Progress In Electromagnetics Research, Vol. 126, 269–283, 2012

COMPARATIVE STUDY OF NEURAL NETWORK TRAINING APPLIED TO ADAPTIVE BEAMFORMING OF ANTENNA ARRAYS

Z. D. Zaharis*, K. A. Gotsis, and J. N. Sahalos

Radiocommunications Laboratory, Department of Physics, Aristotle University of Thessaloniki, GR-54124 Thessaloniki, Greece

Abstract—This paper presents a comparative study of neural network (NN) training. The trained NNs are used as adaptive beamformers of antenna arrays. The training is performed either by a recently developed method called Mutated Boolean PSO (MBPSO) or by a well known beamforming method called Minimum Variance Distortionless Response (MVDR). The training procedure starts by applying the MBPSO and the MVDR to a set of random cases where a linear antenna array receives a signal of interest (SOI) and several interference signals at random directions of arrival (DOA) different from each other in the presence of additive Gaussian noise. For each case, the MBPSO and the MVDR are independently applied to estimate respective excitation weights that make the array steer the main lobe towards the DOA of the SOI and form nulls towards the DOA of the interference signals. The lowest possible value of side lobe level (SLL) is additionally required. The weights extracted by the MBPSO and the weights extracted by the MVDR are used to train respectively two different NNs. Then, the two trained NNs are independently applied to a new set of cases, where random DOA are chosen for the SOI and the interference signals. Finally, the radiation patterns extracted by the two NNs are compared to each other regarding the steering ability of the main lobe and the nulls as well as the side lobe level. The comparison exhibits the superiority of the NN trained by the MBPSO.

Received 24 January 2012, Accepted 12 March 2012, Scheduled 20 March 2012

* Corresponding author: Zaharias D. Zaharis (zaharis@auth.gr).

1. INTRODUCTION

Adaptive beamforming (ABF) of antenna arrays is an interesting issue in modern communications technology [1–15]. It concerns techniques that make an antenna array change dynamically its behavior and thus become a smart antenna [10, 16–19]. The basic objective of an ABF technique is to find the excitation weights of the array elements that properly shape the radiation pattern in order to steer the main lobe towards a desired incoming signal called signal of interest (SOI) and form nulls towards respective incoming interference signals. This procedure is performed in real time and consequently the excitation weights vary with time according to the incoming signals. Moreover, a beamformer must track immediately any change in the incoming signals and thus the calculations of the ABF algorithm must be completed as fast as possible.

One of the most popular ABF techniques known for its effectiveness is the Minimum Variance Distortionless Response (MVDR) technique [12, 17]. This technique has high ability in steering the main lobe and the nulls of the produced pattern maximizing thus the signal-to-interference-plus-noise ratio ($SINR$). However, it is not capable of improving (i.e., minimizing) the side lobe level (SLL). In order to construct a beamformer, which has the ability to optimize the pattern regarding both the *SINR* and the *SLL*, several evolutionary techniques have been proposed [7–9, 18]. The success of these techniques is usually based on a proper selection of the fitness function. Nevertheless, the increased CPU time required by evolutionary techniques to converge is a major problem.

To overcome this problem, artificial neural networks or simply neural networks (NNs) can be used to implement the beamformer. A NN is a set of interconnected information processing units, called neurons, organized in a layered structure [20]. The NNs combine immediate time response and ability to provide results similar to those derived by an evolutionary technique considering that the NN is trained by this technique. Of course the NN training is a time-consuming procedure which needs data previously acquired by an evolutionary technique which is also a time-consuming procedure. However, this is not a problem because the above procedures are not executed in real time. The actual beamformer that operates in real time is the trained NN.

The present study is a comparison between NNs trained by two different effective techniques. The first one is a novel binary variant of Particle Swarm Optimization (PSO) called Mutated Boolean PSO (MBPSO) while the second is the MVDR technique. To the best of

the authors' knowledge, MBPSO-based training of NNs has never been applied before in antenna array beamforming problems. The MBPSO seems to be more effective than other binary PSO variants due to the exclusively Boolean update formulae and the novel velocity mutation process involved in the MBPSO algorithm [7]. The description of the MBPSO and MVDR algorithms as well as a comparison between MBPSO and the conventional BPSO are given in [7].

The two NNs have been applied to uniform linear arrays (ULAs) as two different beamforming techniques. Each technique starts with the training procedure using a set of random cases. Each case concerns a SOI and several interference signals received by a ULA at random directions of arrival (DOA) different from each other in the presence of additive Gaussian noise. The above directions are usually calculated by DOA algorithms [16, 17, 20–30]. The DOA of all the incoming signals and the power level of the noise signals represent the input parameters for each case. Afterwards, the MBPSO and the MVDR are applied to each case in order to extract the respective array excitation weights that steer the main lobe towards the SOI and form nulls towards the interference signals. Also, the lowest possible *SLL* value is required. The weights derived by the MBPSO are used to train the first NN while the weights derived by the MVDR are used to train the second NN. The two trained NNs are the actual beamformers of the ULA. In order to test their effectiveness, a new set of random cases is selected and the two NNs are independently applied to every case to extract the respective array excitation weights. The radiation patterns produced by the two NNs are compared to each other. The comparison shows the advantages of the NN trained by the MBPSO.

2. FORMULATION

The beamforming structure is displayed in Figure 1. The ULA consists of M isotropic elements and receives several monochromatic signals incoming from respective angles of arrival (AOA) different from each other. These signals are a SOI $s(k)$ received from AOA θ_0 and N interference signals $i_n(k)$ received from AOA θ_n ($n = 1, \ldots, N$). Each AOA is defined as the angle between the DOA of a signal and the normal to the ULA axis direction, while k refers to the time sample. The mean power of the SOI is considered as reference power. Therefore:

$$P_{SOI} = E\left\{|s(k)|^2\right\} = 1 \tag{1}$$

where $E\{\cdot\}$ refers to the mean value. The array elements also receive additive zero-mean Gaussian noise signals $n_m(k)$ ($m = 1, \ldots, M$) that

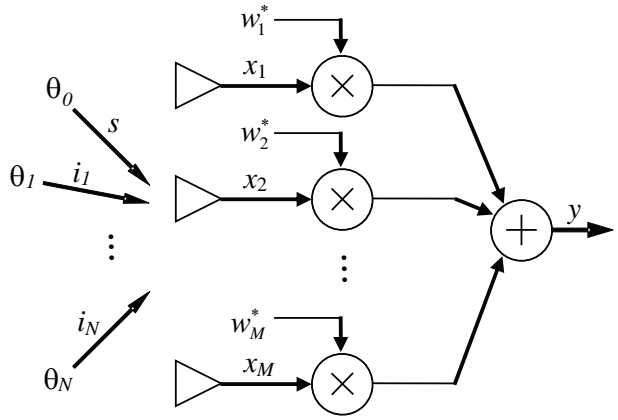

Figure 1. Block diagram of an antenna array beamformer.

have the same variance σ^2 calculated as follows:

$$\sigma^2 = 10^{-SNR/10} \tag{2}$$

where SNR is the signal-to-noise ratio in dB. The noise signals are considered to be uncorrelated with each other. Therefore, the noise correlation matrix is simplified as given below:

$$\bar{R}_{nn} = E\left[\bar{n}(k)\bar{n}^H(k)\right] = \sigma^2 I \tag{3}$$

where

$$\bar{n}(k) = [n_1(k)\ \ n_2(k)\ \ \ldots\ \ n_M(k)]^T \tag{4}$$

is the noise vector, while the superscripts T and H denote respectively the transpose and the Hermitian transpose operation.

The signals at the input of the array elements can be described by the following expression:

$$\bar{x}(k) = \bar{a}_0 s(k) + \bar{A}\bar{i}(k) + \bar{n}(k) = \bar{d}(k) + \bar{u}(k) \tag{5}$$

where

$$\bar{x}(k) = [x_1(k)\ \ x_2(k)\ \ \ldots\ \ x_M(k)]^T \tag{6}$$

$$\bar{i}(k) = [i_1(k)\ \ i_2(k)\ \ \ldots\ \ i_N(k)]^T \tag{7}$$

$$\bar{A} = [\bar{a}_1\ \ \bar{a}_2\ \ \ldots\ \ \bar{a}_N] \tag{8}$$

$$\bar{a}_n = \left[1\ \ e^{j\frac{2\pi}{\lambda}q\sin\theta_n}\ \ \ldots\ \ e^{j(M-1)\frac{2\pi}{\lambda}q\sin\theta_n}\right]^T, \quad n = 0, 1, \ldots, N \tag{9}$$

are, respectively, the input vector, the interference vector, the array steering matrix, the array steering vector that corresponds to θ_n, and

finally q is the distance between adjacent elements of the ULA. Also, $\bar{d}(k)$ is the desired input vector and $\bar{u}(k)$ is the undesired input vector, expressed respectively by:

$$\bar{d}(k) = \bar{a}_0 s(k) \tag{10}$$

$$\bar{u}(k) = \bar{A}\,\bar{i}(k) + \bar{n}(k) \tag{11}$$

The array output is expressed as follows:

$$y(k) = \bar{w}^H \bar{x}(k) = \bar{w}^H \bar{d}(k) + \bar{w}^H \bar{u}(k) \tag{12}$$

where $\bar{w} = [w_1\ w_2\ \ldots\ w_M]^T$ is the excitation weight vector. The term $\bar{w}^H \bar{d}(k)$ represents the desired output vector, while $\bar{w}^H \bar{u}(k)$ represents the undesired output vector.

The output power values that correspond to the desired and undesired signals are given respectively by:

$$\sigma_d^2 = E\left[\left|\bar{w}^H \bar{d}(k)\right|^2\right] = \bar{w}^H \bar{a}_0 \bar{a}_0^H \bar{w} \tag{13}$$

$$\sigma_u^2 = E\left[\left|\bar{w}^H \bar{u}(k)\right|^2\right] = \bar{w}^H \bar{A} \bar{R}_{ii} \bar{A}^H \bar{w} + \sigma^2 \bar{w}^H \bar{w} \tag{14}$$

where $\bar{R}_{ii} = E[\bar{i}(k)\bar{i}^H(k)]$ is the interference correlation matrix. Finally, the signal-to-interference-plus-noise ratio is given by:

$$SINR = \frac{\sigma_d^2}{\sigma_u^2} = \frac{\bar{w}^H \bar{a}_0\, \bar{a}_0^H \bar{w}}{\bar{w}^H \bar{A}\, \bar{R}_{ii}\, \bar{A}^H \bar{w} + \sigma^2\, \bar{w}^H\, \bar{w}} \tag{15}$$

It is obvious that the *SINR* is maximized when the main lobe is steered towards $s(k)$ and the nulls are steered towards $i_n(k)$ ($n = 1, \ldots, N$). Therefore, the steering ability of the array is optimized by maximizing the *SINR*. In order to satisfy the requirement for the maximum *SINR* and additionally the requirement for the lowest *SLL*, the optimal $\bar{w}$ is calculated by applying the MBPSO. This is accomplished by minimizing a suitably chosen fitness function F, which takes into account the above requirements. Then, F can be defined by the expression:

$$F = K_1 \frac{\bar{w}^H \bar{A}\, \bar{R}_{ii} \bar{A}^H \bar{w} + \sigma^2 \bar{w}^H \bar{w}}{\bar{w}^H \bar{a}_0 \bar{a}_0^H \bar{w}} + K_2 SLL \tag{16}$$

where K_1 and K_2 are used to balance the minimization of the above two terms. In fact, both requirements for maximizing *SINR* and minimizing *SLL* create a multi-objective optimization problem, which is converted into a single-objective one by balancing the above requirements into a single fitness function. In this way, the MBPSO method provides sufficient steering ability and low *SLL*. Of course, in some beamforming cases, the steering accuracy might be slightly worse

than that provided by the MVDR method. However, the MBPSO method has the advantage of minimizing the *SLL*, while the MVDR method cannot perform *SLL* control.

The MVDR method calculates the optimal $\bar{w}$ by using the following expression:

$$\bar{w}_{mvdr} = \frac{\bar{R}_{uu}^{-1}\bar{a}_0}{\bar{a}_0^H \bar{R}_{uu}^{-1}\bar{a}_0} \tag{17}$$

where

$$\bar{R}_{uu} = E\left[\bar{u}(k)\bar{u}^H(k)\right] = \bar{A}\bar{R}_{ii}\bar{A}^H + \sigma^2 I \tag{18}$$

is the correlation matrix of $\bar{u}\,(k)$.

3. NEURAL NETWORK BASED ADAPTIVE BEAMFORMING

The application of NNs in various problems of electromagnetics and wireless communications has been quite popular [20, 25, 31–33]. Due to their efficiency and speed, NNs have been used in several ABF and DOA estimation problems [2, 14, 15, 28–30]. So far in ABF applications, the NN training has been based on the mapping of the signal correlation matrix to the respective excitation weights produced by typical beamforming methods, such as the MVDR and the RCB. However, in this paper, the ABF is based on the supervised learning of NNs that map the AOA of the desired and interference signals to the corresponding excitation weights.

Two different training sets are used for the training of two different feedforward Multilayer Perceptron (MLP) NNs [20]. The training takes place in MATLAB® R2010a environment, using the Levenberg-Marquardt backpropagation algorithm [34]. Both NNs have exactly the same structure. Each one consists of an input layer of $N+1$ nodes, two hidden layers and an output layer of M nodes. The number of nodes for each hidden layer depends on the volume of the training set and the dimension of the angle vector. The criterion of their choice is the better training and testing performance. More details about training NNs with the MATLAB Levenberg-Marquardt backpropagation algorithm can be found in [28].

The input layers of both NNs are fed by the same collection of L randomly generated angle vectors $\bar{\theta}_l = [\theta_{0l}\ \theta_{1l}\ \ldots\ \theta_{Nl}]^T$ $(l = 1, \ldots, L)$, where θ_{0l} is the AOA of a SOI and θ_{nl} $(n = 1, \ldots, N)$ are the AOA of N interference signals. The output layer of the first NN called *mbpsoNN* is fed by the respective optimal excitation weight vectors $\bar{w}_{l,mbpso} = [w_{1l,mbpso}\ w_{2l,mbpso}\ \ldots\ w_{Ml,mbpso}]^T$ $(l = 1, \ldots, L)$,

produced by the MBPSO technique. Also, the output layer of the second NN called *mvdrNN* is fed by the respective optimal weight vectors $\bar{w}_{l,mvdr} = [w_{1l,mvdr}\ w_{2l,mvdr}\ \ldots\ w_{Ml,mvdr}]^T$ $(l = 1, \ldots, L)$, produced by the MVDR. Thus, the first training set is constituted by the L pairs $(\bar{\theta}_l, \bar{w}_{l,mbpso})$, while the second by the L pairs $(\bar{\theta}_l, \bar{w}_{l,mvdr})$. During the training, the weights of the neurons dynamically change in order to model the mapping between $\bar{\theta}_l$ and $\bar{w}_{l,mbpso}$ (1st NN) or between $\bar{\theta}_l$ and $\bar{w}_{l,mvdr}$ (2nd NN). The weights are updated after the entire training set has been applied to the network. One presentation of all the input/output training pairs is called an epoch. Figure 2 illustrates the NN structure and displays how the training sets are created and used for the NN training.

Once the training has been completed, both the *mbpsoNN* and *mvdrNN* may be used as adaptive beamformers. The NNs are trained to respond instantly to any given input angle vector and extract the excitation weight vector that makes the antenna array produce a radiation pattern with the desired characteristics.

The above methodology is summarized in the following steps:

1. Randomly generate L angle vectors $\bar{\theta}_l$.
2. Produce optimal $\bar{w}_{l,mbpso}$ and $\bar{w}_{l,mvdr}$, that correspond to $\bar{\theta}_l$, by applying MBPSO and MVDR respectively.
3. Apply back propagation training to two MLP NNs using the pairs $(\bar{\theta}_l, \bar{w}_{l,mbpso})$ and $(\bar{\theta}_l, \bar{w}_{l,mvdr})$, $l = 1, 2, \ldots, L$.
4. Apply an angle vector $\bar{\theta} = [\theta_0\ \theta_1\ \ldots\ \theta_N]^T$ to the input layer of each trained NN. The output layers of both NNs will extract the excitation weight vectors that steer the main lobe towards θ_0, place N nulls towards θ_n $(n = 1, \ldots, N)$ and achieve low SLL.

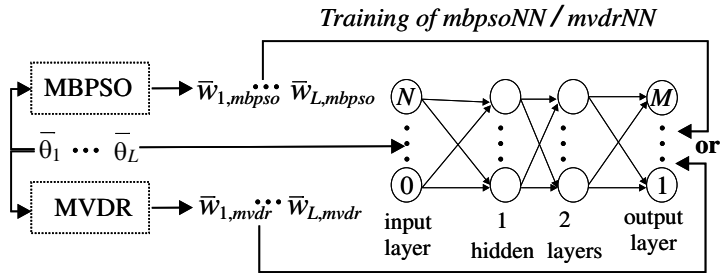

Figure 2. Block diagram that illustrates the NN structure and summarizes the NN training procedure.

4. NUMERICAL RESULTS

The two NNs have been independently applied to a 9-element ULA ($M = 9$) with $q = 0.5\lambda$. Three different scenarios are studied. The first one concerns a SOI and three interference signals ($N = 3$) in the presence of additive zero-mean Gaussian noise with $SNR = 10\,\text{dB}$. The second concerns a SOI and seven interference signals ($N = 7$) with $SNR = 10\,\text{dB}$. Finally, the third scenario concerns a SOI and six interference signals ($N = 6$) with $SNR = 30\,\text{dB}$. Initially, a training set of 5000 random cases ($L = 5000$) different from each other is selected for each scenario. Each case is a group of $N + 1$ values randomly chosen from a uniform angle distribution and given respectively to θ_n ($n = 0, 1, \ldots, N$). In each case, the MBPSO and the MVDR are applied to find the optimal excitation vectors, respectively $\bar{w}_{mbpso}$ and $\bar{w}_{mvdr}$, that produce a main lobe towards θ_0 and N nulls towards θ_n ($n = 1, \ldots, N$), while $\bar{w}_{mbpso}$ must additionally achieve the lowest possible *SLL*.

A swarm of 20 particles is used by the MBPSO in order to extract the training data for the first NN (*mbpsoNN*). The probabilities of being '1' for the three random bits c_1, c_2, and c_3 found in the velocity update expression of [7] are respectively $C_1 = 0.1$, $C_2 = 0.5$ and $C_3 = 0.5$. Also, the maximum allowed velocity $v_{\max}$ is equal to 4, while the initial value of the mutation probability m is equal to 0.10. Finally, each execution of the MBPSO is completed after 500 iterations.

The extracted 5000 $\bar{w}_{mbpso}$ are used to train the *mbpsoNN*, while the extracted 5000 $\bar{w}_{mvdr}$ are used to train the *mvdrNN*. The graph shown in Figure 3 represents the NN training performance for all the scenarios studied here and shows that training convergence is reached after a small number of runs.

The NNs derived by the training procedure are compared to each other in terms of performance by selecting a new set of 1000 random

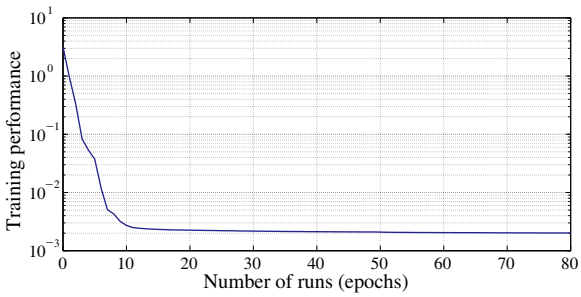

Figure 3. Performance of NN training.

cases. Then, the two NNs are independently applied to each case to extract the excitation weight vectors, respectively $\bar{w}_{mbpsoNN}$ and $\bar{w}_{mvdrNN}$, as well as the corresponding radiation patterns. The 1000 patterns derived by the *mbpsoNN* are statistically analyzed regarding the absolute angular deviation $\Delta\theta^{\text{main}}_{mbpsoNN}$ of the main lobe direction from its desired value θ_0, the absolute angular deviation $\Delta\theta^{\text{null}}_{mbpsoNN}$ of the null directions from their respective desired values θ_n ($n = 1, \ldots, N$), and finally the $SLL_{mbpsoNN}$. The 1000 patterns derived by the *mvdrNN* are also analyzed regarding the corresponding values $\Delta\theta^{\text{main}}_{mvdrNN}$, $\Delta\theta^{\text{null}}_{mvdrNN}$ and SLL_{mvdrNN}. The average values are given in Table 1, while histograms representing the statistical distribution of the main lobe and null angular deviations are illustrated in Figures 4–9. The above Figures as well as Table 1 show that both NNs are quite successful in finding the desired DOA for both the main lobe and the nulls. The greater values of $\overline{\Delta\theta}^{\text{main}}_{mbpsoNN}$ and $\overline{\Delta\theta}^{\text{null}}_{mbpsoNN}$ compared

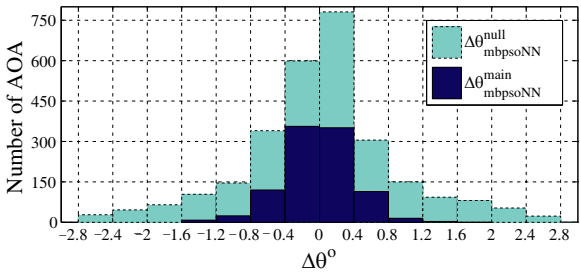

Figure 4. Histograms showing the statistical distribution of the main lobe and null angular deviations derived from the first scenario ($N = 3$, $SNR = 10\,\text{dB}$) by using the *mbpsoNN*.

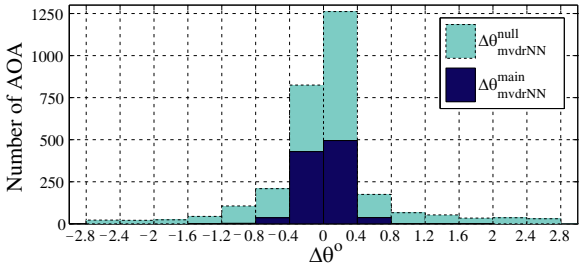

Figure 5. Histograms showing the statistical distribution of the main lobe and null angular deviations derived from the first scenario ($N = 3$, $SNR = 10\,\text{dB}$) by using the *mvdrNN*.

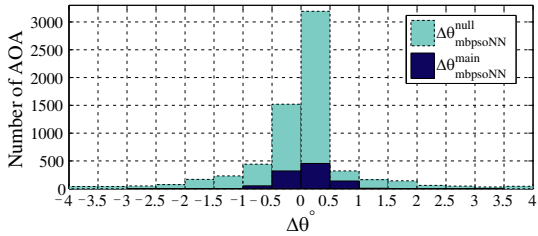

Figure 6. Histograms showing the statistical distribution of the main lobe and null angular deviations derived from the second scenario ($N = 7$, $SNR = 10\,\text{dB}$) by using the *mbpsoNN*.

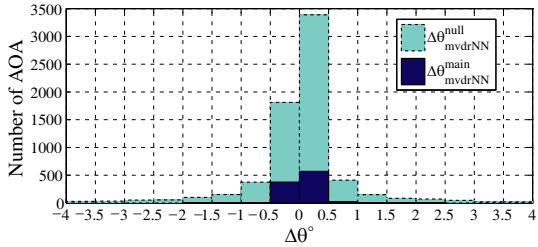

Figure 7. Histograms showing the statistical distribution of the main lobe and null angular deviations derived from the second scenario ($N = 7$, $SNR = 10\,\text{dB}$) by using the *mvdrNN*.

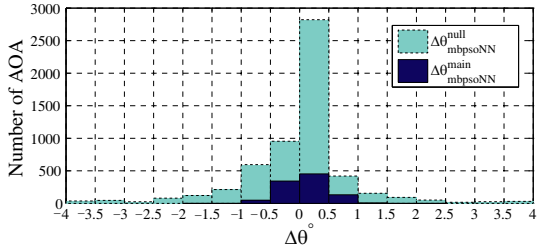

Figure 8. Histograms showing the statistical distribution of the main lobe and null angular deviations derived from the third scenario ($N = 6$, $SNR = 30\,\text{dB}$) by using the *mbpsoNN*.

respectively to $\overline{\Delta\theta}^{\text{main}}_{mvdrNN}$ and $\overline{\Delta\theta}^{\text{null}}_{mvdrNN}$ are due to the slightly worse steering ability provided by the MBPSO training data compared to the MVDR training data and also due to the stochastic nature of the MBPSO training data which complicates the training of the *mbpsoNN* and thus degrades slightly its steering accuracy.

Moreover, the better value of $\overline{SLL}_{mbpsoNN}$ compared to

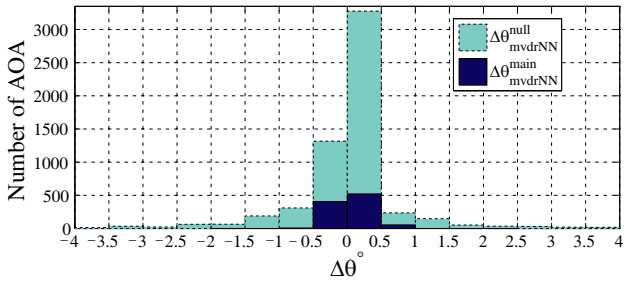

Figure 9. Histograms showing the statistical distribution of the main lobe and null angular deviations derived from the third scenario ($N = 6$, $SNR = 30\,\text{dB}$) by using the *mvdrNN*.

Table 1. Average values concerning the main lobe angular deviation, the null angular deviation and the *SLL*.

Scenario	1st	2nd	3rd
N	3	7	6
SNR	10 dB	10 dB	30 dB
$\overline{\Delta\theta}^{\text{main}}_{mbpsoNN}$	0.329°	0.432°	0.394°
$\overline{\Delta\theta}^{\text{main}}_{mvdrNN}$	0.173°	0.271°	0.236°
$\overline{\Delta\theta}^{\text{null}}_{mbpsoNN}$	0.752°	0.851°	0.816°
$\overline{\Delta\theta}^{\text{null}}_{mvdrNN}$	0.453°	0.552°	0.514°
$\overline{SLL}_{mbpsoNN}$	−13.614 dB	−10.700 dB	−11.508 dB
$\overline{SLL}_{mvdrNN}$	−12.243 dB	−9.461 dB	−10.324 dB

$\overline{SLL}_{mvdrNN}$ is predictable since the MBPSO-based training of the first NN takes into account the *SLL* minimization, while the MVDR-based training of the second NN does not. Thus, the *mbpsoNN* achieves notably better *SLL* values than the *mvdrNN*. This is also shown in Figures 10 and 11, which display the optimal radiation patterns of two typical cases of the first scenario.

Finally, it is observed that in both NNs the increase in the number of nulls is followed by an increase in the angular deviations and a decrease in the *SLL*. The angular deviations slightly increase since the NNs have more inputs and their training becomes more complicated. The *SLL* decreases because the number of nulls imposes restrictions in the radiation pattern characteristics.

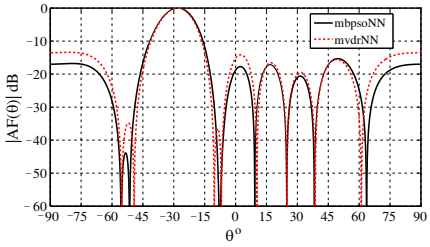

Figure 10. Optimal radiation patterns for $SNR = 10\,\text{dB}$, $\theta_0 = -28°$, $\theta_1 = -55°$, $\theta_2 = -7°$ and $\theta_3 = 38°$.

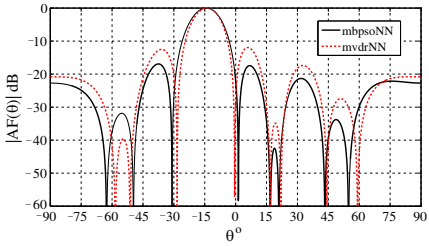

Figure 11. Optimal radiation patterns for $SNR = 10\,\text{dB}$, $\theta_0 = -14°$, $\theta_1 = -51°$, $\theta_2 = 17^o$ and $\theta_3 = 44°$.

5. CONCLUSION

A comparative study between two NN-based ABF techniques has been presented. The NNs have been trained by optimal training sets derived respectively by the MBPSO and the MVDR, in order to acquire the knowledge to produce the appropriate excitation weight vectors that maximize the *SINR* by steering the main lobe towards the SOI and the nulls towards the interference signals. The additional requirement for low *SLL* makes the NN training more challenging. Both requirements for high *SINR* and low *SLL* create a multi-objective optimization problem, which is converted into a single-objective one by balancing the above requirements, and it is solved through the MBPSO procedure. Thus, the MBPSO method provides sufficient steering ability and lower *SLL* compared to the MVDR method. The increased CPU time required by the MBPSO to extract the training data is not an issue, since the MBPSO is not involved in the real time procedure of the *mbpsoNN*, which is the actual beamformer.

The *mbpsoNN* combines the advantages of the MBPSO method, i.e., steering ability and low *SLL*, with the immediate response of a NN. On the other hand, the *mvdrNN* provides only main lobe and null steering since the MVDR method cannot perform *SLL* control. Consequently, the *mbpsoNN* is expected to have better performance compared to the *mvdrNN*. This is also verified from the simulation results.

REFERENCES

1. Li, J. and P. Stoica, *Robust Adaptive Beamforming*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006.

2. Castaldi, G., V. Galdi, and G. Gerini, "Evaluation of a neural-network-based adaptive beamforming scheme with magnitude-only constraints," *Progress In Electromagnetics Research B*, Vol. 11, 1–14, 2009.
3. Umrani, A. W., Y. Guan, and F. A. Umrani, "Effect of steering error vector and angular power distributions on beamforming and transmit diversity systems in correlated fading channel," *Progress In Electromagnetics Research*, Vol. 105, 383–402, 2010.
4. Byrne, D., M. O'Halloran, M. Glavin, and E. Jones, "Data independent radar beamforming algorithms for breast cancer detection," *Progress In Electromagnetics Research*, Vol. 107, 331–348, 2010.
5. Byrne, D., M. O'Halloran, E. Jones, and M. Glavin, "Transmitter-grouping robust capon beamforming for breast cancer detection," *Progress In Electromagnetics Research*, Vol. 108, 401–416, 2010.
6. Lee, J.-H., Y.-S. Jeong, S.-W. Cho, W.-Y. Yeo, and K. S. J. Pister, "Application of the newton method to improve the accuracy of toa estimation with the beamforming algorithm and the music algorithm," *Progress In Electromagnetics Research*, Vol. 116, 475–515, 2011.
7. Zaharis, Z. D. and T. V. Yioultsis, "A novel adaptive beamforming technique applied on linear antenna arrays using adaptive mutated boolean PSO," *Progress In Electromagnetics Research*, Vol. 117, 165–179, 2011.
8. Mallipeddi, R., J. P. Lie, P. N. Suganthan, S. G. Razul, and C. M. S. See, "A differential evolution approach for robust adaptive beamforming based on joint estimation of look direction and array geometry," *Progress In Electromagnetics Research*, Vol. 119, 381–394, 2011.
9. Mallipeddi, R., J. P. Lie, P. N. Suganthan, S. G. Razul, and C. M. S. See, "Near optimal robust adaptive beamforming approach based on evolutionary algorithm," *Progress In Electromagnetics Research B*, Vol. 29, 157–174, 2011.
10. Jabbar, A. N., "A novel ultra-fast ultra-simple adaptive blind beamforming algorithm for smart antenna arrays," *Progress In Electromagnetics Research B*, Vol. 35, 329–348, 2011.
11. Wang, W., R. Wu, and J. Liang, "A novel diagonal loading method for robust adaptive beamforming," *Progress In Electromagnetics Research C*, Vol. 18, 245–255, 2011.
12. Liu, F., J. Wang, C. Y. Sun, and R. Du, "Robust mvdr beamformer for nulling level control via multi-parametric quadratic programming," *Progress In Electromagnetics Research*

C, Vol. 20, 239–254, 2011.

13. Mallipeddi, R., J. P. Lie, S. G. Razul, P. N. Suganthan, and C. M. S. See, "Robust adaptive beamforming based on covariance matrix reconstruction for look direction mismatch," *Progress In Electromagnetics Research Letters*, Vol. 25, 37–46, 2011.
14. Zooghby, A. H. E., C. G. Christodoulou, and M. Georgiopoulos, "Neural network-based adaptive beamforming for one- and two-dimensional antenna arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 46, No. 12, 1891–1893, Dec. 1998.
15. Song, X., J. Wang, and X. Niu, "Robust adaptive beamforming algorithm based on neural network," *IEEE International Conference on Automation and Logistics (ICAL)*, 1844–1849, 2008.
16. Godara, L. C., *Smart Antennas*, CRC Press, Boca Raton, FL, 2004.
17. Gross, F. B., *Smart Antennas for Wireless Communications with Matlab*, McGraw-Hill, New York, 2005.
18. Ho, M.-H., S.-H. Liao, and C.-C. Chiu, "A novel smart UWB antenna array design by PSO," *Progress In Electromagnetics Research C*, Vol. 15, 103–115, 2010.
19. Viani, F., L. Lizzi, M. Donelli, D. Pregnolato, G. Oliveri, and A. Massa, "Exploitation of parasitic smart antennas in wireless sensor networks," *Journal of Electromagnetic Waves and Applications*, Vol. 24, No. 7, 993–1003, 2010.
20. Christodoulou, C. and M. Georgiopoulos, *Applications of Neural Networks in Electromagnetics*, Artech House, Boston, London, 2001.
21. Yang, P., F. Yang, and Z.-P. Nie, "DOA estimation with sub-array divided technique and interporlated esprit algorithm on a cylindrical conformal array antenna," *Progress In Electromagnetics Research*, Vol. 103, 201–216, 2010.
22. Park, G. M., H. G. Lee, and S. Y. Hong, "Doa resolution enhancement of coherent signals via spatial averaging of virtually expanded arrays," *Journal of Electromagnetic Waves and Applications*, Vol. 24, No. 1, 61–70, 2010.
23. Lui, H. S. and H. T. Hui, "Effective mutual coupling compensation for direction-of-arrival estimations using a new, accurate determination method for the receiving mutual impedance," *Journal of Electromagnetic Waves and Applications*, Vol. 24, Nos. 2–3, 271–281, 2010.

24. Liang, J. and D. Liu, "Two L-shaped array-based 2-D DOAs estimation in the presence of mutual coupling," *Progress In Electromagnetics Research*, Vol. 112, 273–298, 2011.
25. Kim, Y. and H. Ling, "Direction of arrival estimation of humans with a small sensor array using an artificial neural network," *Progress In Electromagnetics Research B*, Vol. 27, 127–149, 2011.
26. Xie, J., Z.-S. He, and H.-Y. Li, "A fast DOA estimation algorithm for uniform circular arrays in the presence of unknown mutual coupling," *Progress In Electromagnetics Research C*, Vol. 21, 257–271, 2011.
27. Bencheikh, M. L. and Y. Wang, "Combined esprit-rootmusic for DOA-dod estimation in polarimetric bistatic MIMO radar," *Progress In Electromagnetics Research Letters*, Vol. 22, 109–117, 2011.
28. Gotsis, K. A., K. Siakavara, and J. N. Sahalos, "On the direction of arrival (DoA) estimation for a switched-beam antenna system using neural networks," *IEEE Transactions on Antennas and Propagation*, Vol. 57, No. 5, 1399–1411, May 2009.
29. Fonseca, N., M. Coudyser, J.-J. Laurin, and J.-J. Brault, "On the design of a compact neural network-based DOA estimation system," *IEEE Transactions on Antennas and Propagation*, Vol. 58, No. 2, 357–366, Feb. 2010.
30. Zooghby, A. H., C. G. Christodoulou, and M. Georgiopoulos, "A neural network-based smart antenna for multiple source tracking," *IEEE Transactions on Antennas and Propagation*, Vol. 48, No. 5, 768–776, May 2000.
31. Patnaik, A., D. E. Anagnostou, R. K. Mishra, C. G. Christodoulou, and J. C. Lyke, "Applications of neural networks in wireless communications," *IEEE Antennas and Propagation Magazine*, Vol. 46, No. 3, 130–137, Jun. 2004.
32. Luo, M. and K.-M. Huang, "Prediction of the electromagnetic field in metallic enclosures using artificial neural networks," *Progress In Electromagnetics Research*, Vol. 116, 171–184, 2011.
33. O'Halloran, M., B. McGinley, R. C. Conceição, F. Morgan, E. Jones, and M. Glavin, "Spiking neural networks for breast cancer classification in a dielectrically heterogeneous breast," *Progress In Electromagnetics Research*, Vol. 113, 413–428, 2011.
34. Neural Network Toolbox™ User's Guide R2010a, MATLAB®, The MathWorks, Inc.