ANALYSIS, SYNTHESIS AND DIAGNOSTICS OF ANTENNA ARRAYS THROUGH COMPLEX-VALUED NEURAL NETWORKS.

J. C. Brégains (Student Member, IEEE), and F. Ares (Senior Member, IEEE).

Radiating Systems Group, Department of Applied Physics,
University of Santiago de Compostela
Campus Sur, 15782, Santiago de Compostela, Spain
Tel: +34 981547084; Fax: +34 981547046;
Emails: faares@usc.es (Ares); fajulio@usc.es (Brégains)

ABSTRACT

It is shown in this paper that when artificial neural networks are extended to be complex-valued, they can be incorporated as a very powerful and effective tool in the design and implementation of antenna arrays.

1. INTRODUCTION

Artificial neural networks (ANN) have demonstrated to be a very useful tool in a wide range of scientific, industrial and business applications [1]. In antenna implementations, they have found several branches to deal with, as, for example [1,2]: in antenna arrays, for signal –and power– control; in radar, for targets detection and recognition; in microwave devices, for design and optimization; and even in computational electromagnetics, for reducing the number of operations during certain calculations. In antenna array design, some works due to Christodoulou, Southall, O'Donnell, and Mailloux (some of them with collaborators) are of concern [2, 5]. The tools presented by those authors are limited to work with real-valued ANN, which, in principle, are the ones initially defined by the pioneers of that field (the first work on ANN is attributed to McCullough and Pitts, in a paper dated on 1943) [1]. Nevertheless, some authors have realized that, when dealing with signal processing or analogous mathematical subjects, it would be desirable for the ANN to be complex-valued [5,6]. As seen in next sections, CVANNs (Complex-Valued ANNs) have simpler architectures; consequently, their computations are faster. Besides, the configuration of the ANNs so considered become easier to understand from a point of view based on mathematical analogies. In what follows, it is assumed that the reader has some familiarity with the theory of antennas, but little less knowledge about the ANNs.

2. DEFINITIONS AND APPLICATIONS.

2.1 ARTIFICIAL NEURAL NETWORKS.

Even definitions of ANN are encountered in any specialized book [1,2], it is presented here a brief look about them in order to introduce rapidly some concepts and see clearly their straight applications to array antennas design. Figure 1 depicts one of the most used architectures of an ANN. It consists of:

a) A set of variables (input vector) $\{x_p\} \in \mathbb{R}$, $p \in \{0, 1, 2, \ldots P\}$ ($\mathbb{R}$ is the set of real numbers).

b) A set of weights (weight matrix) $\{w_{qp}\} \in \mathbb{R}$, $q \in \{0, 1, 2, \ldots Q\}$, $p \in \{0, 1, 2, \ldots P\}$.

c) A set of Q+1 neurons (a layer). In the architecture shown in figure 1 (full connected and feedforward ANN [1]), the neuron q receives the sum of all the P+1 inputs, whose values have been previously multiplied by their corresponding weights, to obtain $v_q = \sum_{p=0}^{P} w_{qp} x_p$.

d) A set of transfer functions $\phi_q(.) \in \mathbb{R}$. Every ϕ_q implements some calculation on v_q. They are usually taken to be the same for all neurons in a layer.

e) A set of Q+1 outputs (output vector), whose values are given by the abovementioned operation $\{y_p\}=\{\phi_q(v_p)\} \in \mathbb{R}$.

The outputs $\{y_p\}$ can be used as inputs of another neurons layer, performing a multilayer ANN. In this work a single-layered ANN will suffice. A survey for architectures of ANN, and the properties of their variables and transfer functions could be found in [1]. As seen, all the aforementioned operations constitute a mapping $R^{P+1} \to R^{Q+1}$ through

$$y_p = \phi_q\left(\sum_{p=0}^{P} w_{qp} x_p\right). \tag{1}$$

The utility of the ANN is based on its ability to perform a desired mapping from certain set of input vectors to certain set of output vectors through the proper selection of $\phi_q(.)$ and w_{qp}. The choice of ϕ_q is, more or less, arbitrary, and specified by the problem the designer has in hands. The selection of the w_{qp}, nevertheless, can be more systematically analyzed, and they are usually calculated through certain algorithm, prior to the operation of the neural network, called "training process (TP)" [1,2]. In short, the training process consists in the adjustment of the weight matrix w_{qp} values (such an adjustment being based on a specified "learning rule") through a sequential presentation to the ANN of certain number of R input-output vectors $\{\{x_p\}_r,\{d_q\}_r\}$, so that the Euclidean distance between $\{yq\}_r$ and $\{dq\}_r$ $\forall$ $r \in [1,2,..R]$ is minimized. After the w_{qp} have been selected, and if the TP was successful, the ANN is ready to respond correctly, not only to the training set, but to any other related input vector, not considered during the TP[1].

If ϕ_q are taken to be the identity function $\phi_q(v_p)=v_p$, eq. (1) can be written in matrix form as

$$y_p = \sum_{p=0}^{P} w_{qp} x_p \equiv [Y] = [W][X] \tag{2}$$

The utility of eq. (2) will be apparent in next sections.

2.2 FIELD OF AN ARRAY OF ISOTROPIC ELEMENTS.

With no loss of generality, let us consider the far-field radiation expression of a linear array of N+1 isotropic elements with excitation distribution $\{I_n\} \in \mathcal{C}$ (the set of complex numbers) and equispaced along the *z*-axis a distance d apart, $n \in \{0,1, \ldots N\}$. In frequency domain, its expression, often called array factor, is given by[2]

$$F(u) = \sum_{n=0}^{N} e^{jnu} I_n \, . \tag{3}$$

with $u=(2\pi/\lambda)d\cos\theta$, being λ the wavelength and $\theta \in (0,180º)$ measured from the z-axis.

2.3 UTILITY CASES.

2.3.1 ARRAY PATTERN ANALYSIS

Once d, N and λ have been fixed, we can obtain, by discretizing eq. (3) to M+1 measurement points $F_m(u_m)$, $m \in \{0,1,2,\ldots M\}$, the following matrix expression:

$$F_m = \sum_{n=0}^{N} e^{jnu_m} I_n \equiv [F] = [e][I] \, . \tag{4}$$

It can be seen that equations (2) and (4) are equivalent if the ANN is allowed to be complex valued, and then identifying $[Y] \equiv [F]; [W] \equiv [e]$ and $[X] \equiv [I]$. In this case the analogy is trivial: the weight matrix is directly obtained from [*e*] (no training process is required), and, once fixed the measurement points $M+1 \equiv Q+1$, the ANN will respond correctly to the corresponding excitations distributions [I] of a linear array with $N+1 \equiv P+1$ elements, calculating the M+1 values of the array factor F_m at u_m.

---

[1] The training set could be obtained, for example, by some numerical calculation. Usually, if some inverse problem is difficult to solve analytically, but the direct one is known, the input-output vectors pairs (IOVP) are generated directly, and then presented inversely to the ANN. The (IOVP) can be also obtained through physical measurement, if the ANN is required to find some empirical relationship between them.

[2] For the purposes given here, the generalization to more complicated array expressions (near field, non-isotropic elements, planar or conformal arrays, and so on) is straightforward.

2.3.2 ARRAY PATTERN SYNTHESIS AND DIAGNOSTICS.

If M+1=N+1, and [e] is invertible, the problem of finding a certain excitation distribution when field measurements are given is solved directly from (4) as $[\mathrm{I}]=[\mathrm{e}]^{-1}[\mathrm{F}]$ (d, N and λ are considered to be fixed, as before). Nevertheless, it is usual in antenna measurements to have M>N, so (4) becomes an overdetermined system, and its solution can be found by several ways, one of them being by taking the Moore-Penrose pseudoinverse of [*e*], denoted as $[e]^{\Omega}$, (see, for example, [7]). Now, identifying $[\mathrm{Y}]\equiv[\mathrm{I}];[\mathrm{W}]\equiv[\mathrm{e}]^{\Omega}$ and $[\mathrm{X}]\equiv[\mathrm{F}]$, the CVANN can be used to find the corresponding excitation distribution by taking as vector inputs the field measurements, or field calculations. As in 2.3.1, no TP is needed. In this manner, the CVANN will be available for pattern synthesis, by giving certain desired field distribution as input vector to obtain the required excitation distribution, or for array diagnostics, since if the field values that are presented as input vector belong to a certain, previously specified, excitation distribution, the CVANN will be able to find it (or will give an approximate solution, which is optimum in the mean-square-error sense [7]), even if the arrays have some faulty elements (which constitutes another application of this tool). A straight comparison of this CVANN architecture with those given in earlier works for similar purposes (Radial Basis Function architecture in ANNs prepared to estimate the direction of signal arrival, or beamforming for interference cancellation, for example, applied to array antennas [2,3]) reveals the simplification obtained when using CVANN.

3. FINAL REMARKS

It is understood that the analogies given in 2.3.1 and 2.3.2 are readily –and clearly– seen, and it can be thought as the authors are assigning two names to the same thing. In some sense, this is true: once established the equations (2) and (4) –and their counterparts when using the inverse problem established in 2.3.2–, by different ways and with different significances, the direct utility of the CVANN in array antennas design and implementation is encountered rapidly. But,

a) More important than those analogies, is the view that ANNs can be naturally incorporated to array antennas design by allowing them to be complex-valued. In some other antenna problem, in which any of the analogies shown here can not be applied, surely some training process and a different CVANN architecture will be needed. In that case, the ideas expressed in [5] or [6] will probably be of help.

b) One of the advantages of the ANNs is that they can be easily implemented as hardware devices. It is seen that the allowance of making the ANN be complex-valued has no severe modifications in its physical structure, but they will probably be easier to construct due to the simplification of their architectures. Such simplifications will make them very appropriate for real-time operations.

ACKNOWLEDGEMENT

This work has been supported by the Spanish Ministry of Science and Technology, under project TIC2002–04084–C03–02.

REFERENCES

[1] S. Haykin: “Neural Networks: A Comprehensive Foundation”, Pearson Education, Second Edition, 1998.

[2] Ch. Christodoulou, and M. Georgiopoulos, “Applications of Neural Networks in Electromagnetics”, Artech House Publishers, 2000.

[3] H. Southall, and T. O’Donnel, “Direction Finding in Phased Arrays with a Neural Network Beamformer”, IEEE Transactions on Antennas and Propagat., Vol. 43, pp. 1369-1374, 1995.

[4] R. J. Mailloux, and H. L. Southall, “The Analogy Between the Butler Matrix and the Neural Network Direction-Finding Array”, IEEE Antennas and Propagat. Magazine, Vol. 39, pp. 27-32, 1997.

[5] M. S. Kim, and C. C. Guest, “Modification of Backpropagation Networks for Complex-Valued Signal Processing in Frequency Domain”, Proceedings of the International Joint Conference on Neural Networks, Vol. 3, pp. 27-31, 1990.

[6] T. Nitta, “An Extension of the Back-Propagation Algorithm to Complex Numbers”, Neural Networks, Vol. 10, pp. 1391-1415, 2003.

[7] R. Bronson, “Matrix Operations”, McGraw-Hill, 1988.

Figure 1. Structure of a single-layered artificial neural network.