2016 International Symposium on Fundamentals of Electrical Engineering
University Politehnica of Bucharest, Romania, June 30 - July 2, 2016

Accuracy of Perceptron Based Beamforming for Embedded Smart and MIMO Antennas

K.S. Senthilkumar, K.Pirapaharan, G.A. Lakshmanan, P.R.P Hoole, and S.R.H. Hoole, *Fellow, IEEE*

***Abstract*—Array antennas have a nonlinear, complex relationship between the antenna beams generated and the array input functions that generate the steerable beams. In this paper we demonstrate the use of a simple, computationally less intensive Perceptron Neural Network with non-linear sigmoid activation function to do the synthesis of the desired antenna beam. The single neuron is used, where its optimized weights will yield the beam shape required. This paper presents a successfully implemented Perceptron and discusses the error between the desired and Perceptron generated beams The successful beam control gives high accuracy in the maximum radiation direction of the desired beam, as well as optimization in the direction of null points. Moreover, a comparison between the array antenna beams obtained using the Perceptron Single Neuron Weight Optimization method (SNWOM) and the optimized beams obtained using the Least Mean Square (LMS) method, further demonstrates the reliability and accuracy of the Perceptron based beamformer. The tests were performed for two different desired antenna beams: one braod side beam and the other with the antenna radiating in four different desired directions. The Perceptron based antenna may be embedded in the Arduino microcontroller used. It is also shown why it is not possible to get a single beam, linear array antenna with the Perceptron based array reported herein.**

***Index Terms*—Smart Antenna, Adaptive Array, Adaptive Beamforming, Artificial Neural Network.**

I. Introduction

The Long Term Evolutionary (LTE) communication system is the latest wireless communication technology in use which provides high speed and high capacity wireless communication when compared to the 3G wireless systems. However, what is referred to as the LTE 4G systems is still limited in many areas Smart antennas and the MIMO system are one of the available ways to increase the rapidly increasing demand on capacity. The peak data rate is proportional to the number of antennas at the sending and receiving ends. This paper specifically focuses on beam steering using a fast neural network adaptation to direct the beam towards particular users and/or to steer nulls to reduce interference. This is a crucial role of a smart antenna which is able to provide electrical tilt, beam width and azimuth control suitable for handling moving traffic patterns. The smart antenna solution is far more versatile, and cheaper provided low memory, fast beam steering techniques such as that reported in this paper are used at the base stations. In parallel, the development of cost effective fast cell site addition, increasing the number of cell sectors and bandwidth, and better air interface capabilities will be critical to moving into the proper 4G systems of the future.

Artificial Neural Networks (ANN) are powerful techniques to be used where the mathematical relationship between input and out can be reliably established [1], [2]. The ANN is able to approximately model the input-out relationship by optimizing the weights through using known input-output training pairs. Once the training is done, it is able to obtain the needed antenna radiation beam for a given set of inputs by adaptive signal processing [3-6]. In this paper is presented a simple Perceptron that is able to rapidly adjust the weights by adaptive signal processing for given input-put pairs, and then generate a desired radiation beam using the converged weights. This paper considers the accuracy of a simple, fast perceptron Neural Network to form beams using linear arrays to generate beams that may communicate at a traffic junction along all four roads that meet at the junction, or a single long tunnel in underground communication where the beams need to be generated by the linear array of antennas [7, 8].

II. Single Neuron Weight Optimization Model (SNWOM)

In this section we briefly describe the single neuron modal to optimize the weights which will be used in adaptive beamforming. In the perceptron model as shown in Fig.1, a single neuron with a linear weighted net function and a threshold activation function also known as transfer functions are employed. The model has three parts and at the first part inputs (x_1, x_2 …., x_n) are multiplied with individual weights (w_1, w_2 ...w_n). In the second part of a simple perceptron is the net function that sums all weighted inputs and bias as in

$$z = b + \sum_{k=1}^{n} w_k x_k \tag{1}$$

In the final part of a simple perceptron, the sum of previously weighted inputs and bias is passing through a transfer function to get the output. In case of the linear activation function, the artificial neuron is doing simple linear transformation over the sum of weighted inputs and bias b. There is no single best method for nonlinear optimization and it is based on the characteristics of the problem to be solved.

Paper submitted for review on 20 June, 2016.

K.S. Senthilkumar is with St. George University, Grenada.

K. Pirapaharan is with the Department of Electrical and Communications Engineering, Papua New Guinea University of Technology, Lae 411, PNG.

G.A. Lakshmanan and P.R.P. Hoole are with the Department of Electrical and Electronic Engineering, Faculty of Engineering, University Malaysia Sarawak, Sarawak, Malaysia.

S.R.H. Hoole (srhhoole@gmail.com), the corresponding author, is on the Election Commission, Sri Lanka, on leave from the Department of Electrical and Computer Engineering, Michigan State University.

We simplify the calculation complexity to reduce the processing delay. Hence we have used a single neuran for this problem and a nonlinear activation function σ to find out the output *y*:

$$y = \sigma(z) \quad = \frac{1}{1+e^{-z}} \tag{2}$$

In order to train the weights to meet the desired output of y_0, the deviation Δ is obtained and the weights are iterated until the trained means error *TMR* is below the predefined value, where the deviation, and trained means error are as in (3) and (4) respectively.

$$\Delta = y_0 - y \tag{3}$$

$$TMR = \frac{\Delta}{y_0} \times 100 \tag{4}$$

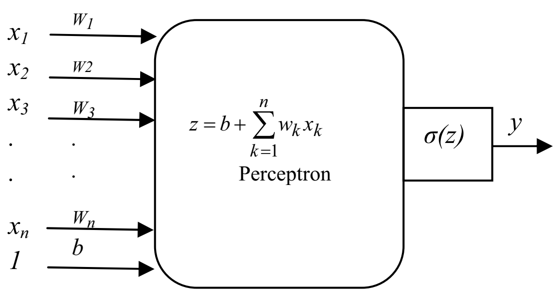

Fig.1. Perceptron model for weight optimization

Also the weights are adjusted in every iteration using the deviation and the selected learning rate, which is also known as coefficient k_0 as given in:

$$w_i = w_i + (k_0 \times \Delta \times x_i) \tag{5}$$

The iteration is allowed until it either reaches the *TMR* below the predefined value TMR_m or the defined maximum number *N* of iterations. A simple array of dipoles placed in a straight line is considered as the array model. We have tested array models with five and seven elements placed in a straight line. The array model equation with respective coefficients can be given for five elements array as in (6).

$$\begin{aligned} &w_1 e^{2j\beta d\cos\phi} + w_2 e^{j\beta d\cos\phi} + w_3 \\ &+ w_4 e^{-j\beta d\cos\phi} + w_5 e^{-2j\beta d\cos\phi} = f(\phi) \end{aligned} \tag{6}$$

Similarly, we can write the seven element model:

$$\begin{aligned} &w_1 e^{3j\beta d\cos\phi} + w_2 e^{2j\beta d\cos\phi} + w_3 e^{j\beta d\cos\phi} \\ &+ w_4 + w_5 e^{-j\beta d\cos\phi} + w_6 e^{-2j\beta d\cos\phi} \\ &+ w_7 e^{-3j\beta d\cos\phi} = f(\phi) \end{aligned} \tag{7}$$

where $f(\phi)$ is the desired beam function.

III. Accuracy of the Perceptron Beam Former

Fixing the desired beam function as shown below,

$$f(\phi) = \cos 2\phi \tag{8}$$

and keeping the distance between two elements as half wavelength, we have optimized the weights for five and seven elements to find the actual output using the above SNWOM model with initial weights, bias and learning rate which is also known as coefficient. For training, we have used different angles ϕ in the range of 0^0 to 360^0. During the testing process also we have used different angles ϕ in the range of 0^0 to 360^0. Having obtained the optimized weights after convergence, we have drawn the radiation patterns using the optimized weights and compared with the radiation patterns of the desired beam for five elements array as shown in Fig. 2.

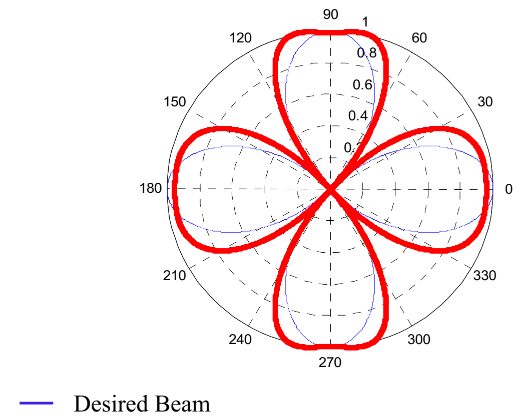

Fig. 2. Comparison of Radiation pattern between desired beam and optimized beam by SNWOM when the number of adaptive array elements is five.

Similarly, we have optimized the weights using the stated SNWOM. The optimized results are shown for seven elements in Fig. 3. As we have expected, with increased number of elements, the adaptive array beamforming is very much closer to the desired beam. However the amplitudes in the 0^o and 180^o are better in the five element array than in the seven element array. That is due to the characterstics of the desired beam selected.

In order to make the comparison of accuracy between 5 and 7 element models, the error between the desired and optimized values' corresponding angles is shown in Fig. 4 and Fig.5, respectively. The error analysis clearly displays that when the number of elements increases the error range reduces with high oscillation.

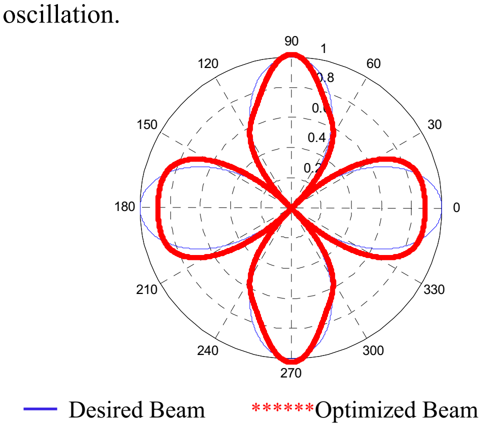

Fig. 3. Comparison of Radiation pattern between desired beam and optimized beam obtained by SNWOM when the number of adaptive array elements is seven.

With the five element array, both end fire beams and the broadside beams have larger beamwidths than the desired beamwidth. With regard to peak radiation, the broad side beams match the desired peak radiation, whereas with the end fire beams, the maximum field strength is less than the desired maximum field strength (Figs. 2 and 4). With seven elements, there is almost a perfect match in the case of the broad side beams, whereas in the end fire beams, the widths of the Perceptron-generated and desired beams match; the peak radiation drops to lower values than for the desired maximum (Figs. 3 and 5).

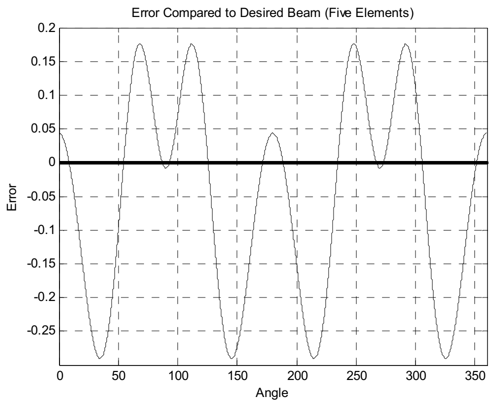

Fig.4. The error between desired and optimized beam with the corresponding angle when the number of elements is five.

In order to further test the precision of the SNWOM method with variety of desired function, we select a desired function as [7]:

$$f(\phi) = \frac{1}{9}\left|3 + 4\cos(\pi\cos\phi) + 2\cos(2\pi\cos\phi)\right| \quad (9)$$

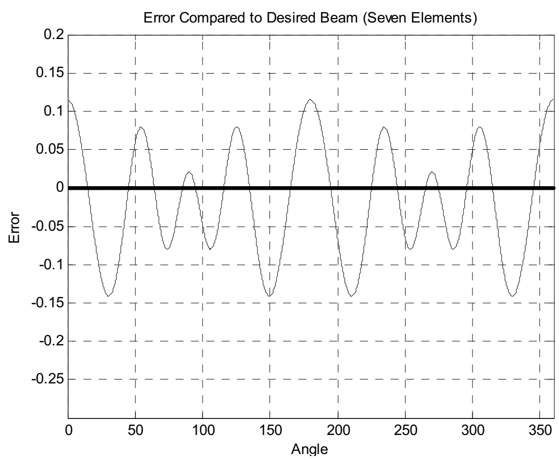

Fig.5. The error between desired and optimized beam with the corresponding angle when the number of elements is seven.

We optimize weights by taking the distance between two elements as half-wavelength for five and seven elements using SNWOM. The optimized radiation patterns are compared with the desired radiation patterns in Fig. 6 and Fig. 7 for five and seven elements, respectively.

We observe the close match between desired and optimized radiation patterns. It is evident from the results that the desired narrow beam could not be achieved using the five element model while it is feasible with the seven element model. Therefore, it can be recognized that the narrow desired beam requires more dipole elements.

The error between desired and optimized values corresponding to the angle for five element and seven element models is shown in Fig. 8 and Fig. 9, respectively. Similar to previous results, the range of error reduces while the frequency of oscillations increases with increasing number of elements. In this broadside array antenna, using the five element array, the beamwidth of the Perceptron-generated array factor is larger than the desired beamwidth (Figs. 6 and 8). But the end fire side lobes are very small. Hence the power degradation takes place along the direction of the main beams. For the eleven element broadside array antenna (Figs. 7 and 9), while the Perceptron-generated broadside beams almost perfectly match the two broadside beams, the power degradation along the end fire directions becomes significant related to larger side lobes in the end fire directions.

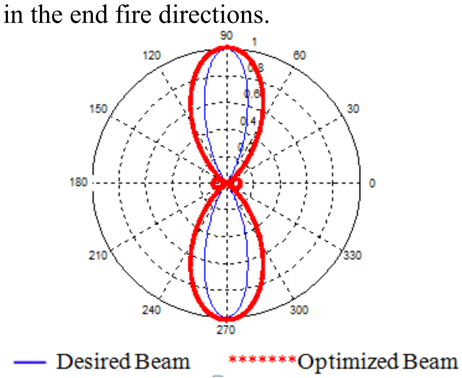

Fig. 6. Comparison of Radiation pattern between desired beam and optimized beam obtained by SNWOM when the number of adaptive array elements is five.

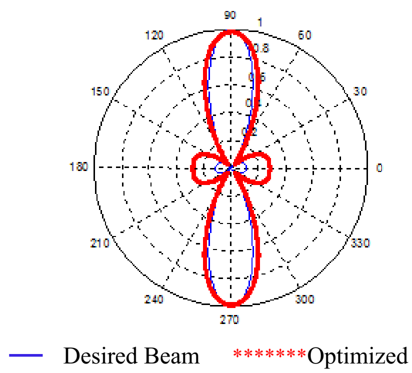

Fig. 7. Comparison of Desired Beam and Optimized Beam Radiation Patterns by SNWOM for 7 adaptive array elements.

Even though, the precision of the SNWOM depends on the dipole placement and the characteristics of the desired beam selected, it is a fast, efficient and simple method for weight optimization compared to the previously proposed neural network based adaptive beamforming methods.

IV. Comparison of SNWOM Perceptron Antenna Beams with LMS Optimized Beams

In order to compare the accuracy of the weights obtained from the Perceptron (SNWOM) method, the Perceptron generated results were compared with the results obtained using the LMS method. The desired beam pattern function to which the Perceptron and LMS beams are optimized is the radiation array pattern of a broadside array defined in (9):

$$f(\varphi) = \frac{1}{9}\left(3 + 4\cos(\pi\cos(\varphi)) + 2\cos(2\pi\cos(\varphi))\right) \quad (9)$$

We have selected a linear array antenna with five dipole elements with weights w_1, w_2, w_3, w_4 and w_5 associated with each of the five elements, and optimized the weights using both SNWOM and the LMS method. Fig. 10 shows the comparison of results of SNWOM and LMS method. Also given in the figure is the desired pattern obtained from the eact solution given by (9). For the fivel lement array it was found that the LMS method gave a beam which is more identical to the desired beam (Fig. 10).

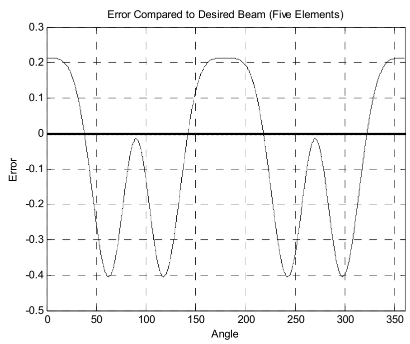

Fig.8. The error between desired and optimized beam with the corresponding angle when the number of elements is five.

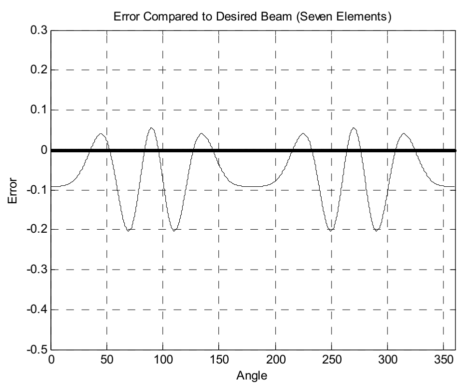

Fig.9. The error between desired and optimized beam with the corresponding angle when the number of elements is seven.

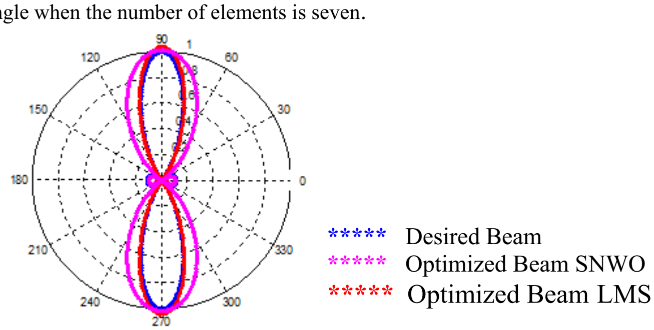

Fig. 10. Comparison of radiation patterns obtained from SNWOM and LMS methods with the desired pattern employing 5 dipole elements

The test was repeated for an array antenna with seven elements, and thus seven weights to be optimized. The results are shown in Fig. 11. With seven elements, both SNWOM and LMS give beams that closely match with the desired beam. The SNWOM was found to be slightly inferior to the LMS beam, in that it gave rise to slightly larger side-lobes in the 0^{o} and 180^{o} directions-Fig. 11.

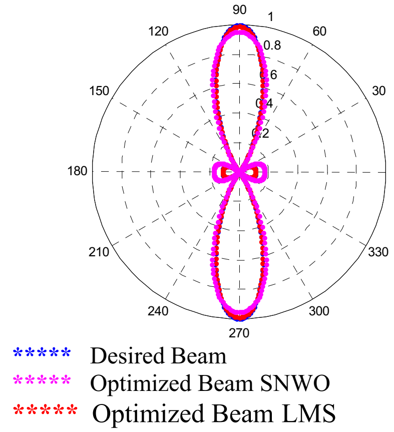

Fig. 11. Radiation patterns from SNWOM and LMS methods 7 dipoles

In order to further compare the two methods, the results were compared for the desired radiation array function defined in:

$$f(\varphi) = \cos(2\varphi) \quad (8)$$

The results are shown for both 5 elements and 7 elements in Fig.12 and Fig.13, respectively. Again it was observed that the radiation patters of both SNWOM and LMS method matched each other for the seven element array antenna (Fig. 13). With five elements, although the LMS method gave a beam that matched the desired beam in beamwidth, the radiation beam produced by the SNWOM matched the desired beam in strength in the direction of maximum radiation (Fig. 12).

V. Embedded Perceptron Based Beam Former

A. *Introduction*

Smart antennas are adaptive and hence, they are employed to improve efficiency in digital wireless systems. Smart antenna technology is emerging in its importance in wireless communication systems. The accuracy and speed of these antenna arrays are improved with the perceptron based algorithm presented in this paper which can easily beam form towards the different wireless objects such as mobiles. As reported above, the Perceptron may be effectively used to compute the beam forming vectors of the signal.

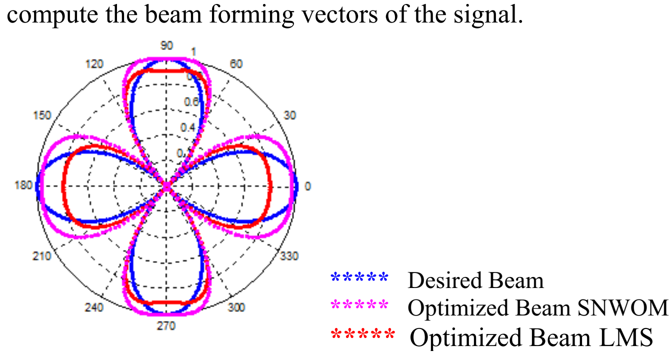

Fig. 12. Comparison of radiation patterns obtained from SNWO and LMS methods with the desired pattern employing 5 dipole elements

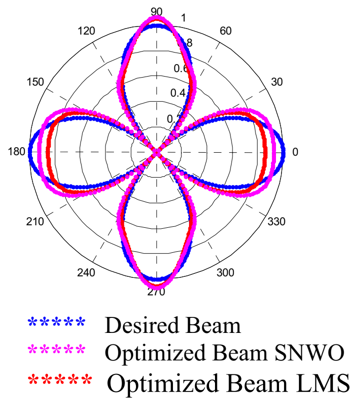

Fig. 13. Comparison of radiation patterns obtained from SNWO and LMS methods with the desired pattern employing 7 dipole elements

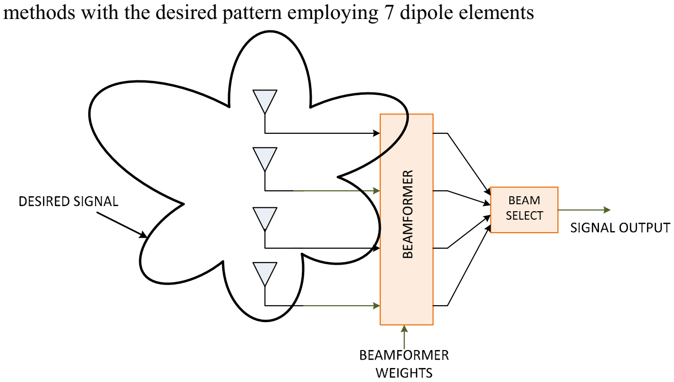

Fig. 14. MIMO adaptive Array Antennas

In beamforming, the mobile units to which the signals are to be sent are first located and then an array antenna beam or radiation pattern of the antenna array is created by adding signal phases. At the same instance, the mobiles which are not needed could be discarded from the pattern. This could be implemented with the help of delay line filters. In this paper we implement our design with Multiple Input-Multiple Output (MIMO) antennas in accordance with the IEEE 802.11n standard. It is the most efficient among the smart antennas such as Multiple Input-Single Output (MISO) and the Single Input – Multiple Output (SIMO) systems. Figure 14 depicts an adaptive array antenna.

B. Embedded Implementation of Perceptron Based MIMO Antennas in Neural Network

Conceptually, the Perceptron model for weight optimization is broadly classified into three blocks, namely: The Connection Layer, the Neuron and network Layer and the back propagation layer. Each block is further sub classified. A connection block determines how much of the signal is passed to the neuron with its respective strength of the weight, w_n. Figure 15 depicts a connection block and Figure 16 depicts the blocks of the Perceptron model.

Floating point arithmetic is employed throughout these connection blocks. The output of the connection block next feeds the neural network layer.

ConnEntry (x): is the value of the receiving signal being fed into the connection block.
Weight (m): is the value that amplifies or weakens the ConnEntry value.
ConnExit (y): is the output value of the connection to the neuron

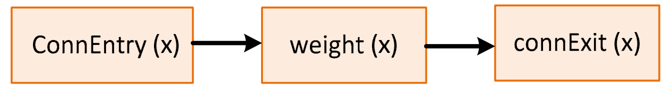

Fig. 15. The connection blocks of the SNWOM

The core of the neural network is the Neuron and the main purpose of the Neuron is to accumulate, add bias value and add to the values from the connection block and to process the activation function, (A). A enables the neuron to make decisions based on the values provided to it. The flow in Figure 16 is explained as follows:

At the connection section:
cNn1, cNn2 = Neuron connection, connEntry1, 2.
bEn = Neuron Bias Entry = 1 (always).
cW1, cW2 = Connection weight 1, 2.
cEx1, cEx2 = Connection Exit 1, 2 where cEx1= cEx1 x cEn1 and cEx2= cEx2 x cEn2
bW= Neuron Bias weight and bEx = Neuron Bias Exit bEx = bWx1.
Neuron Input value = cEx1 + cEx2 + bEx.
Neuron Output value , y = 1/(1+ exp (-1 x Neuron Input value).

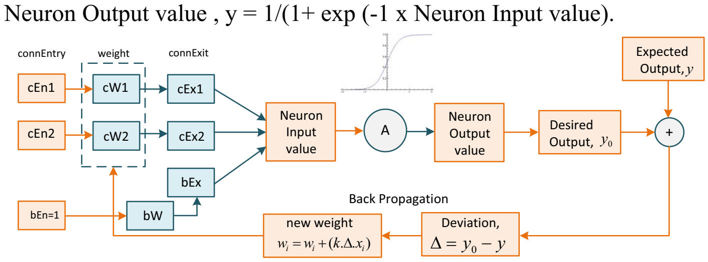

Fig. 16. The Neural Network Block of Perceptron Algorithmic Implementation.

The control section of the embedded system performs the various tasks and functions of the neuron shown in Figure 12. The controller also calculates the delta error, sets the learning rate, adjusts the error by a specific amount through back propagation iteratively and readjusts back the weights in an adaptive manner.

The Arduino microcontroller uses a set of C/C++ functions and a variation of them. Arduino IDE is freeware and it is easily downloadable and configured for diverse applications. The various adaptor boards or Shields are easy to be plugged and programmed on top of the Arduino microcontroller board.

VI. Linear Array and Mirror Image

In this section we shall show why the linear array tends to produce a symmetrical beam, thus making it problematic to get a single beam antenna with the Perceptron neural network. When all dipoles are in a straight line, the radiation pattern is symmetrical over the axis of the dipole placement. Hence a rotatable single beam cannot be obtained.

The respective complex current phasors of the dipoles are taken as I_1, I_2, and I_n. Hence the electric field (far-field terms) at the observation point Pin Fig. 18 could be given as:

$$E = A_0 I_1 e^{-j\beta r_1} + A_0 I_2 e^{-j\beta r_2} + \ldots + A_0 I_n e^{-j\beta r_n} \quad (10)$$

where A_0 and β are a constant magnitude and the phase constant, respectively.

Substituting for distances r_1, r_2, and r_n in terms of the distance from the observation point from the origin, r, we can simplify (10) as follows:

$$E = w_1 e^{j\beta(x_1 \cos\varphi + y_1 \sin\varphi)} + w_2 e^{j\beta(x_2 \cos\varphi + y_2 \sin\varphi)} + \ldots + w_n e^{j\beta(x_n \cos\varphi + y_n \sin\varphi)} \quad (11)$$

where $w_n = A_0 I_n e^{-j\beta r}$.

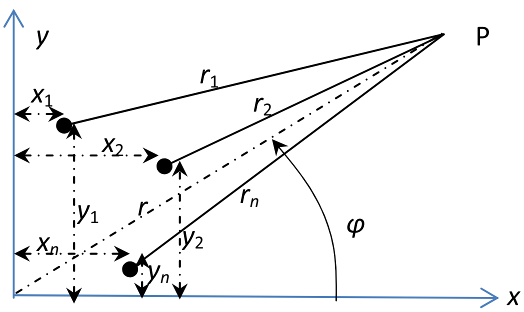

Fig.18. Schematic diagram of dipole placement

Consider all the dipoles to be in a straight line along the x axis. Thus, $y_1 = y_2 .... = y_n = 0$. Hence the expression (11) is simplified thus:

$$E(\varphi) = w_1 e^{j\beta x_1 \cos\varphi} + w_2 e^{j\beta x_2 \cos\varphi} + \ldots + w_n e^{j\beta x_n \cos\varphi} \quad (12)$$

The electric field expression, when the angle φ is in opposite direction, could be obtained replacing φ by –φ in (12).

$$E(-\varphi) = w_1 e^{j\beta x_1 \cos(-\varphi)} + w_2 e^{j\beta x_2 \cos(-\varphi)} + \ldots + w_n e^{j\beta x_n \cos(-\varphi)} \quad (13)$$

Since $\cos(-\varphi) = \cos\varphi$, replacing $\cos(-\varphi)$ in (13) with $\cos\varphi$, we obtain,

$$E(-\varphi) = w_1 e^{j\beta x_1 \cos\varphi} + w_2 e^{j\beta x_2 \cos\varphi} + \ldots + w_n e^{j\beta x_n \cos\varphi} = E(\varphi) \quad (14)$$

Since $E(-\varphi) = E(\varphi)$, the electrical field is symmetrical over the x axis whatever the weight values may be. Hence it is impossible to get a single beam in one the side of the x axis. The only possible solution exists for weights for a single beam is in the positive/negative direction of the x axis. In that case, the beam cannot be rotated to other directions. Therefore, a rotatable single beam cannot be obtained by means of weight optimization when all the dipoles are in a straight line.

VII. Conclusion

The Perceptron Neural network reported in this paper for linear array beamforming is a fast and accurate method, requiring little computational power or time compared to the more commonly used multilayer ANN for beam forming. The Perceptron's accuracy was tested for generating beams in both simultaneous broadside and end fire directions, as well as for the array generating broad side beams only. When the elements are increased in number, the power loss in the main beam significantly reduces, with the beamwidths becoming closer to the desired beams. This is true for both types of antennas experimented with. When the Perceptron was used to construct a broadside beam antenna, it was found that increasing the number of elements gave a greater match for the two main beams, whereas increased elements tended to have bigger end fire side lobes for the Perceptron based broadside antenna. The Perceptron based linear array antenna does not yield a single beam rotatable antenna, which is a limitation. The linear antenna tends to produce a fair amount of back radiation as well.

References

[1] H.L. Southall, J.A. Simmers,J.A and T.H. O'Donnell, "Direction finding in phased arrays with a neural network beamformer," *IEEE Transactions on Antennas and Propagations*, 43:1369-1374, 1995.

[2] A.H. El Zooghby, C.G. Christodoulou, and M. Georgiopoulos, "Performance of radial basis function network for direction of arrival estimation with antenna arrays," *IEEE Transactions on Antennas and Propagations*, vol.45, 1997, pp.1611-1617.

[3] Robert Mozingo and Thomas, Miller,. *Introduction to Adaptive Arrays*. New York: Wiley, 1980.

[4] A.H. El Zooghby, C.G.Christodoulou, and M. Georgiopoulos, "Neural Network-Based Adaptive Beamforming for One and Two-Dimensional Antenna Array," *IEEE Transactions on Antennas and Propagations*, vol. 46, 1998, pp. 1891-1893.

[5] N.Y. Wang, P. Agathoklis, and A. Antoniou,A., 2006. A new DOA Estimation Technique Based on SubarrayBeamforming, *IEEE Transactions on Signal Processing,* vol. 54, no. 9, 2006,:3279-3290.

[6] Jean-Luc Fournier, *et al*,,. Phased Array Antenna Controlled by Neural Network FPGA, Loughborough Antennas and Propagation Conference, 14-15 November 2011, Loughborough, UK, , pp. 1-5.

[7] P.R.P. Hoole, Smart Antennas and Signal Processing in Communications, Radar and Medical Systems, WIT Press, UK, 2001

[8] A. Rawat, R.N. Yadav and S.C. Shrivastava, "Neural Network applications in smart antennas: A review," *Int J of Electronics and Communications.*vol. 66, 2012, pp. 903-91

.