PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Synthesis of electromagnetic devices with a novel neural network

Heriberto Jose Delgado
Michael H. Thursby
Fredric M. Ham

Synthesis of Electromagnetic Devices with a Novel Neural Network

Heriberto J. Delgado[a], Michael H. Thursby[b] and Fredric M. Ham[c]

[a]Harris Corporation, Melbourne, FL 32902

[b]Command Technologies Inc., Rockledge, FL 32955

[c]Florida Institute of Technology, Melbourne, FL 32901

ABSTRACT

A novel Artificial Neural Network (ANN) is presented, which has been designed for computationally intensive problems, and applied to the optimization of electromagnetic devices such as antennas and microwave devices. The ANN exploits a unique number representation in conjunction with a more standard neural network architecture. An ANN consisting of a hetero-associative memory provided a very efficient method of computing the necessary geometrical values for the devices, when used in conjunction with a new randomization process. The number representation used provides significant insight into this new method of fault-tolerant computing. Further work is needed to evaluate the potential of this new paradigm.

Keywords: Antenna synthesis, neural network, randomization processes, group and weight scheme number representation, simple sum scheme number representation, Finite Difference Time Domain Method (FD-TD), microwave device optimization, microwave device synthesis, computationally intensive problems.

1. INTRODUCTION

Traditional techniques used for the synthesis of antennas and microwave devices often require numerous lengthy computer runs using a numerical electromagnetic model. Techniques have included gradient descent algorithms, back-propagation neural networks[1-4], genetic algorithms[5-7], multiple neural networks[8], and evolutionary neural networks[9], to name a few. In addition, electromagnetic artificial neural networks (EM-ANN)[10-11] and knowledge-oriented neural networks[12-14] have been used in the design and optimization of microwave devices. EM-ANNs map inputs to outputs and knowledge-oriented neural networks embed equivalent circuit models into the neural network topology.

The new synthesis neural network presented[15], named SYNTHESIS-ANN, finds solution subspaces by first mapping inputs to outputs during supervised training of a hetero-associative memory. The number representation consisted of a simple sum bipolar scheme, which is fault tolerant and allows for finding more than one solution to the problem. Inputs and outputs are randomized during each synthesis iteration. An analysis tool is used to find corresponding inputs to the generated randomized outputs. These new input/output pairs are then used to re-train the mapping neural network, and the process continues until a solution is found. Examples used to verify the usefulness of this new neural network consist of a microstrip line and a printed dipole antenna with an integrated balun[16].

2. DESCRIPTION OF THE NEURAL NETWORK

SYNTHESIS-ANN is a new type of neural network that uses a hetero-associative memory and extends a subspace containing training data to solution subspaces containing data pairs by random variations of inputs and outputs, between lower and upper bounds, as shown in Fig. 1. The fundamental assumption is that there exists a solution subspace in the

Independent Component Analyses, Wavelets, Unsupervised Smart Sensors, and Neural Networks II, edited by Harold H. Szu, et al., Proc. of SPIE Vol. 5439 (SPIE, Bellingham, WA, 2004) 0277-786X/04/$15 · doi: 10.1117/12.549466

vector space defined by a problem containing all possible combinations of inputs and outputs. The vector space is limited by physical constraints of a particular problem. Initially, the ANN produces the best possible solution in the training subspace. Randomization of outputs and the computation of corresponding inputs with an analysis tool allows for the generation of data sets in subspaces outside the original training subspace, and closer to a solution subspace. If a solution subspace is not found, changes can be made to the neural network while still modeling an equivalent problem.

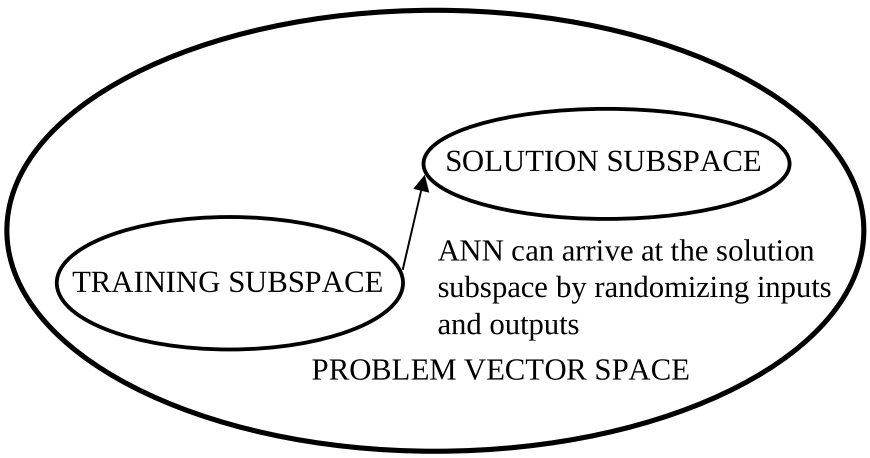

Fig. 1. Diagram illustrating mathematical foundations for SYNTHESIS-ANN using vector spaces and subspaces

The primary function of the neural network is to map inputs to outputs during each synthesis iteration. The inputs and outputs are interconnected by weights with one hidden layer, forming a hetero-associative memory with weights updated in the direction of the steepest descent. The activation function is linear, and learning rates for each weight are adaptive using the Delta-Bar-Delta algorithm[3]. Convergence criteria include minimization of the difference between actual and desired electrical performance. During training, the criteria used for convergence is

$$\mathrm{ERROR} = \sqrt{\sum_j \left[y_j - t_j \right]^2} \le 0.1, \tag{1}$$

where y_j is the output vector and t_j is the target vector. During synthesis iterations, the convergence condition is,

$$\mathrm{BIP\_ERROR} = \sqrt{\sum_j \left[g\left(y_j\right) - t_j \right]^2} = 0, \tag{2}$$

which provides much faster convergence, where g(y) is a step activation function with a zero threshold, given by

$$g\left(y\right) = \begin{cases} +1 \quad \text{if} \quad y > 0 \\ +0 \quad \text{if} \quad y = 0 \\ -1 \quad \text{if} \quad y < 0 \end{cases}. \tag{3}$$

The training data is generated with an analysis tool. At least one input/output pair is needed. The analysis tool can consist of a mathematical model, which computes inputs from generated outputs. An analysis tool may be computationally efficient closed form set of equations or computationally intensive simulations using FD-TD[17], Finite Elements[18] and Method of Moments[19]. Other analysis tools can even consist of physical measurements. Inputs and outputs are represented using a bipolar number representation with a simple sum scheme[1,20], which is not efficient in terms of storage. The number representation is fault tolerant, a property that speeds up the convergence process and can provide creative solutions because more than one set of weights can produce the desired solution. For example, the real number 2.3 is converted to a simple sum binary representation as follows. Integer 2 is represented by the sum of the first four digits and integer number 3 is represented by the sum of the last four digits, that is,

$$2.3 = \sum_{i=1}^{4} V_i + \frac{1}{10}\sum_{i=5}^{8} V_i \; . \tag{4}$$

For example,

$$2.3 = \begin{bmatrix}1 & 1 & 0 & 0 & 0 & 1 & 1 & 1\end{bmatrix} \tag{5}$$

is a valid simple sum number representation for 2.3. The binary simple sum number is converted to bipolar bit-by-bit, for use by the neural network. The flowchart of the SYNTHESIS-ANN is presented in Fig. 2, and the steps associated with a synthesis iteration are shown next.

1) Specify desired electrical parameters values (real numbers).
2) Randomize electrical parameters between lower and upper bounds.
3) Convert parameters from step 2 from real to bipolar using the simple sum scheme.
4) Run mapping neural network and generate geometrical parameters (bipolar numbers).
5) Convert geometrical parameters to a real number representation.
6) Generate outputs by randomizing geometrical parameters (real numbers) between lower and upper bounds.
7) Using the analysis tool, compute inputs associated with outputs from step 6.
8) Check for convergence. Input parameters found in step 7 must be equal or better than specified electrical performance. If convergence is achieved, stop the process; otherwise, go to step 9.
9) Compute the bipolar number representation for real inputs from step 7 and real outputs from step 6.
10) Re-train the mapping neural network using new input/output pair found in step 9.
11) Update the weight inter-connections of mapping neural network.
12) Go to step 2.

3. EXAMPLE PROBLEMS

Problems used for testing the SYNTHESIS-ANN include a microstrip line and a printed dipole antenna with integrated balun illustrated in Sections 3.1 and 3.2 respectively.

3.1. MICROSTRIP LINE

The synthesis of the line width W for an infinitely long microstrip line is performed. The geometry is shown in Fig. 3. The analysis tool uses closed form equations[21], where given the line width W, dielectric substrate thickness h and dielectric constant ε_r, the line impedance is found. The input is the microstrip line impedance in ohms, and the output is the line width W in mils. These quantities are represented by 520 bits, where the first 500 bits represent the integer part, and the last 20 bits represent the decimal portion. A total of 23 input/output pairs are used for training. The number of epochs needed for convergence during training is 333. The synthesis goal is to find the width W for a 50-ohm microstrip line. Inputs are randomized between 49 ohms and 51 ohms, and outputs are randomized between the width W plus or minus 5 mils. A total of 10 synthesis iterations are needed. Error did not decrease continuously with iteration due to the randomization feature. A line impedance of 50.1 ohms is obtained, and the corresponding output parameter is a line width W of 91.6 mils.

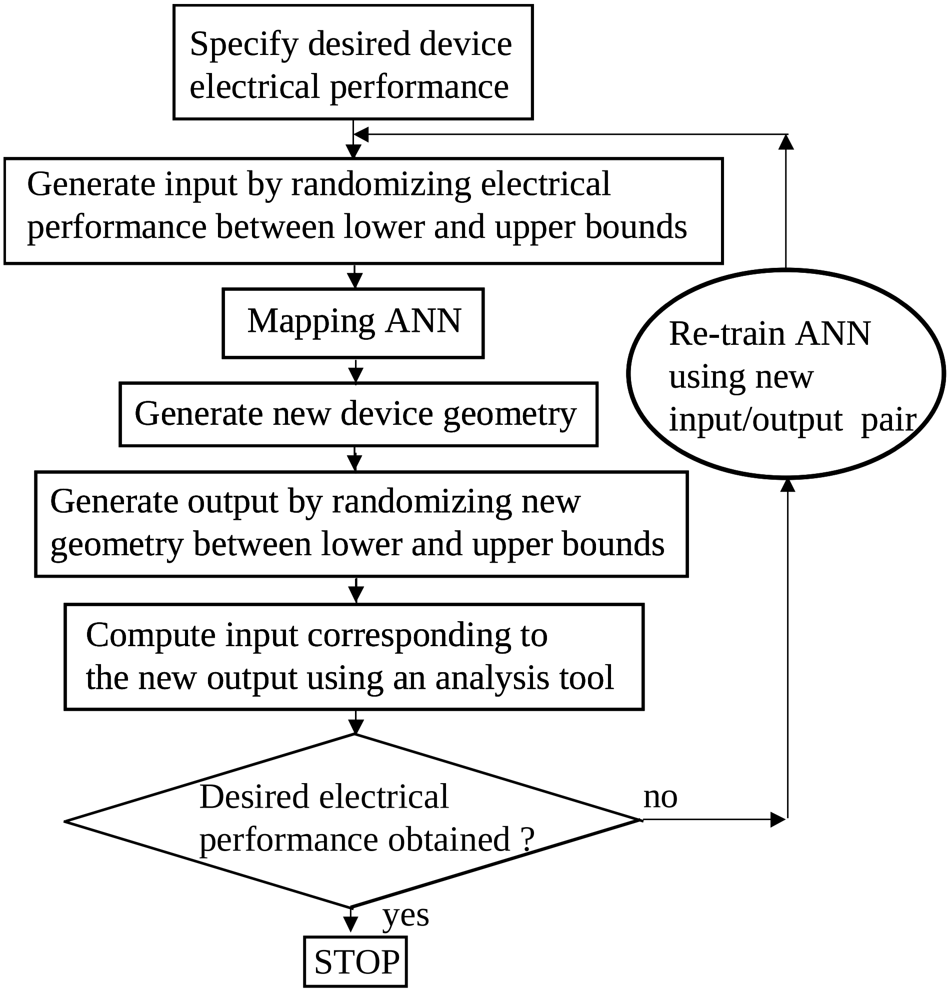

Fig. 2. Flowchart of neural synthesis process

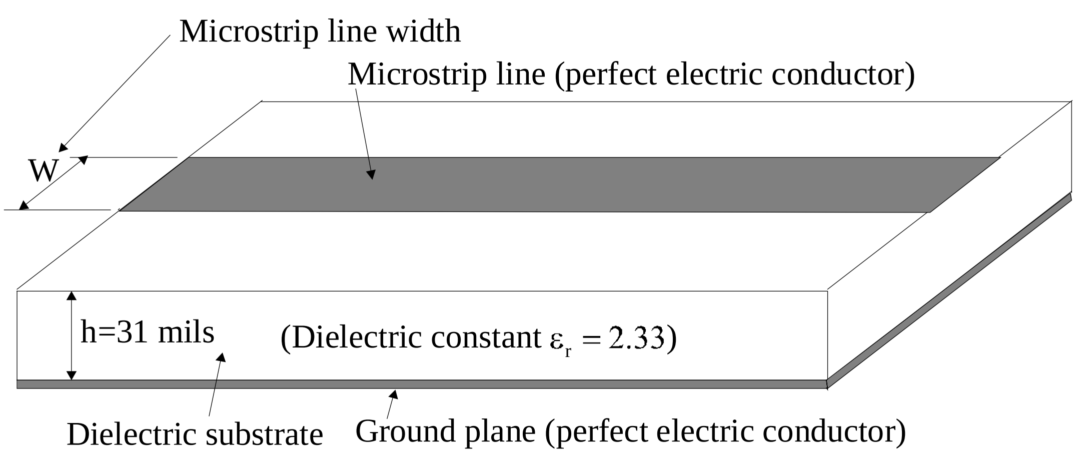

Fig. 3. Microstrip line geometry

3.2. PRINTED DIPOLE ANTENNA WITH INTEGRATED BALUN

The dipole geometry is shown in Fig. 4. Dimensions shown are used as outputs. The microstrip feed line has a constant width throughout, it is attached to the inner conductor of an SMA connector and is centered between two dipole arms located on the other side of dielectric substrate, where the printed dipole with integrated balun and ground plane are placed. The outer conductor of SMA connector is attached to a ground plane.

The analysis tool used is an FD-TD simulation[15], used to compute Voltage Standing Wave Ratio (VSWR) over a frequency band from 8 GHz to 12 GHz. Inputs are five electrical parameters consisting of three VSWR numbers and two frequencies, each with two decimal places. VSWR numbers are minimum VSWR (X1), average VSWR (X2) and maximum VSWR (X3). Frequency inputs are frequencies at which minimum and maximum VSWRs occur, and are represented by X4 and X5. VSWR numbers are represented with 60 bits and frequency numbers with 80 bits, for a total of 340 bits. Outputs are eight integer geometrical parameters, each represented with 600 bits resulting in a total of 4800 bits.

A total of 70 training data sets are used and 6067 epochs are needed. During synthesis, inputs X1 through X5 are randomized as follows: $1.00 \le X1 \le 1.25$, $1.25 \le X2 \le 1.50$, $1.50 \le X3 \le 2.00$, $8.00 \le X4 \le 12.00$. When $10.00 \le X4 \le 12.00$, X5 is randomized with bounds $8.00 \le X5 \le 10.00$. When $8.00 \le X4 \le 10.00$, X5 is randomized with bounds $10.00 \le X5 \le 12.00$. Outputs are randomized so their value would remain unchanged, increase, or decrease by 5 mils.

Only 3 synthesis iterations are required to arrive at a VSWR of 2.09. Baseline and optimized inputs are shown in Table 1. Note that quantity X3, the maximum VSWR, improved from 3.18 to 2.09. A plot of VSWR versus frequency for baseline and optimized cases is shown in Fig. 5. Note the significant improvement.

The baseline and optimized geometrical parameters are shown in Table 2. Geometries corresponding to baseline and optimized cases are shown in Fig. 6. Note that width of the microstrip feed line decreased from 45 to 40 mils, and the thickness of dipole arms increased from 45 to 70 mils.

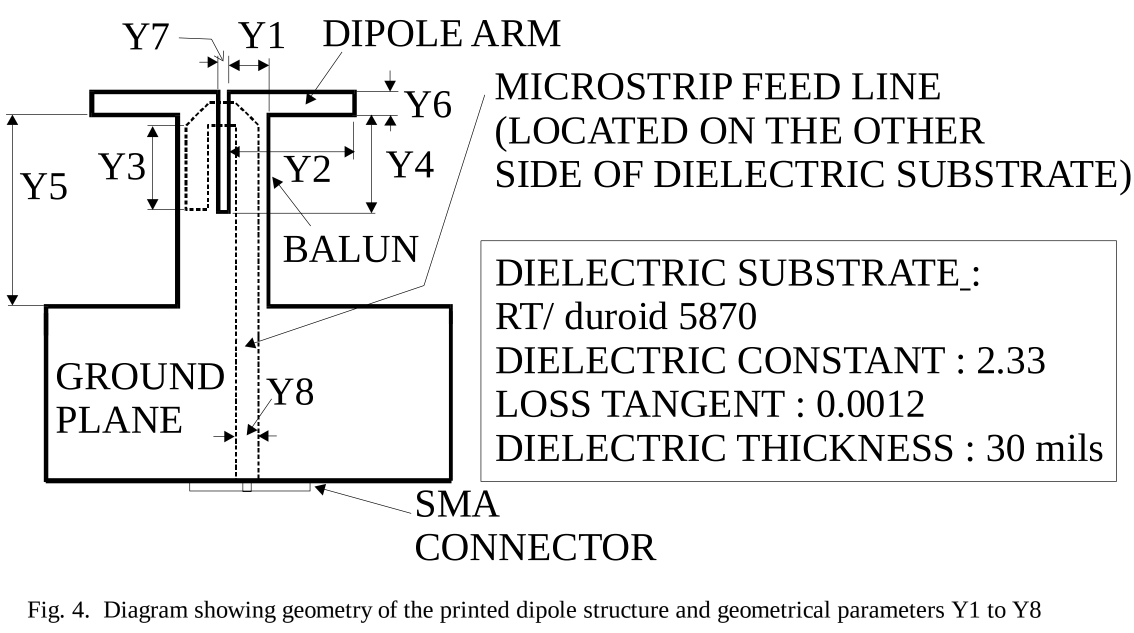

Fig. 4. Diagram showing geometry of the printed dipole structure and geometrical parameters Y1 to Y8

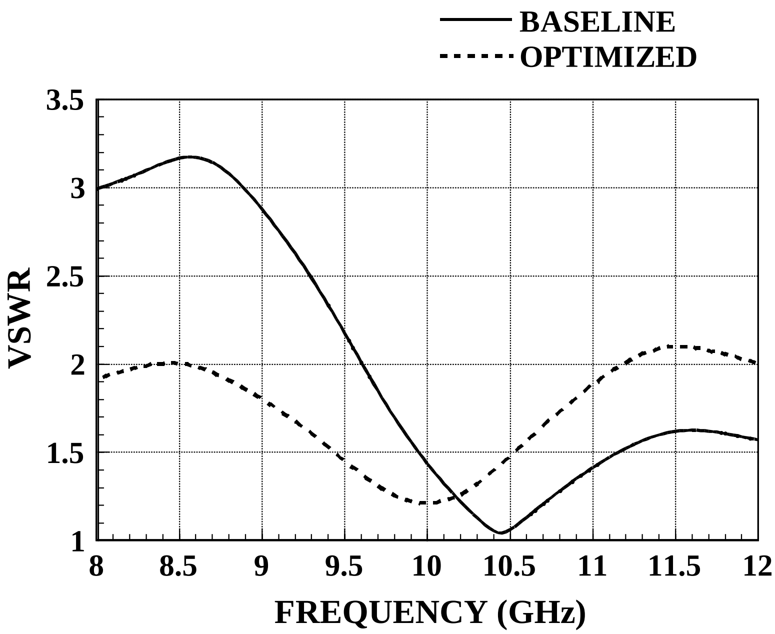

Fig. 5. VSWR versus frequency for baseline and optimized geometries

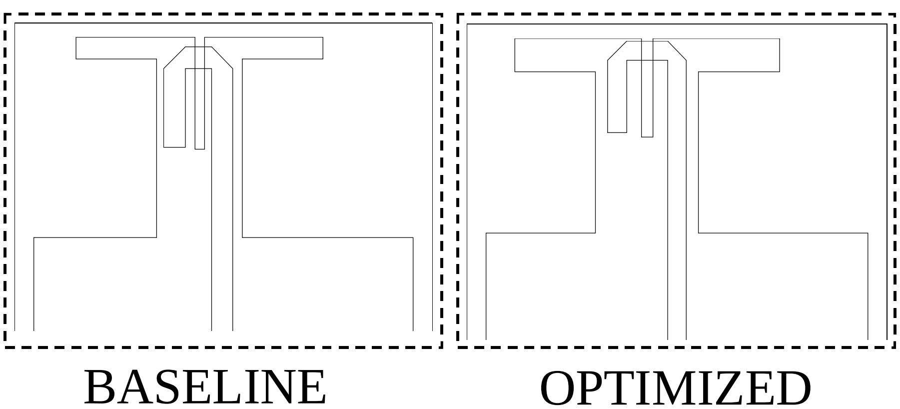

Fig. 6. Scaled geometry corresponding to baseline and optimized cases

Geometry	**X1**	**X2**	**X3**	**X4**	**X5**
Baseline	1.05	2.01	3.18	10.43	8.57
Optimized	1.21	1.74	2.09	10.00	11.58

Table 1. Inputs corresponding to baseline and optimized geometries

Geometry	**Y1**	**Y2**	**Y3**	**Y4**	**Y5**	**Y6**	**Y7**	**Y8**
Baseline	80	250	165	190	375	45	20	45
Optimized	95	265	150	135	335	70	25	40

Table 2. Outputs showing geometrical parameters corresponding to Fig. 5 and Fig. 6

4. CONCLUSIONS

The SYNTHESIS-ANN utilizes existing training data sets, no matter how small, and automatically retrains itself in the search for solution subspaces. This is done by randomizing inputs and outputs, while using traditional neural network topologies for mapping inputs to outputs. This minimizes computational requirements by reducing the number of times an analysis tool is used. When the SYNTHESIS-ANN fails to find a solution subspace, after several iterations, the description of the problem geometry can be varied. For instance, inputs or outputs can be added, or a different number representation can be used. A fault tolerant number representation consisting of the simple sum scheme is successfully used that allows fast convergence and creative solutions.

The SYNTHESIS-ANN is tested with two examples, consisting of a microstrip line and a printed dipole antenna with integrated balun, for which the analysis tool consisted of a computational intensive FD-TD model. Inputs consisted of desired electrical parameters and outputs consisted of resulting geometrical parameters.

Future work can consists of the application of SYNTHESIS-ANN to other types of antennas and microwave circuits. In addition, further work in the number representation of inputs and outputs can be very fruitful. There remains work to be done in the development of numerical methods for computing lower and upper bounds during random variations, utilizing the neural network error function. SYNTHESIS-ANN provides a new bridge into the application of neural networks to the solution of synthesis problems in electromagnetics, and other applicable areas limited only by our imagination.

REFERENCES

1. C. Christodoulou and M. Georgiopoulos, *Applications of Neural Networks in Electromagnetics*, Artech House, Inc., 2002.
2. A. Linden, and J. Kindermann, "Inversion of Multilayer Nets," *International Joint Conference on Neural Networks*, Washington D. C., pp. II425-II430, July 1989.
3. L. Fausett, *Fundamentals of Neural Networks Architectures, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, NJ 07632, 1994.
4. F. M. Ham and I. Kostanic, *Principles of Neurocomputing for Science and Engineering*, McGraw-Hill, New York, 2001.
5. E. Michielssen, J. M. Sajer, and R. Mittra, "Design of Multilayered FSS and Waveguide Filters Using Genetic Algorithms," *IEEE Antennas and Propagation AP-S International Symposium*, vol. 3, pp. 1936-1939, June 28-July 2, 1993.
6. D. E. Goldberg, *Genetic algorithms in search, optimization and machine learning*, Addison Wesley, MA, October, 1989.
7. D. Eclercy, A. Reineix and B. Jecko, "FDTD genetic algorithm for antenna optimization," *Microwave and Optical Technology Letters*, vol. 16, no. 2, pp 72-74, John Wiley & Sons Inc, New York, October 5, 1997.
8. G. Fedi, S. Manetti, G. Pelosi and S. Selleri, "Profiled Corrugated Circular Horns Analysis and Synthesis Via an Artificial Neural Network," *IEEE Transactions on Antennas and Propagation*, vol. 49, no. 11, November 2001.
9. D. Wicker, M. M. Rizki, and L. A. Tamburino, "E-Net: Evolutionary neural network synthesis," *Neurocomputing*, vol. 42, pp. 171-196, 2002.
10. Q. J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design*, Artech House Inc., 685 Canton Street, Norwood, MA 02062, 2000.
11. Watson, P. M. and K. C. Gupta, "Design and Optimization of CPW Circuits Using EM-ANN Models for CPW Components," *IEEE Trans. Microwave Theory and Techniques*, vol. 45, p. 2515-2523, Dec. 1997.
12. F. Wang, and Q. J. Zhang, "Knowledge Based Neural Models for Microwave Design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, 1997, pp. 2333-2343.
13. Zhang, Q. J., F. Wang, and V. K. Devabhaktuni, "Neural Network Structures for RF and Microwave Applications," *IEEE AP-S Antennas and Propagation Int. Symp.*, Orlando, FL, pp. 2576-2579, July 1999.
14. Wang., F., et al., "Neural Network Structures and Training Algorithms for RF and Microwave Applications," *Int. Journal of RF and Microwave CAE, Special Issue on Applications of ANN to RF and Microwave Design*, vol. 9, pp. 216-240, 1999.

15. H. J. Delgado, "Automated Antenna Synthesis Using Neural Networks and the Finite Difference Time Domain," Ph.D. dissertation, *Florida Institute of Technology*, May, 2003.
16. B. Edward and D. Rees, "A Broadband Printed Dipole With Integrated Balun," *Microwave Journal*, May 1987.
17. A. Taflove, *Computational Electrodynamics, The Finite-Difference Time-Domain Method*, Artech House, Inc., MA, 1995.
18. M. S.-Palma, T. K. Sarkar, L.-E. García Castillo, Tanmoy Roy, Antonije Djordjević, *Iterative and Self-Adaptive Finite-Elements in Electromagnetic Modeling*, Artech House, Inc., 1998.
19. R. F. Harrington, *Field Computation by Moment Methods*, IEEE Press, 1993.
20. M. Takeda, and J. Goodman, "Neural Networks for Computation: Number Representations and Programming Complexity," *Applied Optics*, vol. 25, no. 18, pp. 3033-3046, 1986.
21. K. C. Gupta, Ramesh Garg, Rakesh Chadha, *Computer-Aided Design of Microwave Circuits*, Artech House, 1981.