29 June – 1 July 2006, Sofia, Bulgaria

Neural Network – Based DOA Estimation and Beamfoming for Smart Antenna

Maja Sarevska, Bratislav Milovanović, and Zoran Stanković

Abstract – **This paper considers the *Neural Network* (NN) based smart antenna solution for both *Direction Of Arrival* (DOA) estimation and null-steering beamforming, providing a complete NN solution. The main purpose is to present the influence of the imprecise DOA estimations on the NN beamforming performances. Computer simulations for given example will show that the uncertainty in DOAs in the range less than ±0.5° will provide satisfactory NN based beamforming.**

Keywords – **neural network, DOAs, beamforming.**

I. Introduction

A NN [1,2,3] is a powerful tool in signal processing. Due to its strong numerical approximation capability, it is widely used in identification and optimization. The research in antenna arrays is most active due to its military and commercial applications, and also in new solutions for future radiotelescopes. The focus of antenna array signal processing is on DOA estimation and beamforming.

Currently, superresolution algorithms, such as the *Multiple Signal Classification* (MUSIC) and *Estimation of Signal Parameters via Rotational Invariance Technique* (ESPRIT) [4], can be used to perform the direction finding or angle of arrival estimation. One drawback of these algorithms is the difficulty of implementing them in real time because of their intensive computational complexity. NNs, on the other hand, due to their high-speed computational capability, can obtain results in real time. Once the DOAs of sources are available, the beamforming algorithm can be used to track, in real time, sources of interest, and null out the other sources as interference. This is done by controlling the beampattern of an antenna array in an adaptive means. Conventional methods are typically linear algebra-based methods. They require time-consuming matrix inversion computation and cannot meat real-time requirements. Conventional beamformers require highly calibrated antennas with identical element properties. Performance degradation often occurs due to the fact that these algorithms poorly adapt to element failure or other sources of errors. On the other hand, NN-based antenna array do not suffer from these shortcoming. They use simple addition, multiplication, division, and threshold operations in the basic processing element. They possess advantages as massive parallelism, nonlinear property, adaptive learning capability, generalization capability, strong fault-tolerant capability and insensitivity to uncertainty.

Maja Sarevska, Bratislav Milovanović, and Zoran Stanković are with Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18 000 Niš, Serbia and Montenegro, E-mail: maja_sarevska@yahoo.com, [bata,zoran]@elfak.ni.ac.yu

The paper is organized as follows: Section II describes the DOA estimation problem, Section III presents the NN Null-Steering algorithm for beamforming, Section IV presents the NN DOA uncertainty problem, Section V is presenting the results gained from computer simulations, and in Section VI some conclusion remarks are noted.

II. NN DOA Estimation

Let observe a linear antenna array with M elements, let K (K<M) be the number of narrowband plane waves, centered at frequency ω_0 impinging on the array from directions $\{\theta_1\ \theta_2\ ...\ \theta_K\}$. Using complex signal representation, the received signal in the ith array element is:

$$x_i = \sum_{m=1}^{K} s_m(t)e^{-j(i-1)K_m} + n_i(t), \quad i = 1,2,...M \qquad (1)$$

where $s_m(t)$ is the signal of the m-th wave, $n_i(t)$ is the noise signal received at the i-th sensor and

$$K_m = \frac{\omega_0 d}{c}\sin(\theta_m) \qquad (2)$$

where d is the spacing between the elements of the array, and c is the speed of the light in free-space. In vector notation the output of the array is:

$$\boldsymbol{X}(t)=\boldsymbol{A}\boldsymbol{S}(t)+\boldsymbol{N}(t) \qquad (3)$$

where $\boldsymbol{X}(t)$, $\boldsymbol{N}(t)$, and $\boldsymbol{S}(t)$ are:

$$\begin{aligned}\boldsymbol{X}(t)&=[\ x_1(t)\ x_2(t)\ ...\ x_M(t)]^T\\ \boldsymbol{N}(t)&=[\ n_1(t)\ n_2(t)\ ...\ n_M(t)]^T\\ \boldsymbol{S}(t)&=[\ s_1(t)\ s_2(t)\ ...\ s_K(t)]^T\end{aligned} \qquad (4)$$

In (3) $\boldsymbol{A}$ is the M × K steering matrix of the array toward the direction of the incoming signals:

$$\boldsymbol{A}=[\boldsymbol{a}(\theta_1)\ \boldsymbol{a}(\theta_2)\ ...\ \boldsymbol{a}(\theta_K)] \qquad (5)$$

where $\boldsymbol{a}(\theta_m)$ is the steering vector associated with direction θ_m:

$$\boldsymbol{a}(\theta_m)=[1\ e^{-jKm}\ e^{-j2Km}\ ...\ e^{-j(M-1)Km}]^T \qquad (6)$$

The received spatial correlation matrix $\boldsymbol{R}$ of the received noisy signals can be estimated as:

$$\boldsymbol{R}=\mathrm{E}\{\boldsymbol{X}(t)\boldsymbol{X}(t)^H\}=\boldsymbol{A}\mathrm{E}[\boldsymbol{S}(t)\boldsymbol{S}^H(t)]\boldsymbol{A}^H+\mathrm{E}[\boldsymbol{N}(t)\boldsymbol{N}^H(t)] \qquad (7)$$

Following the Fig.1, the antenna array is performing the mapping G: $\boldsymbol{R}^K \rightarrow \boldsymbol{C}^M$ from the space of DOAs, $\{\Theta=[\theta_1,\theta_2,\ ...\ ,\ \theta_K]^T\}$ to the space of sensor output $\{\mathbf{X}(t)=[x_1(t)\ x_2(t)\ ...\ x_M(t)]^T\}$. A neural network is used to perform the inverse mapping F: $\boldsymbol{C}^M \rightarrow \boldsymbol{R}^K$. For this task a *Radial Basis Function*

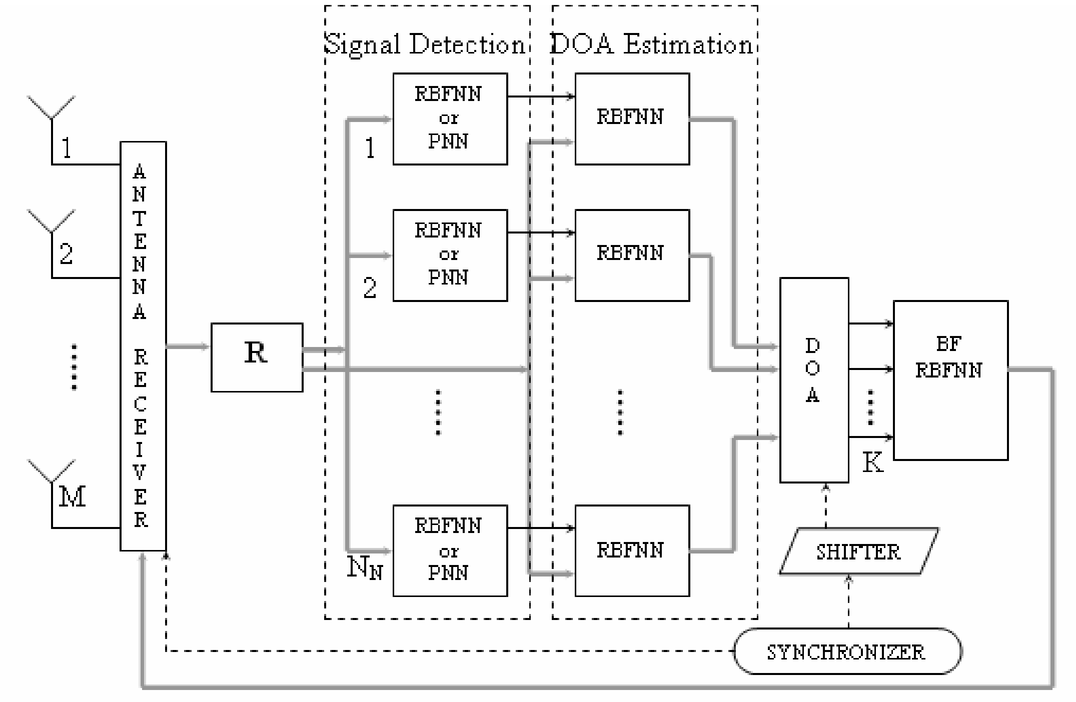

Fig.1 The Block Diagram of NN-Based Smart Antenna

(RBFNN) is used [5], instead of backpropagation neural network because the second is slower in training. In [5] the values of **R** are used at the input of the NN estimated in block **R** in Fig.1. The antenna view is divided in N_N sectors and the algorithms for detection and estimation stage are same, the difference is only in the number of nodes in the output layer. Namely, the number of the nodes in the output layer of the first stage (detection) is one (there is a signal gives one, and no signal gives 0), and the number of the nodes in the output layer of the second stage is determined by the angular resolution of the algorithm and the width of the corresponding sector.

There are a lot of learning strategies that have appeared in the literature to train RBFNN. The one used in [5] was introduced in [6], where an unsupervised learning algorithm (such as *K-means* [2]) is initially used to identify the centers of the Gaussian functions used in the hidden layer. The standard deviation of the Gaussian function of a certain mean is the average distance to the first few nearest neighbors of the means of the other Gaussian functions. This procedure allows us to identify the weights (means and standard deviations of the Gaussian functions) from the input to the hidden layer. The weights from the hidden layer to the output layer are estimated by supervised learning known as delta rule, applied on single layer networks [3]. With this procedure, for training we need 5min in detection stage and about 15min in estimation stage. An alternative is instead of using the same neural networks in both stages, to use different neural network in the first stage. The reason for this is the fact that the task of signal detection is a vector classification problem. Any input vector should be classified as 0 (there is NO signal in the corresponding sector) or 1 (there IS a signal in the corresponding sector). For this task an appropriate neural network is *Probabilistic Neural Network* (PNN), which is proposed in [7].

III. NN-Based Null-Steering Beamformer

Let $\boldsymbol{a}(\theta_1)$ be the steering vector in the direction where unity response is desired, and that $\boldsymbol{a}(\theta_2)$, $\boldsymbol{a}(\theta_3)$,. . ., $\boldsymbol{a}(\theta_K)$ are K-1 steering vectors of interference signal directions. We are trying to put nulls in these K-1 directions and to receive the signal from direction θ_1. We can create the antenna radiation pattern by associating a weight value to each antenna element. The desired weight vector is the solution to the following equations:

$$\boldsymbol{w}^H \boldsymbol{a}(\theta_1)=1 \quad (8)$$

$$\boldsymbol{w}^H \boldsymbol{a}(\theta_i)=0\ ,\quad i=2,\ldots,K \quad (9)$$

Using matrix notation this becomes:

$$\boldsymbol{w}^H \boldsymbol{A}=\boldsymbol{e}^T \quad (10)$$

were $\boldsymbol{e}$ is a vector with all zeros except the first element which is one:

$$\boldsymbol{e}=[1\ 0\ .\ .\ .\ 0]^T \quad (11)$$

For K=M, $\boldsymbol{A}$ is square matrix. Assuming that the inverse of $\boldsymbol{A}$ exists, which requires that all steering vectors are linearly independent, the solution for weight vector is:

$$\boldsymbol{w}^H=\boldsymbol{e}^T\boldsymbol{A}^{-1} \quad (12)$$

When steering vectors are not linearly independent $\boldsymbol{A}$ is not invertible and its pseudo inverse can be used. Observing the Eq.(12) it follows that the first row of the inverse of $\boldsymbol{A}$ forms the desired weight vector.

When the number of required nulls is less than M, $\boldsymbol{A}$ is not square matrix. A suitable estimate of weights may be produced using:

$$\boldsymbol{w}^H=\boldsymbol{e}^T\boldsymbol{A}^H(\boldsymbol{A}\boldsymbol{A}^H)^{-1} \quad (13)$$

RBFNN can successfully perform this Beamforming (BF) procedure and it is presented with block BF RBFNN in Fig.1. Unlike the other authors who use **R** at the input of the NN, in our case we use the DOAs at the input of the BF RBFNN. Given combination of DOAs correspond to given radiation pattern (antenna weight vector) that produce unity response in desired direction, since the NN is trained to give unity response only for one DOA (let say the first one). For multi-user detection we can divide the time into K slots, and each slot will correspond to one user. In k-th time slot the position of the desired signal direction: θ_k, in the input vector is first one. This time division multiplexing is synchronized with the antenna array.

The BF RBFNN same as in the DOA estimation stage, is consisted of three layers of neurons. The input layer has K neurons and the number of neurons in the output layer is 2M corresponding to real and imaginary parts of the weights of the antenna array elements. The hidden layer dimension is larger than that of the input layer. The NN weights from the input to the hidden layer are determined by mentioned *K-means* algorithm and the NN weights associated to the neuron connections from the hidden to the output layer are determined with NN training using *delta learning rule*. The BF RBFNN receives input vectors as combinations of DOAs and produces the antenna element weights at the output. Training pairs are produced using Eq, (12). In this case, dividing the space into sectors cannot perform the reduction of the number of training samples. The reason for this is the fact that the antenna element weights are associated to the whole antenna view. Some other means must be developed in order to decrease the number of training samples. Also as discussed in [8,9] limitation should be expected and future interest is to solve these limitations in order a large number of users to be served.

IV. NN DOA Uncertainty Problem

DOA estimation using NN concept is related to some degree of uncertainty. Namely, the actual vector of DOAs: $\Theta=[\theta_1,\theta_2, \ldots, \theta_K]^T$ is presented with estimated vector: $\Theta'=[\theta_1',\theta_2', \ldots, \theta_K']^T$ where:

$$\theta_i'=\theta_i+\Delta\theta_i\ ,\quad i=1,2,\ldots,K \tag{14}$$

The parameter $\Delta\theta_i$ receives random values with uniform distribution in the interval [-maxerr, maxerr], where maxerr is maximal angle error in degrees. This maximal error is dependant from the performances of the NN concept in the DOA estimation phase. It is very important to found out the degree of accuracy that is necessary for DOA estimation in order satisfactory beamforming to be performed. This DOA uncertainty can be decreased by appropriate NN training in DOA estimation phase or by additional training in NN beamforming stage, which will probably overburden the total training in beamforming stage. In the next section the worst case will be analyzed, that is when all DOAs are assumed to be imprecise.

V. Computer Simulations

Many different examples were investigated, here the results for the example when there are K=6 users and M=6 antenna elements are exposed. A regular linear antenna array was used with inter-element spacing of d=0.5 wavelengths. The BF RBFNN has 6 neurons in the input layer, 30 in the hidden, and 12 neurons in the output one. The centers of Gaussian transfer functions in the hidden layer were determined with *K-means* clustering algorithm. The variances were estimated as the mean distance of the three nearest neighboring centers from the corresponding center. The case for ϕ=const. and $\theta\in(0°\div180°)$ was analyzed. The users were placed in the space with mutual distance of 20°.

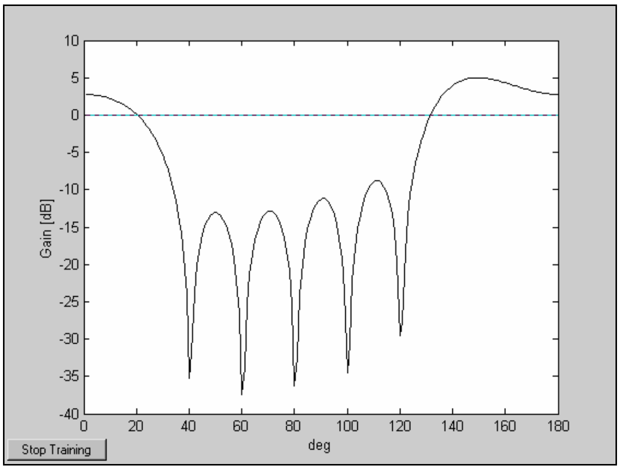

Fig. 2. Radiation pattern for 6 users at mutual distance of 20°

Fig.2 is presenting the results gained for antenna array gain (radiation pattern) for 6 users placed at mutual distance of 20°. It can be seen that NN successfully places five nulls and successfully receives the user of interest placed at 20,4°. The suppression of interference is about 30dB.

Now let assume that there are six users at mutual distance of 20 degrees and that the DOA estimation NN has performed the DOA estimation with accuracy within the range of [-maxerr,maxerr]. Fig.3 is presenting the relative error of the absolute value of the estimated array weights and Fig.4 is presenting the relative error of the argument of the estimated array weights, for the case when maxerr=0.1°. It can be concluded that null-steering NN successfully performs the beamforming almost completely neglecting the DOA uncertainty. Fig.5 and Fig.6 are presenting the results gained for the case when maxerr=0.5°. The influence of DOA uncertainty is obvious. It can be easily concluded that further enlargement of the DOA imprecision (the higher value of maxerr) will largely damage the null-steering NN beamforming performances.

We should mention that we have analyzed the case when the relative error due to imperfect NN beamforming generalization is almost zero in order to observe only the influence of the DOA estimation uncertainty.

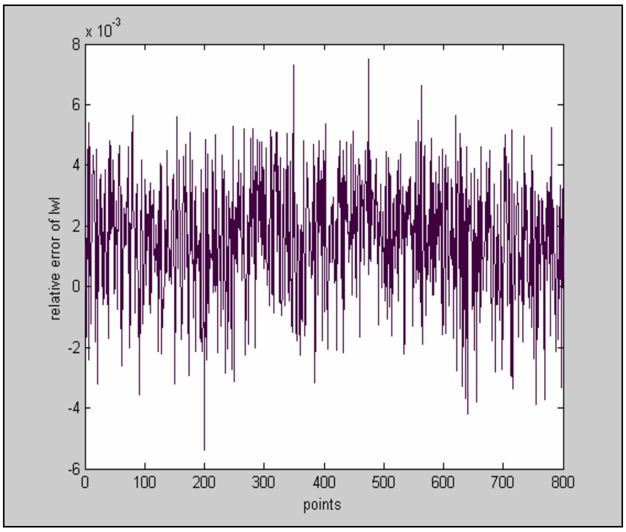

Fig. 3. Relative mean error of |*w*| for maxerr=0.1°

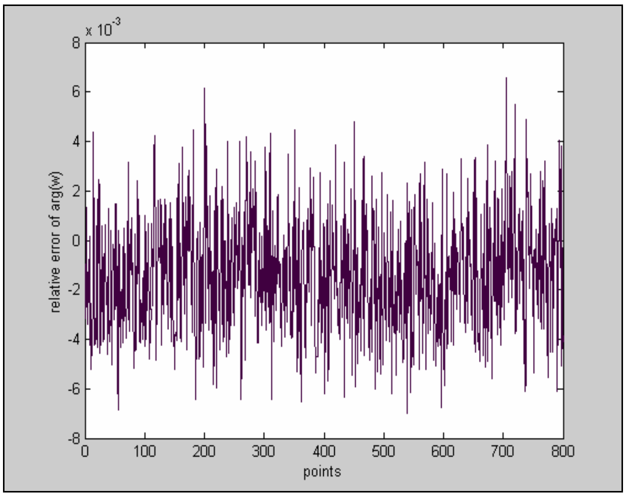

Fig. 4. Relative mean error of arg(*w*) for maxerr=0.1°

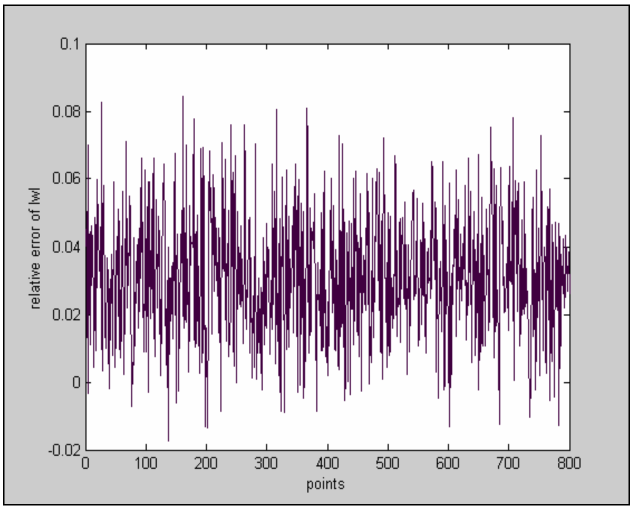

Fig. 5. Relative mean error of |*w*| for maxerr=0.5°

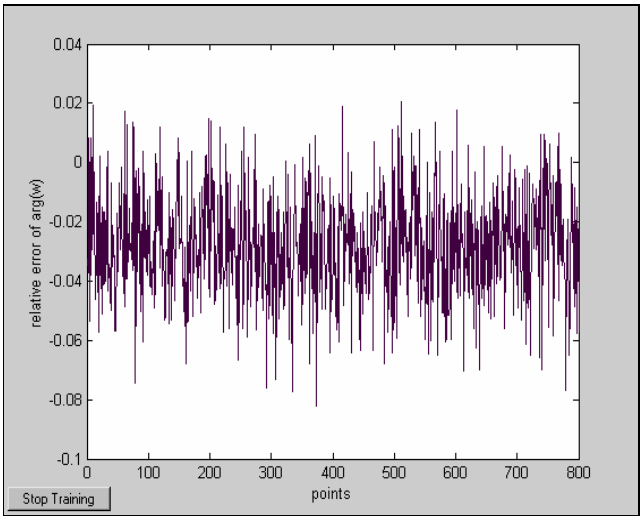

Fig. 6. Relative mean error of arg(*w*) for maxerr=0.5°

VI. Conclusions

A neural network based smart antenna solution was presented, both for DOA estimation and for null-steering beamforming providing a complete NN solution. The main issue was to present the influence of DOA uncertainty while NN estimation, on the NN beamforming performances. The results from computer simulations showed that the small DOA uncertainty doesn't infect the beamforming. For the given example the DOA precision of 0.5 degrees is providing successful NN beamforming. The DOA precision can be increased with appropriate NN training in the DOA estimation phase or additional training in beamforming stage can overcome the DOA uncertainty but in the same time it will probably overburden it.

References

[1] Yu Hen Hu, and Jenq-Neng Hwang, "*Handbook of Neural Network Signal Processing*", CRC Press LLC, Boca Raton Florida, 2001.

[2] M. M. Gupta, L. Jin, and N. Homma, "*Static and Dynamic Neural Networks*",A John Wiley & Sons, Inc., Publication, 2003.

[3] C.G. Christodoulou, and M. Georgiopoulos, "*Applications of Neural networks in Electromagnetics*", Artech House, Inc., 2001.

[4] R.O. Schmidt, "Multiple emitter location and signal parameter estimation", *IEEE Trans. Antennas propagat.*, vol.34, pp. 276-280, March1986.

[5] A.H. El Zooghby, C.G. Christodoulou, and M. Georgiopoulos, "A neural network-based smart antenna for multiple source tracking", *IEEE Trans. Antenna Prop*., vol. 48, no.5, May 2000.

[6] T.J. Moody and C.J. Darken, "Fast learning in networks of locally tuned processing units", *Neural Computat.*, vol.1, pp. 281-294, 1989.

[7] M. Sarevska, B. Milovanović, and Z. Stanković, "Alternative Signal Detection For Neural Network-Based Smart Antenna", *Neurel'04*, Belgrade, pp. 85-89, Sept. 2004.

[8] Maja Sarevska, Liljana Gavrilovska "Capabilities of Neural Network – Based Smart Antenna for DOA Estimation", *Conf.* ETAI'05, Ohrid, R.of Macedonia, pp. 94-99, September 2005.

[9] Maja Sarevska, Bratislav Milovanovic, and Zoran Stankovic, "Reliability of Radial Basis Function – Neural Network Smart Antenna", *Conf.*WSEAS'05 on Communications, Athens, Greece, July 2005