflector<sup>1</sup> and reflector<sup>2</sup>, respectively. So,  $\mathbf{A} \cdot s(t)$  is the array response due to the direct path,  $\mathbf{A}_1 \cdot s(t) \cdot \rho_1$  is the array response due to the reflected path ray<sup>1</sup>, and  $\mathbf{A}_2 \cdot s(t) \cdot \rho_2$  is the array response due to the reflected path ray<sup>2</sup>. The array steering vectors  $\mathbf{A}$ ,  $\mathbf{A}_1$ , and  $\mathbf{A}_2$  can be expressed as vectors whose  $i$ th elements are

$$a_i = \exp \left\{ j \frac{2\pi}{\lambda} (x_i \xi_s + y_i \zeta_s) \right\} \quad (2)$$

$$a_{1i} = \exp \left\{ j \frac{2\pi}{\lambda} (x_i \xi_1 + y_i \zeta_1) \right\} \quad (3)$$

and

$$a_{2i} = \exp \left\{ j \frac{2\pi}{\lambda} (x_i \xi_2 + y_i \zeta_2) \right\} \quad (4)$$

respectively, where  $\lambda$  is the wavelength,  $(x_i, y_i)$  is the coordinate of the  $i$ th sensor, and  $(\xi_s, \zeta_s) = (\cos \alpha, \cos \beta)$ .  $\alpha$  and  $\beta$  are the angles against the  $x$ - and  $y$ -axes, respectively (see Fig. 1).  $(\xi_1, \zeta_1)$ ,  $(\xi_2, \zeta_2)$  are defined as  $(\xi_s, \zeta_s)$  in the same way to represent the directions of image<sup>1</sup> and image<sup>2</sup>.

It is assumed that  $s(t)$  and  $\mathbf{n}(t)$  are uncorrelated and that they are stationary, ergodic and complex-valued random processes with zero mean. Let  $\mu^2$  denote the covariance (or power) of  $s(t)$ . Let the covariance matrix of  $\mathbf{n}(t)$  be  $\sigma^2 \mathbf{I}$ ,

where  $\sigma^2$  is the noise power at each sensor, and  $\mathbf{I}$  is an identity matrix. Thus we have

$$\begin{aligned} \mathbf{R} &\triangleq E[\mathbf{x}(t)\mathbf{x}(t)^H] \\ &= \mu^2 \{ \mathbf{A}\mathbf{A}^H + \|\rho_1\|^2 \mathbf{A}_1\mathbf{A}_1^H + \|\rho_2\|^2 \mathbf{A}_2\mathbf{A}_2^H + 2\text{Re}[\mathbf{A}\rho_1^H \mathbf{A}_1^H] \\ &\quad + 2\text{Re}[\mathbf{A}\rho_2^H \mathbf{A}_2^H] + 2\text{Re}[\mathbf{A}_1\rho_1\rho_2^H \mathbf{A}_2^H] \} + \sigma^2 \mathbf{I} \end{aligned} \quad (5)$$

where  $E[y]$  is the mean of  $y$ , and  $(\cdot)^H$  denotes a conjugate transpose. Instead of being given the exact ensemble covariance matrix  $\mathbf{R}$ , we are given a series of samples from  $\mathbf{x}(t)$ , say  $\{\mathbf{x}(t_1), \mathbf{x}(t_2), \dots, \mathbf{x}(t_N)\}$ . The 2-D direction finding problem is to estimate  $(\xi_s, \zeta_s)$  using the samples.

### 3 Fuzzy neural networks

The 2-D direction finding problem can be solved by the well known spatial smoothing 2-D MUSIC algorithm [16]. In the multipath environment, as described in Section 2, the MUSIC algorithm has to estimate three 2-D directions, and choose one of them as the estimate of the source direction. Actually, our purpose is to estimate only one 2-D direction. The signal radiated from image<sup>1</sup> and image<sup>2</sup> is helpful for estimating the source direction. In other words, when estimating the source direction, the reflected signal can be taken into account to improve the estimation performance. One way to implement this idea is by using the fuzzy neural networks to map the array output to the source direction. In this Section we describe a 4-layer fuzzy neural network, which will be adopted to estimate the  $(\xi_s, \zeta_s)$  in the next Section. The fuzzy neural network has two outputs. One of the outputs is for estimating  $\xi_s$ , the other is for estimating  $\zeta_s$ .

#### 3.1 Structure of the FNN

As shown in Fig. 2, the fuzzy neural network adopted in this paper is an  $n$ -input, 2-output, and  $m$ -rule fuzzy neural network that maps  $\{u_i\}_{i=1}^n$  into  $y_1$  and  $y_2$ . It constructs fuzzy rules one by one and adds all rules together. Suppose that the  $j$ th fuzzy rule reads

$$\begin{aligned} &\text{IF } u_1 \text{ IS } A_{1,j}, u_2 \text{ IS } A_{2,j}, \dots, u_n \text{ IS } A_{n,j}, \text{ THEN} \\ &y_1 \text{ IS } \omega_{1j} \text{ and } y_2 \text{ IS } \omega_{2j} \end{aligned}$$

Fig. 3 shows the implementation of this rule. Putting all of the fuzzy rules together, we get, the whole fuzzy neural

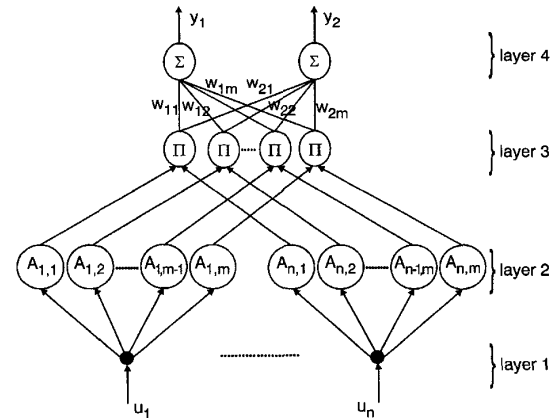


Fig. 2 Structure of a fuzzy neural network

$\Sigma$  = weighted sum  
 $\Pi$  = product operation  
 $A_{i,j}$  = fuzzy membership function (Gaussian)

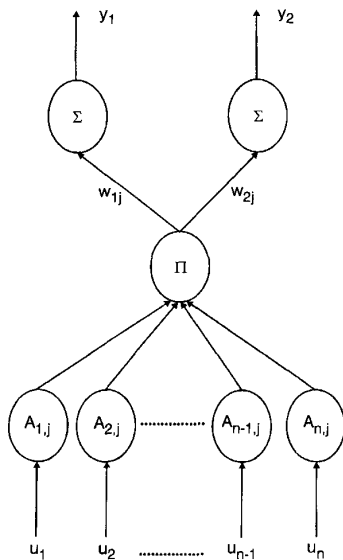


Fig. 3 Implementation of a fuzzy rule

network as shown in Fig. 2. Layer 1 of such a network is the input layer. It propagates the crisp input  $u_i$  to layer 2. Layer 2 is the singleton fuzzification layer, which maps a crisp input value  $u_i$  into the fuzzy set  $A_{ij}$  with membership degree  $\mu_{A_{ij}}(u_i)$ . In this paper,  $\mu_{A_{ij}}(\cdot)$  is a Gaussian function, i.e.

$$\mu_{A_{ij}}(u_i) = \exp\left[-\frac{u_i - a_{i,j}}{2 \cdot (\sigma_{i,j})^2}\right] \quad (7)$$

where  $a_{i,j}$  is the mean (or centre) and  $(\sigma_{i,j})^2$  is the variance (or width) of the Gaussian function. The membership degree  $\mu_{A_{i,j}}(u_i)$  is then propagated to layer 3. Layer 3, the fuzzy reasoning layer, performs IF-condition reasoning by a product operation and generates the firing strength  $\alpha_j$  of the  $j$ th fuzzy rule by

$$\alpha_j = \prod_{i=1}^n \mu_{A_{ij}}(u_i) \quad (8)$$

After the firing strength,  $\alpha_j$ , of each rule is computed, the network multiplies  $\alpha_j$  by the consequence weight  $\omega_{1j}$  (or  $\omega_{2j}$ ) to obtain the contribution of the  $j$ th rule to the output  $y_1$  (or  $y_2$ ). Finally, at layer 4, all the contributions of the  $m$  fuzzy rules are summed to produce the output of the FNN, say

$$y_k = \sum_{j=1}^m \omega_{kj} \alpha_j, \quad k = 1, 2 \quad (9)$$

We refer to the defuzzification process in eqn. 9 as a 'weighted sum defuzzifier'. It is a universal approximator which is capable of approximating any real continuous function with satisfactory accuracy, provided that sufficient fuzzy rules are used [14]. Note that the FNN proposed herein and the FNN proposed in [13] are different, in that one uses a weighted sum defuzzifier and the other uses an averaged sum defuzzifier.

### 3.2 Designing an FNN

Although the FNN is considered as a universal approximator to any real continuous function, the determination of the membership function  $\mu_{A_{ij}}(\cdot)$  and consequence weight  $\omega_j$  in eqns. 8 and 9 plays a crucial role when

designing the fuzzy neural network. The advantage of the FNN is that we are able to adjust all the  $\omega_{1j}$ ,  $\omega_{2j}$ , and  $\mu_{A_{ij}}(\cdot)$  using the back propagation algorithm, provided that a lot of input-output training pairs are available. The back-propagation algorithm performs the supervised gradient descent learning. Interested readers may find the details of the learning processes in [6, 15]. In the following we summarise the design procedure.

Step 1: Computer a lot of input-output pairs of the desired mapping.

Step 2: Construct an initial FNN according to expert knowledge or any other initialisation procedure, e.g., the on-line initialisation procedure proposed in [13].

Step 3: Train the FNN (i.e., adjust  $\omega_{1j}$ ,  $\omega_{2j}$  and  $\mu_{A_{ij}}(\cdot)$ ) by using the back propagation algorithm to make the FNN fit the input-output pairs obtained in step 1.

Step 4: Eliminate redundant rules, which consist of fuzzy sets which look like impulsive functions.

## 4 FNN direction estimator using reference-point preprocessing scheme

The universal function approximation capability of an FNN is suitable for creating an estimator for the two-dimensional direction of a radiating source. As shown in Fig. 4, the FNN-estimator includes two main parts: the preprocessing scheme and the FNN mapping network. It also includes simple postprocessing units. The postprocessing merely scales and shifts the outputs  $y_1$  and  $y_2$ . According to our experience, this will make the FNN have a faster learning speed. Regarding the preprocessing scheme, we keep in mind the following two guidelines:

- (i) At the array output, the power of the source signal  $s(t)$  and the power of the noise are unrelated to the direction, which should be removed before being fed into an FNN.
- (ii) We should use the array output covariance matrix instead of the array outputs to make the FNN-estimator robust to noise.

Based on these two guidelines, we develop a so-called reference-point preprocessing scheme.

By the term 'reference point', we mean some points we choose in the space of  $(\xi, \zeta)$ , which will be used for reference when producing the inputs of the FNN. Let  $(\xi^i, \zeta^i)_{i=1}^n$  denote the  $n$  reference points. Obviously, there are distances between reference points  $(\xi^i, \zeta^i)_{i=1}^n$  and the true source direction  $(\xi_s, \zeta_s)$ . If the distances can be estimated,  $(\xi_s, \zeta_s)$  can be estimated too. This is something like the positioning technique used in the Global Positioning System (GPS). Herein, we attempt to use the same concept

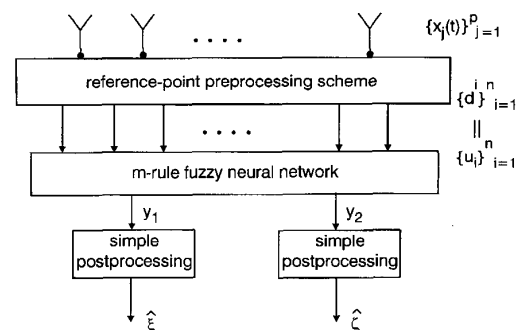


Fig. 4 Diagram of an FNN-based 2-D direction estimator

to ‘position’ the target direction. Although the distances are not known, we may define another kind of distance for our purpose. The reference-point preprocessing scheme defines and computes some novel distances, which can be used to position the source direction  $(\hat{\xi}_s, \hat{\zeta}_s)$ . Considering guideline 2, the novel distances should come from the array sample covariance matrix instead of from the array outputs. In the following, we give the definition of the novel distance and show how to calculate them.

Let  $\mathbf{x}^i$  be the array output when the source is located at the  $i$ th reference point  $(\xi^i, \zeta^i)$ . Assume that  $\mu = 1$  and  $\sigma = 0$ . The covariance matrix of  $\mathbf{x}^i$ ,  $\mathbf{R}^i$ , becomes

$$\mathbf{R}^i = \mathbf{A}^i \mathbf{A}^{iH} + \|\rho_1\|^2 \mathbf{A}_1^i \mathbf{A}_1^{iH} + \|\rho_2\|^2 \mathbf{A}_2^i \mathbf{A}_2^{iH} + 2\text{Re}[\mathbf{A}^i \rho_1^H \mathbf{A}_1^{iH}] + 2\text{Re}[\mathbf{A}^i \rho_2^H \mathbf{A}_2^{iH}] + 2\text{Re}[\mathbf{A}_1^i \rho_1^H \rho_2^H \mathbf{A}_2^{iH}] \quad (10)$$

where  $\mathbf{A}^i$ ,  $\mathbf{A}_1^i$ , and  $\mathbf{A}_2^i$  are the steering vectors of the direct path and reflected path with respect to the  $i$ th reference point. Eqn. 10 shows the covariance matrices associated with the reference points. The covariance matrix associated with the source direction, however, should be calculated from the array outputs. Let  $\hat{\mathbf{R}}$  be the array sample covariance matrix when the source direction is  $(\hat{\xi}_s, \hat{\zeta}_s)$ , which is calculated as

$$\hat{\mathbf{R}} = \frac{1}{N} \sum_{t=1}^N \mathbf{x}(t) \mathbf{x}(t)^H \quad (11)$$

We define vector  $\mathbf{r}$  as follows:

$$\mathbf{r}' = [r_{12}, r_{13}, \dots, r_{1p}, r_{23}, r_{24}, \dots, r_{2p}, \dots, r_{(p-1)p}] \quad (12)$$

$$\mathbf{r} = \mathbf{r}' \times \frac{1}{\|\mathbf{r}'\|} \quad (13)$$

where  $r_{ij}$  is the element of  $\hat{\mathbf{R}}$  at the  $(i, j)'$  is the 2-norm of  $\mathbf{r}'$ . Since eqn. 12 does not include the diagonal of  $\hat{\mathbf{R}}$ ,  $\mathbf{r}$  has no strong relationship to the noise power  $\sigma^2$ .  $\mathbf{r}$  is almost uncorrelated to the signal power  $\mu^2$ , since it appears in both the numerator and denominator of eqn. 13. Therefore,  $\mathbf{r}$  satisfies the first guideline.

We now define vector  $\{\mathbf{r}^i\}_{i=1}^n$  in the same manner as we defined  $\mathbf{r}$ , and calculate the distances between  $\{\mathbf{r}^i\}_{i=1}^n$  and  $\mathbf{r}$ . The novel distances are then designated as

$$d^i = \|\mathbf{r} - \mathbf{r}^i\| \text{ for } i = 1, 2, \dots, n \quad (14)$$

We find that there is a mapping between  $\{d^i\}_{i=1}^n$  and  $(\hat{\xi}_s, \hat{\zeta}_s)$ . This means that we can use  $\{d^i\}_{i=1}^n$  to ‘position’  $(\hat{\xi}_s, \hat{\zeta}_s)$  through a mapping. The FNN can establish such a mapping. Using  $\{d^i\}_{i=1}^n$  as the inputs of the FNN, the FNN’s outputs can be scaled and shifted to provide the direction estimates  $(\hat{\xi}_s, \hat{\zeta}_s)$ , i.e.

$$u_i = d^i \text{ for } i = 1, 2, \dots, n \quad (15)$$

where  $u_i$  is the  $i$ th input of the FNN, and

$$y_1 = \text{shifted and scaled version of } \hat{\xi}_s \quad (16)$$

$$y_2 = \text{shifted and scaled version of } \hat{\zeta}_s \quad (17)$$

where  $y_1$  and  $y_2$  are the outputs of the FNN. The simple scaling and shifting give the FNN a faster learning speed. Note that  $\{d^i\}_{i=1}^n$  satisfies both the guidelines mentioned before and the FNN provides good mapping capability. As a result, the FNN approach is expected to estimate the 2-D direction with high accuracy.

## 5 Simulations

In this Section, the estimation performance of the proposed FNN-based 2-D direction estimator is compared with the well known spatial smoothing 2-D MUSIC algorithm. In all simulations, the array is chosen to be a uniform rectangular planar array with  $4 \times 4$  sensors and half-wavelength interelement spacing in both row and column directions. As shown in Fig. 1 in Section 2, the centre of the array is located at the origin of the coordinate system, and the broadside of the array faces the  $z$ -axis. One unit length of the coordinate system is one half-wavelength. The reflector 1 is implemented by the plane  $x = 100$ . The reflector 2 is implemented by the plane  $y = -100$ . The radiating source is 500 unit lengths away from the origin and is therefore considered to be in the far-field. Moreover,  $\xi < 0.1$  and  $\zeta > -0.1$ . Herein we are going to develop an FNN-based estimator to estimate the source direction which may vary in the following region:

$$-0.1 < \xi < 0.1 \quad (18)$$

$$-0.1 < \zeta < 0.1 \quad (19)$$

We start by choosing some reference points, and calculating  $\{d^i\}_{i=1}^n$  using the equations mentioned in previous Section. The  $\{d^i\}_{i=1}^n$  will be the inputs of an FNN. The FNN’s outputs are shifted and scaled to produce the estimate of  $(\hat{\xi}_s, \hat{\zeta}_s)$ . By changing the target direction  $(\xi_s, \zeta_s)$ , we get a lot of training data for the FNN, i.e., the pair of  $\{d^i\}_{i=1}^n$  and  $(\xi_s, \zeta_s)$ . Using the training data and the training method presented in Section 3, we can develop an FNN-based direction estimator. In this example, we find that only three (i.e.  $n=3$ ) reference points are enough to build an accurate FNN direction estimator in a noise free environment. Since a large number of reference points increases the complexity of the FNN, we choose a minimum, but sufficient, number of reference points. Note that the sufficient number of reference points may be greater than three in other cases. Now, 25 source directions are generated randomly in order to examine the accuracy of the FNN-based direction estimator. For each direction, 100 array outputs are used to calculate the  $\hat{\mathbf{R}}$ . Using the  $\hat{\mathbf{R}}$ , we are able to calculate  $\{d^i\}_{i=1}^3$ . These  $d^i$  are then fed into the

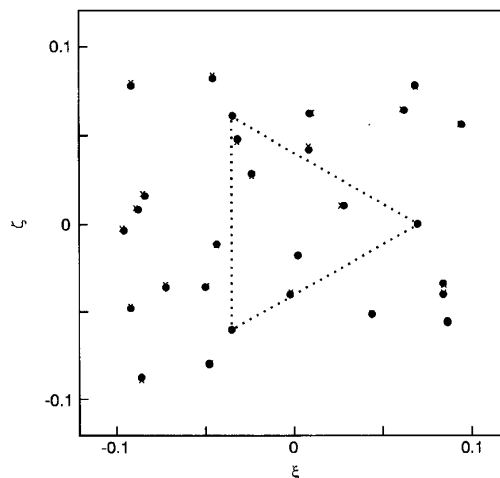
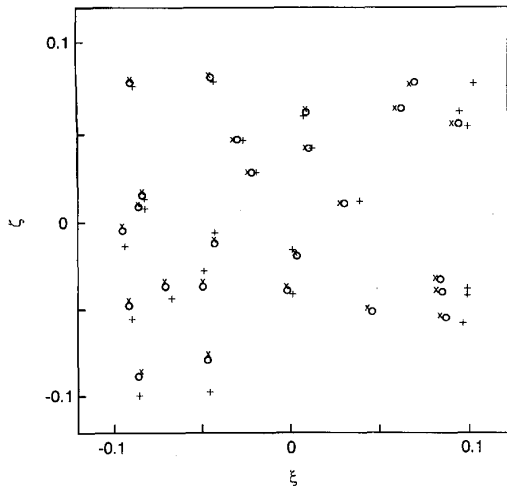


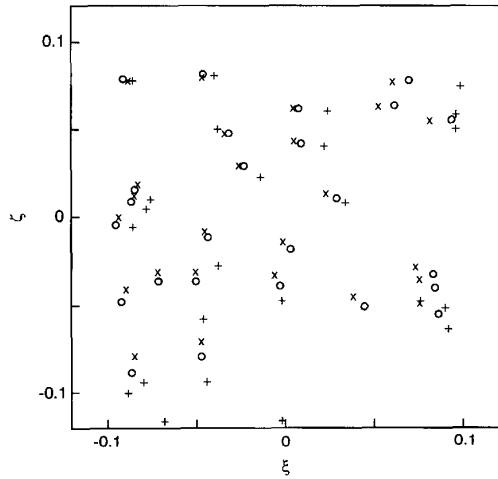
Fig. 5 Results in noise free environment

- \* reference point
- O true source direction
- x FNN estimates
- + MUSIC estimates



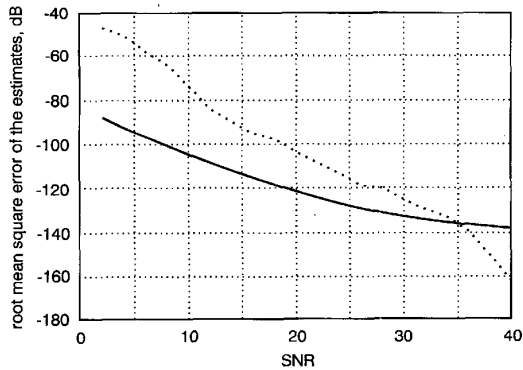
**Fig. 6** Results when SNR = 20dB

○ true source direction  
 × FNN estimates  
 + MUSIC estimates



**Fig. 7** Results when SNR = 10dB

○ true source direction  
 × FNN estimates  
 + MUSIC estimates



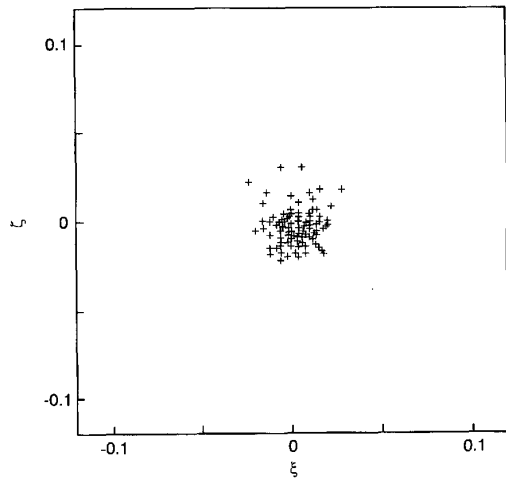
**Fig. 8** Root mean squares error against SNR

— FNN approach  
 --- MUSIC method  
 True source direction at (0,0)

FNN to produce the estimates of the source direction. Meanwhile, the source direction is also estimated by the spatial smoothing 2-D MUSIC algorithm for comparison. The results are plotted in Fig. 5. We observe that the estimation error of the FNN-based estimator is satisfactorily small. This accuracy is a result of the universal function approximation capability of the FNNs. We also observe that the spatial smoothing 2-D MUSIC algorithm provides error-free estimates in the noise-free environment.

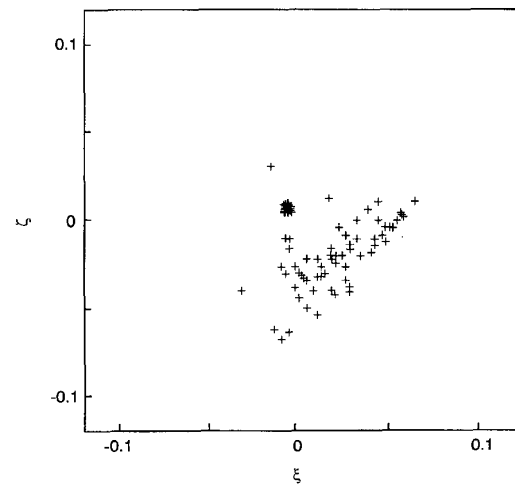
We proceed to examine the estimation performance of these two algorithms in a noisy environment. Figs. 6 and 7 show the results when SNR = 20 dB, and 10 dB, respectively. Clearly, the FNN is more robust to noise than the MUSIC algorithm.

To further investigate these two algorithms, we set the source direction at (0, 0) and make 100 independent Monte Carlo trials for various SNR values. The root mean square errors of these two algorithms are plotted in Fig. 8 as a function of SNR. We see that the FNN approach drastically outperforms the MUSIC algorithm for most SNR values.



**Fig. 9** Results when SNR = 12dB

True source direction at (0,0)  
 × FNN estimates  
 + MUSIC estimates



**Fig. 10** Results when SNR = 4dB

True source direction at (0,0)  
 × FNN estimates  
 + MUSIC estimates

When SNR is higher than 34dB, the MUSIC algorithm performs better. This is a result of the limited approximation capability of the FNN. The limited approximation capability, however, is accurate enough. Figs. 9 and 10, are the results of all these 100 estimates when SNR = 12dB and 4dB, respectively. These figures tell us more about the performance improvement of the FNN approach over the spatial smoothing MUSIC approach.

## 6 Conclusions

The two dimensional direction finding problem in a multipath environment is studied and solved by a fuzzy neural network with an efficient reference-point preprocessing scheme. By computer simulations, we find that a small number of reference points is enough. The distances, calculated based on these reference points, possess enough information for estimating the direction of the radiating source. In the multipath environment, the FNN-based 2-D direction estimator substantially outperforms the spatial smoothing 2-D MUSIC algorithm, which is a high-resolution direction estimator. This may result from the poor decorrelation capability of the spatial smoothing scheme for close spacing signals. It may also result from the fact that the FNN-based estimator estimates only one direction while the MUSIC based algorithm has to estimate three directions using the same information. To sum up, for some complicated array signal processing problems, the artificial intelligence approach may be very helpful. In this paper, we have set up an example for this problem. Preprocessing is important when designing an artificial intelligence approach. Proper preprocessing reduces the complexity that the artificial intelligence mechanism should cope with, and preserves useful information as much as possible. The reference-point preprocessing

scheme proposed herein does this job very well for the 2-D direction finding problem.

## 7 References

- 1 SCHMIDT, R.O.: 'Multiple emitter location and signal parameter estimation', *IEEE Antennas Propag. Mag.*, 1986, **34**, pp. 276-380
- 2 ROY, R., and KAILATH, T.: 'ESPRIT - Estimation of signal parameter via rotational invariance', *IEEE Trans. Acoust. Speech Signal Process.*, 1989, **37**, pp. 984-995
- 3 VIBERG, M., OTTERSTEN, B., and KAILATH, T.: 'Sensor array processing based on subspace fitting', *IEEE Trans. Signal Process.*, 1991, **39**, pp. 2436-2448
- 4 STOICA, P., SHARMAN, K.C.: 'Novel eigenanalysis method for direction estimation', *IEE Proc., Radar Signal Process.*, 1990, **137**, pp 19-26
- 5 KUMMERT, A.: 'Fuzzy technology implemented in sonar systems', *IEEE J. Ocean. Eng.*, 1993, **18**, pp. 483-490
- 6 MA, C.W., and TENG, C.C.: 'Tracking a near-field moving target using fuzzy neural networks', *Fuzzy Sets Syst.* (to be published)
- 7 SOUTHALL, H.L., SIMMERS, J.A., and O'DONNELL, T.H.: 'Direction finding in phased arrays with a neural network beamformer', *IEEE Trans. Antennas Propag.*, 1995, **43**, pp. 1369-1374
- 8 WONG, T., LO, T., LEUNG, H., LITVA, J., and BOSSE, E.: 'Low angle radar tracking using radial basis function neural network', *IEE Proc., Radar Signal Process.*, 1993, **140**, pp. 323-328
- 9 LO, T.K.Y., LEUNG, H., and LITVA, J.: 'Artificial neural network for AOA estimation in a multipath environment over the sea', *IEEE J. Ocean. Eng.*, 1994, **19**, pp. 555-562
- 10 LEUNG, H., LO, T., and LITVA, J.: 'Angle-of-arrival estimation in multipath environment using chaos theory', *Signal Process.*, 1993, **140**, pp. 57-68
- 11 ZOLTOWSKI, M., and HABER, F.: 'A vector space approach to direction finding in a coherent multipath environment', *IEEE Trans. Antennas Propag.*, 1986, **AP-34**, pp. 1069-1079
- 12 YUAN, Y.X., and SALT, J.E.: 'Range and depth estimation using a vertical array in a correlated multipath environment', *IEEE J. Ocean. Eng.*, 1993, **18**, pp. 500-507
- 13 CHAO, C.T., CHEN, Y.J., and TENG, C.C.: 'Simplification of fuzzy-neural systems using similarity analysis', *IEEE Trans. Syst. Man Cybern.*, 1996, **26**, pp. 344-354
- 14 WANG, L.X.: 'Adaptive fuzzy systems and control: design and stability analysis' (Prentice Hall, Englewood Cliff, NJ, 1994)
- 15 CHEN, Y.C., and TENG, C.C.: 'A model reference control structure using a fuzzy neural network', *Fuzzy Sets Syst.*, 1995, **73**, pp. 291-312
- 16 YEH, C.C., LEE, J.H., and CHEN, Y.M.: 'Estimating two-dimensional angles of arrival in coherent source environment', *IEEE Trans. Acoust. Speech Signal Process.* 1989, **37**, pp. 153-155