ANTENNAS ARRAY ADJUST WITH ADAPTIVE NEURONAL SYSTEM

A. Padrón, J. I. Garduño, A. A. Herrera & R. Prieto

Laboratorio de Computación Adaptativa, Centro de Ciencias Aplicadas y Desarrollo Tecnológico - UNAM
Apdo. Postal 70 - 186, Coyoacán, 04510, México D.F.
Tel. (52) 5622 8608 al 13, Ext. 106, Fax: (52) 5622 8653
E-mail: apadron@aleph.cinstrum.unam.mx

Received: November 18th, 2002. Accepted: Jaunary 10th, 2003

ABSTRACT

In this work an array failure correction for Linear Antenna Array (LAA) is presented. This is carried out by means of an Adaptive Artificial Neural Network (AANN) that adjusts the amplitude and phase at beamforming. The appropriated corrections are given, when one, or two, or three elements have a failure in the antenna linear array. The AANN corrects the corresponding parameters in the radiation pattern obtained due to the failure, when we know the coefficients of the array factor (AF). This yields a reduction of side lobe level and some interferences disappear.

RESUMEN

En este trabajo se realiza una corrección de fallas para un Arreglo Lineal de Antenas (ALA). Esto se lleva a cabo mediante una Red Neuronal Artificial Adaptable (RNAA) que ajusta al generador del haz "beamforming" en amplitud y fase. Las pertinentes correcciones se dan, cuando en el arreglo lineal de antenas fallan uno, dos y hasta tres elementos. La RNAA corrige los debidos parámetros en el patrón de radiación obtenido para la falla, dados los coeficientes del Factor de Arreglo (AF).

KEYWORDS: Adaptive Artificial Neural Networks (AANN), Array Factor (AF), Side Lobe Level (SLL), Linear Antenna Array (LAA)

1. INTRODUCTION

For a Linear Antenna Array with its traditional beam generator, if one or more elements are damaged by an unfortunate reason, the LAA would stop operating due to an unacceptable distortion on its radiation pattern, for example a significant increment may exist in the side lobes level (SLL). By means of the AANN it is possible not to replace the damaged element, but by recalculating the radiation pattern's parameters with the spare elements to approximate the new pattern to the original one. This capacity to correct the failure in the radiation pattern of the AANN produces a cost-effective alternative to hardware replacement which might be time-consuming, especially for arrays performing critical operations. Other applications include satellite or extraterrestrial communications (earth-air-earth) for example in spaceships, where a damaged antenna element could not be replaced easily by means of substitution [1]. In the literature no analytic technique has been devised to yield a set of new beamforming weights that effectively corrects the damaged pattern. Thus, a failure array LAA can be visualized as an non-uniformly spaced array, the analytic approaches are unable to solve this type of problems. In past years, many algorithms have been proposed to correct the damaged patterns, however, due to the outrage of the geometric distribution of the spare elements in the array and the way of the wanted beamforming, the correction of the failure is still been solved by numerical approaches [2].

The adaptive arrays have been used mainly for the beamforming and to direct the null ones in several civil and military systems, including the GPS, cellular and mobile communications. The main idea in each case is to determine the angle of reception of arrival of the desired signals, as well as the directions of interfering or jamming signals. Once all these signals have been determined, the elements of the adaptive array antenna are excited with the appropriate inputs to place null in the direction of the interfering signals and beams in the desired direction. This is accomplished by finding a cost function that allows maximum gain in the directions of the desired signal and the rejection of the interference by minimizing the antenna power where it is necessary. After mathematical calculations, the cost function can be reformulated so that the optimization can be made by means of Artificial Neuronal Networks with the aim of estimating the appropriated array element weights [3].

The neural generators have the advantage of fast convergence rate that can easily allow the antenna to track the mobile user. Other problems associated with generators with the conventional beamforming stem from the requirements of highly calibrated antennas that assume identical element properties. The degradation often occurs due to the fact that this algorithms have a poor adaptation to the element failure or other sources of errors like interferences. An AANN can be trained with a number of observations of the antenna behavior under a specific set of circumstances. The net can be generalized and then it can be used to predict the aperture behavior at all points.

The LAA can be designed to control its radiation characteristics by selecting the appropriate distribution of phase and amplitude among the elements of the array. It has been proven that the phase control can significantly alter the radiation pattern of a LAA. In fact, the principle of the antennas, where the maximum pattern of the array can be aimed in different directions, is mainly based on the phase control of the elements excitement. Likewise, it has been demonstrated that an appropriate decrease on the excitement of the amplitude among the elements can be used to control the beam width and the side lobes level [4].

The Adaptive Neuronal System carries out the appropriate correction of the beamforming and generates the radiation pattern of a linear array antenna when a failure is presented in one of the elements. This system estimates the proper weights for the new amplitude and phase conditions of the beam generator. These adjustments or corrections can be clearly visualized in figure 1 [5].

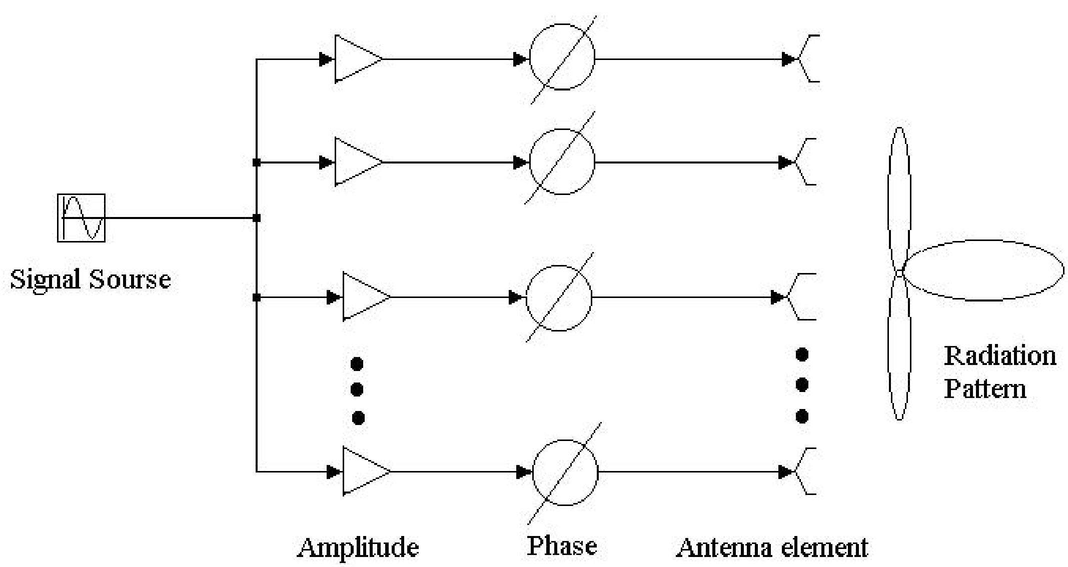

Figure 1. The Beamforming of a linear array antenna

2. PROBLEM FORMULATION

The beamforming of a LAA, can be used only for amplitude, for phase or amplitude-phase approach. In the case of amplitude approach it cannot compensate for the degradation of a damaged array pattern, as the failing elements introduce an asymmetrical aperture distribution. On the other hand, the case of the phase approach with constant amplitude requires a large number of elements to yield low side lobes. Consequently, beamforming using, both amplitude and phase is necessary for the redistribution of the weights, in order to correct the damaged pattern [6].

For an arbitrary arrangement, the Array Factor (AF) can generally be given by:

$$AF = W^T S(\theta,\theta_m) \quad , \tag{1}$$

where

$$w = \{w_1, w_2, \ldots, w_N\}^T,\ w_n \, \varepsilon \, C^n \tag{2}$$

is the weighting vector, S the steering vector, θ y θ_m are the direction variables and main beam direction, respectively. C^n is the set or a subset of all complex numbers.

Take, for example, a linear array of N identical elements, its steering vector is given by:

$$S = \exp\left\{ jkd\left(n - \frac{N-1}{2} \right) . (\cos\theta - \cos\theta_m) \right\} \tag{3}$$

in such a way that the same set of optimum weights for the main beam at broadside can be used for other directions, when the above S vector is recalculated for the new beam-pointing direction.

When the Mth-element fails in the array, its weight w_m is supposed to be similar to zero. Thereafter, the AANN is applied to correct the SLL and the main beam shape of the pattern taking in to account the previously specified failures.

3. ADAPTATIVE ARTIFICIAL NEURAL NETWORK

The networks with adaptive linear neurons (ADALINE) are similar to the perceptron, but their activation function is linear compared with the step or "hard limiter." This allows their outputs to take any value, whereas the perceptron output can only take values between 0 and 1. Both, the ADALINE and the perceptron can only solve linearly separable problems. However, making use of the least mean square (LMS) learning rule, which is much more powerful than the perceptron learning rule, it is possible to minimize the mean square error and thus to move the decisions boundaries as far as it can from the training patterns [7].

In an adaptive neural network when a set of input vectors is presented, produces outputs of corresponding objective or target vectors. For each input vector we can calculate the network output vector. The difference between an output vector and its target vector is the error. We would like to find values for the networks weights and bias such that the sum of the squares of the errors is minimized or below a specific value. This problem is manageable because linear systems have a single error minimum. In most cases we can calculate a linear network directly, such that its error is minimum for the given input vectors and target vectors. In other cases the numeric problems prohibit direct calculation. Fortunately, we can always train the network to have a minimum error using the Widrow-Hoff learning rule.

With this we can design an adaptive linear system that responds to changes in its environment as it is operating. The linear networks which are adjusted at each time step based on new input and target vectors can find weights and appropriate bias which minimize the network and the sum squared error for recent input and target vectors. The networks of this type are often used in problems of error cancellation, signal processing and control systems.

Adaptive filtering is just one of the many more applications areas where the ADALINE has been used. To carry out this kind of application is necessary a new component to make complete use of the power of an adaptive network, this is a delay line. In figure 2 is shown the neuron pattern like adaptive filter that was used in this paper. The pattern has input signal from the left, and each one of them passes through N-1 delays. The output of the tapped delay line (TDL) is an N-dimensional vector, made up of the input signal at the current time, the previous input signal, etc. These in turn pass through a linear activation function to obtain their output.

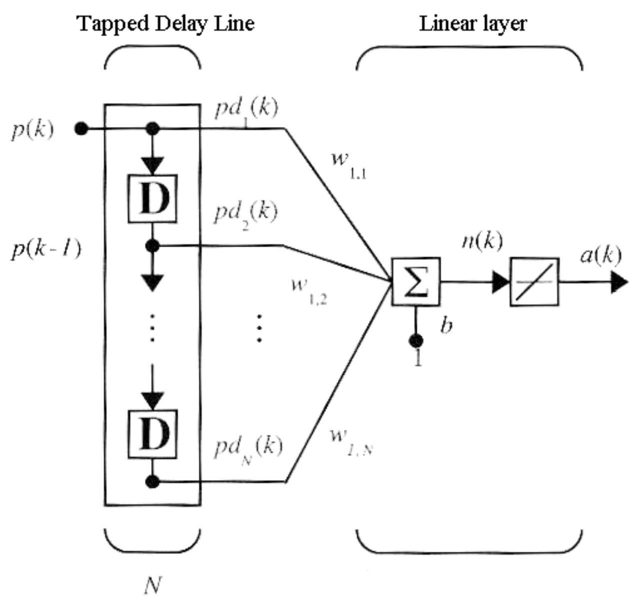

Figure 2. Neuron model like an adaptive filter

4. PROCEDURE

When a failure occurs in a 16 elements linear adaptive array (LAA), the steering vector S is affected and therefore this damages the radiation pattern in the coefficients of the array factor *(AF)*. Then is supposed a zero in the fail element, with this a new radiation pattern is obtained for the M-1 remaining array elements and the AANN adjusts the new parameters to adapt them to the original radiation pattern. Each radiation pattern is formed by 901 points that correspond to its amplitude and its phase for 180º, with a step of 0.2. For AANN to carry out its work, we made a preprocess of the original data, improving its functionality, moving all the points to positive values and then normalizing them in an interval of [-1,1]. To train the AANN we used the following code in MatLab:

First we read the data of a file, as much for the inputs as for the targets vectors:

```
'P = dlmread ("16e35db.dat","\t",0,0,[0,0,901,0]);',
'T = dlmread ("16e35db.dat","\t",0,1);',
```

Then the vectors are normalized in an interval of [-1,1]:

```
'[Pn,minp,maxp] = premnmx(P);',
'[Tn,mint,maxt] = premnmx(T);',
```

Thus they are linked and transpose (they make the data confortable for the AANN)

'P = con2seq(Pn.'');'
'T = con2seq(Tn.'');'

The linear network must have tapped delay in order to learn the time-shifted correlation between P (input vector) and T (target vector), NEWLIN creates a linear layer.',

'net = newlin([-1 1],1,[0 1],0.5);'

NEWLINE: has as parameters, ([-1 is the minimum value of input, 1 is the maximum value of the input (is the expected input range)], The second argument 1 is the number of neurons in the layer, [0 1] specifies one input with no delay and one input with a delay of 1, The last argument 0.5 is the learning rate);

'[net,Y,E,Pf]=adapt(net,P,T);'

ADAPT: simulates adaptive networks. It takes a network, an input signal, and a target signal, and filters the signal adaptively. By t=2 the network has learned the relationship between the input and the target and the error drops to near zero.

The process mentioned above repeats when there are two and three fails in the elements of LAA. The coefficients of the AF are presented for the remaining n-elements to generate the patterns with damage. These radiation patterns are shown in the figure 3, then the radiation patterns adjustment is presented together with the error for each array [8].

5. SIMULATIONS RESULTS

A 16-elements Linear Antennas Array (LAA) is used with Dolph-Tschebyscheff configuration, and separation of 0.5[λ] (wavelength) with a non-uniform spacing, a Side Lobe Level (SLL) of -35 dB is used as a reference, and with the main beam pointing directed toward 90°.

The damage radiation patterns are obtained when one or more elements of ALA stop working by unforeseen reason, its correspond excitation coefficient in the array factor (AF) turns to zero. Then the adaptive artificial neural network (AANN) adjusts the parameters from these radiation patterns to the original parameters of 16-elements array. In figure 3 the radiation pattern is shown for 16 elements of LAA. The excitation coefficients of the AF for 16-elements are:

Excitation Coefficients for 16-elements LAA

2.078	2.827	4.512	6.316	8.130	9.744	10.959	11.605
11.605	10.959	9.744	8.130	6.316	4.512	2.827	2.078

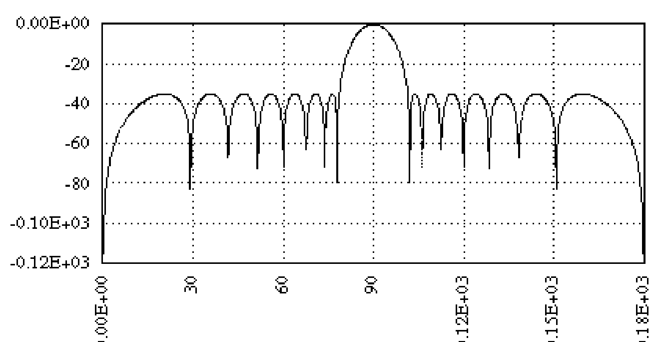

Figure 3. Radiation pattern for 16-elements of LAA

Characteristics of the output for 16-elements of the array:

DIRECTIVITY = 11.120 [dB]
DIRECTIVITY = 12.943 dimensionless
NUMBER OF MAXIMA BETWEEN 0 AND 180 DEGREES = 1
Half-Power Beam Width for maximum element = 8.5° for θ_{MAX} = 90.0°

Following, when there is a failure in the second, fourth and sixth element of the linear antennas array the radiation patterns are shown.

Failure of 1 – element

Excitation Coefficients for 16-elements LAA with 1 failure

2.359	0.000	6.232	8.974	11.703	14.033	15.603	8.078
8.078	15.603	14.033	11.703	8.974	6.232	3.812	2.359

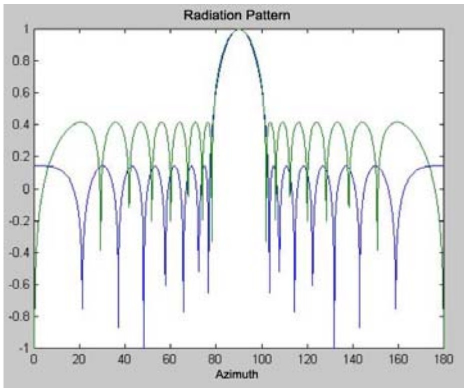

Figure 4. Radiation Pattern of 16-e (green) and radiation pattern with 1 failure (blue)

Characteristics of the output for 16-elements with 1 failure:

DIRECTIVITY = 10.730 [dB]
DIRECTIVITY= 11.830 dimensionless
NUMBER OF MAXIMA BETWEEN 0 AND 180 DEGREES = 1
Half-Power Beam Width for maximum element = 9.3° for θ_{MAX} = 90.0°

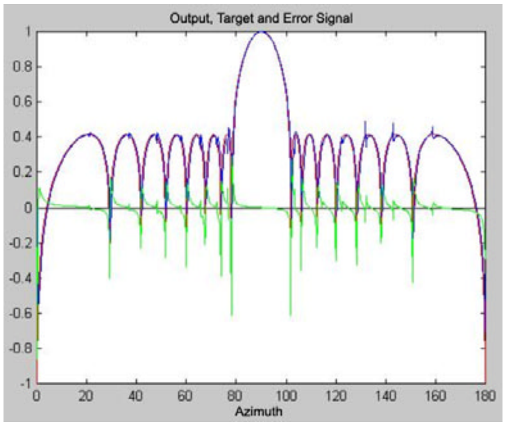

Figure 5. Radiation pattern of 16-e (red), radiation pattern with adjustment (blue) and error (green)

Failure of 2 – elements

Excitation Coefficients for 16-elements LAA with 2 failures							
2.622	0.000	4.697	0.000	11.593	15.134	17.947	19.506
19.506	17.947	15.134	11.593	7.933	4.697	2.827	2.622

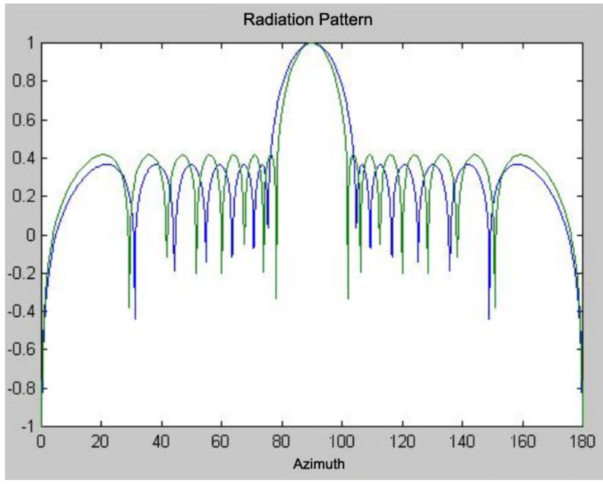

Figure 6. Radiation Pattern of 16-e (green) and radiation pattern with 2 failures (blue)

Characteristics of the output for 16-elements with 2 failures:

DIRECTIVITY = 10.374 [dB]
DIRECTIVITY = 10.898 dimensionless
NUMBER OF MAXIMA BETWEEN 0 AND 180 DEGREES = 1
Half-Power Beam Width for maximum element = 10.1° for θ_{MAX} = 90.0°

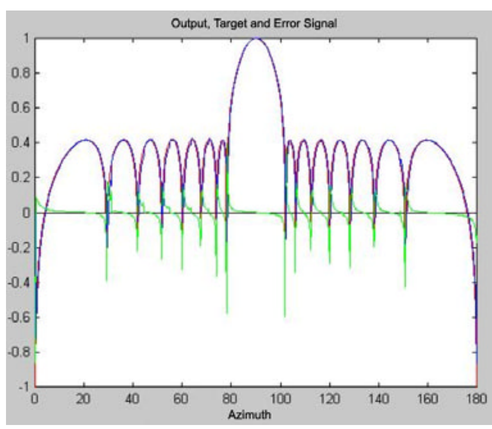

Figure 7. Radiation pattern of 16-e (red), radiation pattern with adjustment (blue) and error (green)

Failure of 3 – elements

Excitation Coefficients for 16-elements LAA with 3 failures							
2.963	0.000	5.888	0.000	15.195	0.000	22.980	12.074
12.074	22.980	19.750	15.195	10.275	5.888	4.512	2.963

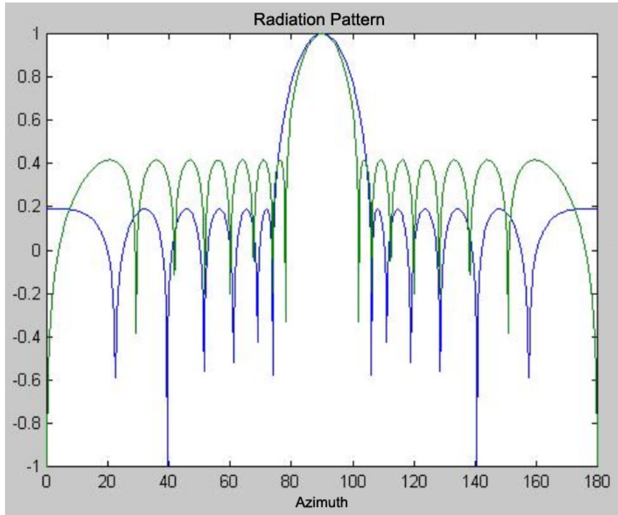

Figure 8. Radiation Pattern of 16-e (green) and radiation pattern with 3 failures (blue)

Characteristics of the output for 16-elements with 3 failures:

DIRECTIVITY = 9.997 [dB]
DIRECTIVITY = 9.994 dimensionless
NUMBER OF MAXIMA BETWEEN 0 AND 180 DEGREES = 1
Half-Power Beam Width for maximum elements = 11° for θ_{MAX} = 90.0°

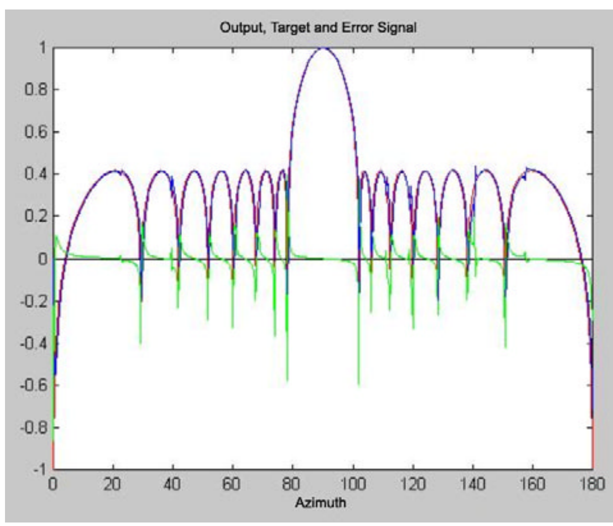

Figure 9. Radiation pattern of 16-e (red), radiation pattern with adjustment (blue) and error (green)

6. CONCLUSIONS

The radiation patterns correction that carries out the AANN (Adaptive Artificial Neural Network), when failures are presented in the linear array of antennas of 16-elements, are below 1% in the softest areas, and it is increased in the abrupt changes of the original radiation patterns. In some cases variations are presented in the adjustment (figure 5), but it is necessary to remember that during the training the learning rate of the AANN is very important in the adaptation of both patterns, the original and failure one.

The adjustment of the radiation patterns depends on excitation coefficients, because these coefficients support to generate the radiation pattern. However in the AANN the weights are parameters of correction which takes the damaged signal to the original, to reduce the level of the side lobes level (SLL). Thus the weights and biases of the network are the ones that adapt and correct the failures of damaged radiation pattern. These obtained AANN parameters are related with the excitation coefficients of the array and with the beamforming which yield the amplitude and the phase of the array.

The success correcting a damaged pattern depends strongly on the original weighting of the failed elements and of the number of failures in the array. In this instance, if in the linear antennas array (LAA) the eighth or ninth element fails resulting in a blockage, it would be impossible to correct or yield improvement using AANN or using any technique. We can observe that the failures occur on the same side with respect to the central element, but nevertheless could happen in both sides of the array also.

7. REFERENCES

[1] Neri-Vela R., Análisis de antenas lineales por el método de momentos, Ingeniería Investigación y Tecnología, I.2 UNAM, 1999. 73-83.

[2] Balanis C. A., Antenna Theory analysis and design, New York: John Wiley & Sons, Inc., 1997.

[3] Garduño J. I. & Padrón A. Análisis de Antenas Lineales Empleando Redes Neuronales Artificiales. X Congreso Internacional de Electrónica, Comunicaciones y Computadoras, CONIELECOMP-2000, Cholula-Puebla, México, 2000, 393-397.

[4]. Garduño J. I & Padrón A., Aproximación de la distribución de corrientes para una Antena Yagi-Uda empleando Redes Neuronales Artificiales, in Sosa J. H., De Albornoz A., Kuri A. F. (Eds.), Avances en Inteligencia Artificial, MICAI/TAINA/TIARP 2000, México, D. F., SMIA, 2000, 21-30.

[5] Ramos J. L. et. al., Arreglos de antenas con separación no uniforme, Congreso de Instrumentación SOMI XII, San Luis Potosí, S.L.P., México, 1997, 748-754.

[6] Beng-Kiong Y. & Yilong Lu, Array Failure Correction with a Genetic Algorithm, IEEE Transactions on Antennas and Propagation, Vol. 47, No. 5, 1999.

[7] Demuth H. & Beale M., Neural Networks ToolBox, MatLab, USA: The Math Works, Inc., 1998.

[8] Padrón A., Garduño J.I., Herrera A., Prieto R., Current Distribution Interpolation for Antennas Linear Arrays with Nonuniform Spacing. Instrumentation & Development, Journal of the Mexican Society of Instrumentation, Vol. 5, No. 2, 2001.

Authors Biography

Alejandro Padrón

Was born in Mexico City, in 1965. He is a physicist graduated in 1992 from *Facultad de Ciencias at Universidad Nacional Autonoma de Mexico*, Mexico City, and got a Master degree in Electrical Engineering at *Division de Estudios de Posgrado de la Facultad de Ingenieria*, 1998, UNAM-Mexico. He is a Doctoral Student in Electrical Engineering in Graduated Program at the same University. He is currently Professor at *Facultad de Ingenieria*, UNAM, member of the *Sociedad Mexicana de Instrumentacion* and Academic Technician of *Laboratorio de Computacion Adaptable at Centro de Ciencias Aplicadas y Desarrollo Tecnologico*, UNAM-Mexico. His research interests include software simulation systems in severals processes with soft computing technics.

Jeny Ivette Garduño

Was born in Mexico City, in 1978. She graduated in Telecommunication Engineering at *Universidad Nacional Autonoma de Mexico*, Mexico City, 2001. She is a student of the Master Program in Telecommunications Administration at *Tecnologico de Monterrey Campus Ciudad de Mexico*, 2002. Her research interests include wireless communications and cellular telephony. She is currently the Head of Administration, Control and Networks Monitoring Department at *Secretaria de Gobernacion*, Mexico City.

Rael Prieto

Was born in Mexico City in 1969. He graduated in Mechanical-Electrical Engineering at Engineering School in National Autonomous University of Mexico (1993) and received the Master Degree in Engineering at Postgraduate Studies Division of Engineering School in National Autonomous University of Mexico (1999). He is currently working in the Adaptive Computing Laboratory of the Center of Applied Sciences and Technological Development of the National Autonomous University of Mexico. His research interest include artificial neural networks (neural oscillators, neural classifiers and discrete neural models), electronic implementation of neural models, and modeling and simulation of processes.

Alberto Arturo Herrera

Was born in Mexico City in 1961. He is a Chemist graduated in 1980 from the National Autonomous University of Mexico (UNAM). At this University he also received a Master Degree in Biomathematics in 1985. Nowadays, he is a Technological Researcher at the Center of Applied Sciences and Technological Development of UNAM, where he is in charge of the Adaptative Computation Laboratory. His research interests include the design, construction and implementation of simulation systems based on soft computation techniques.