

 Open access • Journal Article • DOI:10.1002/1099-047X(200101)11:1<4::AID-MMCE2>3.0.CO;2-I

## Neural Networks for Microwave Modeling: Model Development Issues and Nonlinear Modeling Techniques — [Source link](#)

V.K. Devabhaktuni, Mustapha C. E. Yagoub, Yonghua Fang, Jianjun Xu ...+1 more authors

**Institutions:** Carleton University

**Published on:** 01 Jan 2001 - International Journal of Rf and Microwave Computer-aided Engineering (John Wiley & Sons, Inc.)

**Topics:** Neural modeling fields, Recurrent neural network, Types of artificial neural networks, Artificial neural network and Stochastic neural network

Related papers:

- [Neural Networks for RF and Microwave Design](#)
- [Artificial neural networks for RF and microwave design - from theory to practice](#)
- [Artificial neural networks for fast and accurate EM-CAD of microwave circuits](#)
- [Neural Networks: A Comprehensive Foundation](#)
- [Design and optimization of CPW circuits using EM-ANN models for CPW components](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/neural-networks-for-microwave-modeling-model-development-4hl2k2hhue>

# Neural Networks for Microwave Modeling: Model Development Issues and Nonlinear Modeling Techniques

Vijaya Kumar Devabhaktuni, Mustapha C. E. Yagoub, Yonghua Fang, Jianjun Xu, Qi-Jun Zhang

Department of Electronics, Carleton University, Ottawa, Canada, K1S 5B6

Received 9 August 2000; revised 14 September 2000; accepted 22 September 2000

**ABSTRACT:** Artificial neural networks (ANN) recently gained attention as a fast and flexible vehicle to microwave modeling and design. Fast neural models trained from measured/simulated microwave data can be used during microwave design to provide instant answers to the task they have learned. We review two important aspects of neural-network-based microwave modeling, namely, model development issues and nonlinear modeling. A systematic description of key issues in neural modeling approach such as data generation, range and distribution of samples in model input parameter space, data scaling, etc., is presented. Techniques that pave the way for automation of neural model development could be of immense interest to microwave engineers, whose knowledge about ANN is limited. As such, recent techniques that could lead to automatic neural model development, e.g., adaptive controller and adaptive sampling, are discussed. Neural modeling of nonlinear device/circuit characteristics has emerged as an important research area. An overview of nonlinear techniques including small/large signal neural modeling of transistors and dynamic recurrent neural network (RNN) modeling of circuits is presented. Practical microwave examples are used to illustrate the reviewed techniques. © 2001 John Wiley & Sons, Inc. *Int J RF and Microwave CAE* 11: 4–21, 2001.

**Keywords:** neural networks; microwave; model development issues; Huber quasi-Newton; nonlinear modeling

## INTRODUCTION

In the recent years, a computer-aided design (CAD) approach based on neural networks has been introduced for microwave modeling, simulation, and optimization. Fast, accurate, and reliable neural network models can be developed from measured/simulated microwave data. Once developed, these neural models can be used in place of computationally intensive physics/EM models of active/passive devices to speed up microwave design [1, 2]. Neural network techniques

have been used to model a wide variety of microwave devices/circuits such as transmission line components [3–16], bends [17, 18], vias [19–23], CPW components [22, 24, 25], spiral inductors [26, 27], FET [1, 3, 4, 9, 11, 15, 28–38], HBT [30, 39, 40], HEMT [36, 41, 42], waveguides [22, 43], laser diodes [44], filters [16, 24, 30, 45–51], amplifiers [11, 29, 31, 35, 37, 52–54], mixers [54], antennas [25, 55–66], and embedded resistors [51]. Neural networks have also been used in object recognition [64, 67, 68], wave propagation [69, 70], impedance matching [71–73], electronic packaging [10], inverse modeling [33, 43, 68, 74–76], circuit design and optimization [7, 8, 10, 11, 14, 16, 18, 24, 29, 31, 35, 45, 47–49, 52, 72, 77],

Correspondence to: Q. J. Zhang; e-mail: qjz@doe.carleton.ca.

synthesis [16, 52, 77], and yield prediction [33, 38]. Neural models are much faster than original detailed physics/EM models [1, 35], more accurate than polynomial and empirical models [78], allow more dimensions than table lookup models [79], and are easier to develop when a new device/technology is introduced [7].

The process of neural model development is not trivial and involves many critical issues such as data generation, scaling, neural network training, etc. [1]. As neural network techniques are relatively new to the microwave community, it is often not easy for the microwave engineers to make decisions regarding these issues. For example, the range and the distribution of training data in the model input parameter space where training data needs to be generated, methods of input/output scaling to be performed, technique for neural network weights initialization to be employed, etc., are not obvious. Some of these issues are either briefly discussed or implied in the existing literature [27, 50, 80], while others are not. In this work, we present a detailed description of the key and fundamental issues in microwave neural modeling.

To begin with, a neural network does not know any information about the microwave device/circuit problem. In order to represent the device/circuit behavior, the neural network has to be trained with the corresponding data. As such, neural network training is the most crucial step in model development. One of the frequently encountered problems is the presence of measurement/simulation errors in the microwave data. A novel training algorithm known as the Huber quasi-Newton (HQN) has been recently proposed [30, 81]. The HQN training algorithm is robust against small/large measurement/simulation errors in training data and is capable of producing reliable neural network models in the presence of such errors. We describe the HQN algorithm followed by an illustration through a MESFET example.

There are other critical issues involved in developing neural models for practical and complicated microwave problems. For example, the structure of the model (i.e., the number of hidden layer neurons) cannot be deterministically determined [76]. The problems of overlearning and underlearning have to be dealt with. It is not easy to determine or predict the number of data samples needed to develop a reasonably accurate model. These problems led to examination of techniques that could lead to automation of neu-

ral model development. Recent techniques discussed in this work include the multilayer perceptron neural network (MLPNN) process [27], adaptive controller [30], and neural network training-driven adaptive sampling algorithm [9].

Neural modeling of nonlinear devices/circuits and their dynamic behaviors is one of the most important areas of microwave CAD. First, let us consider transistor modeling. In general, existing approaches for transistor modeling are based on lumped equivalent circuits, e.g., [82–85]. The equivalent circuit approach involves determination of an equivalent circuit topology and formulation of the circuit elements. Such an approach not only requires experience but also a difficult trial-and-error process. As a result of rapid changes in semiconductor technology, development of models to represent the new transistor behaviors has become a continuous activity. Recently, researchers started investigating neural network approaches to model transistor DC [1, 3, 15, 32], small-signal [4, 39], and large-signal [29, 35, 41, 42, 53] behaviors. Neural network transistor models can be developed through a computerized training process, and the models for a new semiconductor device can be developed even if the device theory/equations are unavailable. We review various existing transistor modeling approaches in this study. Second, consider modeling of dynamic characteristics of nonlinear circuits. Conventional neural network structures, e.g., an MLP, are not suitable for modeling the dynamic characteristics. We review a recent work in this direction [54] that advocates the use of recurrent neural networks [76] for dynamic modeling.

In the following section, the problem of microwave neural network modeling is defined. Key issues in neural model development are described in the third section. Techniques that could lead to automation of neural model development are presented in the fourth section. In the fifth section, neural network modeling of nonlinear devices and dynamic behaviors of nonlinear circuits is described. The final section contains the conclusions. Several illustration examples are presented throughout the work.

## MICROWAVE NEURAL MODELING: PROBLEM STATEMENT

Let  $\mathbf{x}$  represent an  $N_x$  vector containing physical/geometrical parameters of a microwave device/circuit, e.g., gate length and gate width of an

FET or width and spacing of transmission lines. Let  $\mathbf{y}$  represent an  $N_y$  vector containing the responses of the device/circuit under consideration, e.g., drain current of an FET or mutual inductance between transmission lines. The physics/EM relationship between  $\mathbf{y}$  and  $\mathbf{x}$  can be represented as

$$\mathbf{y} = \mathbf{y}(\mathbf{x}). \quad (1)$$

This relation can be highly nonlinear and multidimensional. The theoretical model for this relationship may not be available (e.g., a new semiconductor device), or theory may be too complicated to implement, or the theoretical model may be computationally too intensive for online microwave design and repetitive optimization [e.g., three-dimensional (3D) full-wave EM analysis inside a Monte Carlo statistical design loop]. We aim to develop a fast and accurate neural model by teaching/training a neural network to learn the microwave problem through a set of measured/simulated sample pairs called training data  $\{(\mathbf{x}_p, \mathbf{d}_p), p \in T_R\}$ , where  $\mathbf{d}_p$  represents the measured/simulated output  $\mathbf{y}$  for the sample input  $\mathbf{x}_p$ , and  $T_R$  represents the index set of training data. Let the neural network model be defined as

$$\tilde{\mathbf{y}} = \tilde{\mathbf{y}}(\mathbf{x}, \mathbf{w}), \quad (2)$$

where  $\mathbf{w}$  represents the parameters inside the neural network also called the weight vector. For training purpose, we define an error function  $E(\mathbf{w})$  as

$$E(\mathbf{w}) = \sum_{p \in T_R} (e_p(\mathbf{w}))^q, \quad (3)$$

where  $q$  represents the  $q$ th norm and  $e_p(\mathbf{w})$  is the error due to  $p$ th sample given by

$$e_p(\mathbf{w}) = \left[ \frac{1}{q} \sum_{k=1}^{N_y} |\tilde{y}_k(\mathbf{x}_p, \mathbf{w}) - d_{pk}|^q \right]^{1/q}, \quad (4)$$

where  $d_{pk}$  is the  $k$ th element of  $\mathbf{d}_p$  and  $\tilde{y}_k(\mathbf{x}_p, \mathbf{w})$  is the  $k$ th output of the neural network for input sample  $\mathbf{x}_p$ . The objective of neural network training is to find  $\mathbf{w}^*$  such that  $E(\mathbf{w})$  is minimized. The definition of  $\mathbf{w}$  and the approach by which  $\tilde{\mathbf{y}}$  is computed through  $\mathbf{x}$  and  $\mathbf{w}$  determine the structure of the neural network. Some of the neural network structures that have been used for mi-

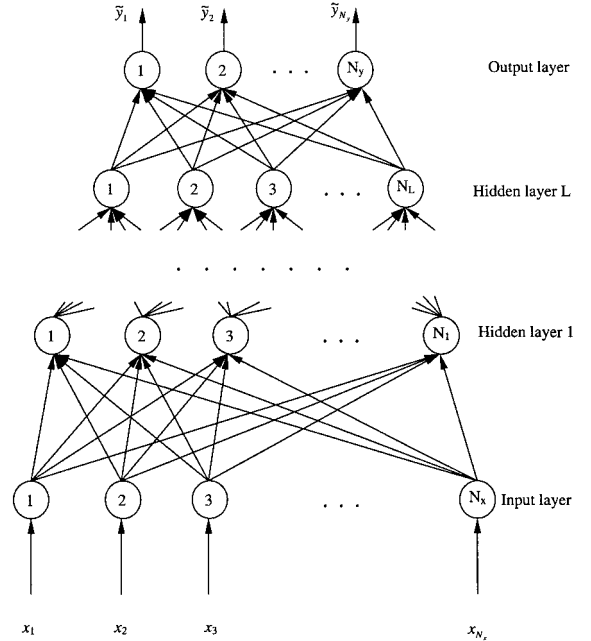
crowave modeling are multilayer perceptrons (MLP) [1, 4, 19, 20, 27, 28, 53, 76, 86], radial basis function (RBF) networks [28, 36, 56, 80, 86], wavelet networks [28, 48, 49, 53], and knowledge-based neural networks (KBNN) [1, 3, 13, 86].

As an example, for the MLP structure shown in Figure 1,  $\tilde{\mathbf{y}}$  is computed starting with the input layer  $z_i^0 = x_i$ , and then proceeding through the hidden layers,

$$z_i^l = \sigma \left( \sum_{j=1}^{N_{l-1}} w_{ij}^l z_j^{l-1} + w_{i0}^l \right),$$

$$i = 1, 2, \dots, N_l, l = 1, 2, \dots, L + 1, \quad (5)$$

and finally  $\tilde{y}_k = z_k^{L+1}$ , where input and output layers are denoted as hidden layer 0 and hidden layer  $L + 1$ ,  $x_i$  is the  $i$ th input to the neural network,  $N_l$  is the number of neurons in hidden layer  $l$ ,  $z_i^l$  is the output of  $i$ th neuron of  $l$ th hidden layer,  $w_{ij}^l$  represents weight of the link between  $j$ th neuron of  $l - 1$ th hidden layer and  $i$ th neuron of  $l$ th hidden layer,  $w_{i0}^l$  is the bias parameter of  $i$ th neuron of  $l$ th hidden layer, and  $L$  is the total number of actual hidden layers. In eq. (5),  $\sigma(\cdot)$  is the neuron activation function,



**Figure 1.** Multilayer perceptrons (MLP) as an example of neural network structure. Typically, an MLP consists of an input layer, one or more hidden layers, and an output layer.

which is usually a sigmoid function

$$\sigma(\gamma) = \frac{1}{(1 + e^{-\gamma})} \quad (6)$$

for the hidden layers and a linear function for the output layer.

## NEURAL MODEL DEVELOPMENT: KEY ISSUES

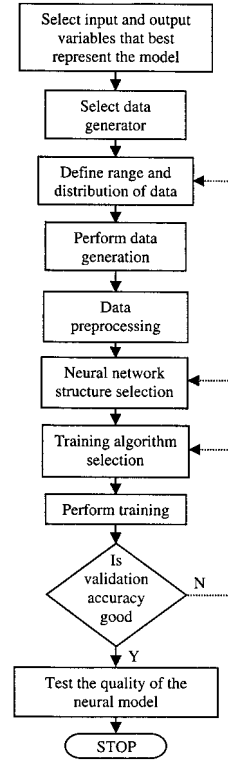
The process of developing neural models is not trivial and involves critical issues such as data generation, scaling, etc. Although some of these aspects are either implied or described in the existing neural network literature, their description is mainly from signal processing or pattern recognition perspective. Since neural network modeling is relatively new to microwave readers, we gathered and translated these ideas into microwave language. Based on this information and our own experiences, we provide in this section, a systematic overview of the key and fundamental issues of microwave-oriented neural model development. A sequential flow diagram summarizing various steps involved in neural model development is shown in Figure 2.

### Data Generation

The preliminary step in neural model development is the identification of model inputs and outputs. Once the inputs and outputs are identified, microwave device/circuit data needs to be collected/generated. For microwave applications, there are two types of data generators, namely measurement and simulation. In general, data generation means using a data generator to obtain the output  $\mathbf{d}_p$ , for each input sample  $\mathbf{x}_p$ . The total number of samples to be generated for a given microwave problem is chosen such that the developed neural model accurately represents the original problem. The choice of a data generator depends on the application and the availability of the data generator. Microwave neural model development using data from measurement and data from simulation are compared in Table 1.

### Range and Distribution of Samples in Model Input Parameter Space

Typically, neural model development requires three sets of data, namely the training data, the



**Figure 2.** Sequential flowchart summarizing various steps involved in neural model development.

validation data, and the test data. We define  $T_R$ ,  $V$ , and  $T_E$  as the index sets of training data, validation data, and test data, respectively. Training data is used to guide the training process, i.e., to update the neural network weights during training. Validation data is used to monitor the quality of the neural model during training, so as to indicate when to terminate the training process. Test data is used to examine the final quality of the developed model.

Suppose the range of input parameters over which the neural model would be used during microwave design is  $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$ . Validation data and test data should be generated in the range  $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$ . Training data could be generated in the same range as well. We suggest, where feasible, that the training data be sampled slightly beyond the model utilization range, e.g.,  $[\mathbf{x}_{\min} - \Delta, \mathbf{x}_{\max} + \Delta]$ . This is to ensure good performance of the neural model at the boundaries of input parameter space.

Once the range of input parameters is decided, the next step is to choose a sampling strategy. Suggested sample distributions are uniform grid distribution, nonuniform grid distribution, star distribution, central-composite distribution, and

**TABLE 1. Comparison of Neural-Based Microwave Model Development Using Data from Two Types of Data Generators, Namely Measurement and Simulation**

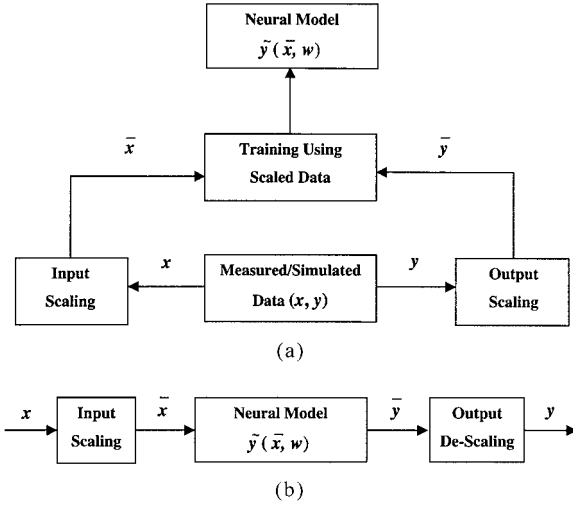
Basis of Comparison	Neural Model Development Using Measurement Data	Neural Model Development Using Simulation Data
Availability of problem theory equations	Model can be developed even if the theory/equations are not known or difficult to implement in CAD.	Model can be developed only for the problems that have theory and implementation.
Assumptions	No assumptions involved and the model could include all the effects of the original problem, e.g., 3D-fullwave effects, fringing effects, etc.	Often involves assumptions and the model will be limited by the same assumptions as simulator, e.g., 2.5D EM.
Input parameter sweep	Data generation could either be expensive or infeasible, if a geometrical parameter, e.g., transistor gate-length needs to be sampled.	Relatively easier to sweep any parameter in the simulator.
Sources of small and large/gross errors	Due to equipment limitations and tolerances.	Due to accuracy limitations and nonconvergence of simulations.
Feasibility of getting desired output	Type of data is limited to measurable responses only. For example, drain charge of a transistor may not be easy to measure.	Any response can be modeled as long as it can be computed in the simulator.

random distribution [1]. In uniform grid distribution, each input parameter is sampled at equal intervals, while in nonuniform grid distribution, each input parameter is sampled at unequal intervals. Uniform grid distribution could be a default strategy. Nonuniform grid distribution is used when we have a certain understanding of the problem and deliberately choose dense samples in subregions of the input space where problem behavior is highly nonlinear. Total number of samples for grid distribution is  $\prod_{i=1}^{N_x} n_i$ , where  $n_i$  is the number of grids along  $i$ th axis, i.e.,  $x_i$  axis. In the case of star distribution [18], a middle point of the input space  $\mathbf{x}_{\text{mid}}$  is first determined. Samples are then obtained by perturbing one element of  $\mathbf{x}_{\text{mid}}$  at a time, either toward its maximum value or toward its minimum value. Total number of samples in star distribution is  $2N_x + 1$ . This sample distribution is used when data generation is very expensive, and model behavior is assumed to be smooth within  $[\mathbf{x}_{\text{min}}, \mathbf{x}_{\text{max}}]$ . In central-composite distribution,  $2^{N_x}$  corner points are generated in addition to the samples of star distribution [19]. In random distribution, each input sample  $\mathbf{x}_p$  is a random variable between  $\mathbf{x}_{\text{min}}$  and  $\mathbf{x}_{\text{max}}$ . Random distribution is used when input parameter space is of high dimension.

## Data Scaling

Data scaling is an essential step to improve the learning/training process of neural networks. The order of magnitude of input/output parameter values in microwave applications can be very different even for the same parameter, in contrast to binary “0” and “1” situation in typical pattern recognition applications. As such, scaling of training data is suggested for efficient microwave neural model training. Scaling of the data samples can be performed on their input and/or output parameters. Input/output scaling and descaling in neural model development and in neural model usage are shown in Figure 3. Commonly suggested scaling schemes are linear scaling, log scaling, and two-sided log scaling. Let  $x$ ,  $x_{\text{min}}$ ,  $x_{\text{mid}}$ , and  $x_{\text{max}}$  represent a generic element in the vectors  $\mathbf{x}$ ,  $\mathbf{x}_{\text{min}}$ ,  $\mathbf{x}_{\text{mid}}$ , and  $\mathbf{x}_{\text{max}}$  of original data, respectively. Let  $\bar{x}$ ,  $\bar{x}_{\text{min}}$ , and  $\bar{x}_{\text{max}}$  represent a generic element in the vectors  $\bar{\mathbf{x}}$ ,  $\bar{\mathbf{x}}_{\text{min}}$ , and  $\bar{\mathbf{x}}_{\text{max}}$  of scaled data, respectively, where  $[\bar{\mathbf{x}}_{\text{min}}, \bar{\mathbf{x}}_{\text{max}}]$  is the input parameter range after scaling. Linear scaling is given by

$$\bar{x} = \bar{x}_{\text{min}} + \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} (\bar{x}_{\text{max}} - \bar{x}_{\text{min}}). \quad (7)$$



**Figure 3.** Input/output scaling and descaling (a) in neural model development and (b) in neural model usage.

The log scaling is given by

$$\bar{x} = \ln(x - x_{\min} + 1), \quad (8)$$

and two-sided log scaling is defined as

$$\bar{x} = \text{sgn}(x - x_{\text{mid}}) \frac{\ln \left[ 1 + \frac{|x - x_{\text{mid}}| \alpha}{(x_{\text{max}} - x_{\text{mid}})} \right] \bar{x}_{\text{max}}}{\ln(1 + \alpha)}, \quad (9)$$

where  $\text{sgn}(\cdot)$  is a sign function and  $\alpha$  is a constant.

The same formulas hold good for scaling the output values of data samples. Linear scaling of data balances the ranges of different inputs or outputs. Applying log scale to outputs with large variations balances large and small magnitudes of the same output in different regions of the model. Two-sided log scale is similar to log scale except that it is used to avoid the overshadowing of midrange values of the response by large increasing and large decreasing trends. In general, the input/output scaling makes the problem better conditioned for training, thereby facilitating easier learning process.

### Initialization of Neural Model Weight Parameters

In order to provide a good starting point for training (optimization), the neural network weights need to be initialized. The random-weights method is the most widely used strategy

for MLP weight initialization in which the weights are initialized with small random values, e.g., in the range  $[-0.5, 0.5]$ . RBF and wavelet networks can be initialized by estimating the parameters of hidden neuron activation functions, i.e., centers and radii of RBF and translation and dilation of wavelets, with the help of an unsupervised learning process based on training data. The physical/electrical experience or existing knowledge of the problem can be utilized to provide a good starting point for KBNN.

### Overlearning and Underlearning

After initializing the weight parameters, a training algorithm can be used to train the neural network. Ability of a neural network to estimate output  $y$  accurately when presented with input  $x$  never seen during training is called generalization ability. Overlearning is observed when the neural network memorizes the training data, but cannot generalize well. We denote the error defined in eq. (3) as the training error  $E_{T_R}$ . The validation error  $E_V$  is similarly defined, except that the index set  $T_R$  is replaced by index set  $V$ . Overlearning is detected when  $E_{T_R}/N_{T_R}$  is small and  $E_V/N_V \gg E_{T_R}/N_{T_R}$ , where  $N_{T_R}$  and  $N_V$  are the number of samples in training and validation data sets, respectively. Possible reasons are too many hidden neurons or insufficient training data. To remedy the situation, a certain number of hidden neurons can be deleted from the neural network, and/or more samples can be added to the training data.

Underlearning is a situation where the neural network has difficulties even to learn the training data itself, i.e.,  $E_{T_R}/N_{T_R} \gg 0$ . Possible reasons for underlearning are insufficient hidden neurons, or insufficient training, or training gets stuck in a local minimum. The suggested remedies are adding more hidden neurons, or continuing training, or perturbing the current solution  $\mathbf{w}$  to escape from the local minimum and then continue training. Good training of a neural network is observed when  $E_{T_R}/N_{T_R}$  and  $E_V/N_V$  are close to each other and both are small.

### Quality Measures for a Developed Neural Model

There are various quality measures that can be used to evaluate the final performance of a trained neural model. For this purpose, an independent set of data, i.e., the test data is used. We define a

quantity  $\delta_{pk}$  as

$$\delta_{pk} = \frac{\tilde{y}_k(\mathbf{x}_p, \mathbf{w}) - d_{pk}}{d_{k,\max} - d_{k,\min}},$$

$$k = 1, 2, \dots, N_y, p \in T_E. \quad (10)$$

A quality criterion based on the  $q$ th-norm measure is then defined as

$$M_q = \left[ \sum_{p \in T_E} \sum_{k=1}^{N_y} |\delta_{pk}|^q \right]^{1/q}. \quad (11)$$

When  $q = 1$ , the average test error can be calculated directly from  $M_1$  as

$$\text{Average test error} = \frac{M_1}{N_{T_E} N_y}, \quad (12)$$

where  $N_{T_E}$  is the number of samples in test data set, and  $N_y$  is the number of neural model outputs. When  $q = 2$ , the  $q$ th-norm measure is the Euclidean distance between neural model prediction and test data. When  $q = \infty$ , the  $q$ th-norm measure is the maximum test error, which is also called the worst-case error among entire test data and all model outputs. Statistical measures such as error mean and correlation coefficient can also be used.

## Neural Network Structures and Training Algorithms

Appropriate neural network structure and suitable training algorithm are two major issues in developing neural models [80]. Commonly used neural network structures are MLP, RBF, KBNN, wavelet networks, and recurrent neural network (RNN) [1]. The most important and time-consuming step in model development is neural network training. Learning the microwave behavior, however difficult it is, is ultimately achieved through this process. The neural network would be taught with measured/simulated samples from the training set. Training of neural networks is an optimization process in the weight space and is often done using optimization-based training algorithms such as backpropagation (BP) [87, 88], conjugate-gradient [89], quasi-Newton [90], Levenberg–Marquardt [91], etc. However, global optimization algorithms, e.g., genetic algorithm [41], can also be used for neural network training. In this study, we review an advanced training algorithm called Huber quasi-Newton.

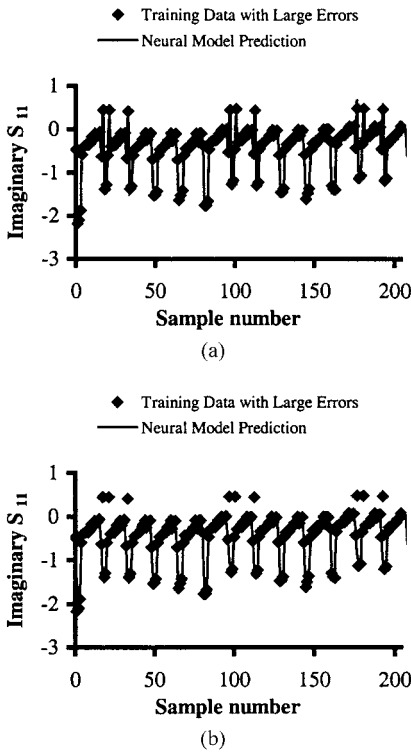
**Huber quasi-Newton Technique.** One of the most frequently encountered challenges in developing microwave neural models is the presence of small and large (gross) errors in training data. We, therefore, need a training technique that can yield accurate neural models in the presence of such unavoidable errors. The objective functions of conventional neural network training algorithms are formulated in  $l_2$  sense with  $q$  set as 2 in eqs. (3) and (4). Although  $l_2$ -norm-based training can handle small errors in training data, it can be misled by large (gross) errors resulting in non-reliable neural models. On the other hand,  $l_1$ -norm-based training is robust against large errors but is not very effective in dealing with small errors [92]. In summary, neither of the  $l_1$ - and the  $l_2$ -norm-based training can discriminate and treat small and large errors differently. Huber function [93, 94] is a smooth combination of  $l_1$  and  $l_2$  norms and has been recently applied to circuit modeling and optimization [95, 96]. The Huber quasi-Newton training technique [30, 81] utilizes the Huber norm along with quasi-Newton update formulas [1, 80].

As an illustration, a MESFET was modeled using the HQN training algorithm. The MESFET model has four inputs, namely frequency, drain voltage, gate voltage, and channel thickness. There are eight outputs, i.e., real and imaginary parts of two-port  $S$  parameters  $S_{11}$ ,  $S_{12}$ ,  $S_{21}$ , and  $S_{22}$ . Data was generated from simulation using *OS490* [97]. There were a few obvious large errors in data that can be attributed to nonconvergence of the simulation due to input samples in extreme locations. A three-layer MLP neural network with 15 hidden neurons was used. A comparison of neural models developed using  $l_2$ -based training method and the HQN can be seen in Figure 4. HQN technique yields accurate neural models of MESFET as compared to standard  $l_2$  technique in the presence of large data errors. Neural models in this example and all the subsequent examples in the study were developed using *NeuroModeler* [98].

## NEURAL MODEL DEVELOPMENT: TOWARD AUTOMATION

As mentioned earlier, the process of developing neural models involves critical issues. For example, identifying or choosing the structure of the model (i.e., number of hidden layers, number of neurons in each hidden layer), estimating the





**Figure 4.** Example of comparison between neural model prediction and training data for a MESFET. The training data has a few large/gross errors. (a) Neural model trained by standard  $l_2$  method is affected by large errors. (b) Neural model trained by HQN technique is not affected by large errors.

number of samples needed for training and validation, detecting the situations of overlearning and underlearning during the training process, etc. There has been significant work in the neural network research area to determine proper network size, e.g., constructive algorithm [99], network pruning [100], and regularization [101]. In this section, we review the new techniques in the microwave-artificial neural networks (ANN) area that are aimed at addressing the above-mentioned issues to some extent. In future, these techniques could either directly or indirectly lead to automation of neural model development.

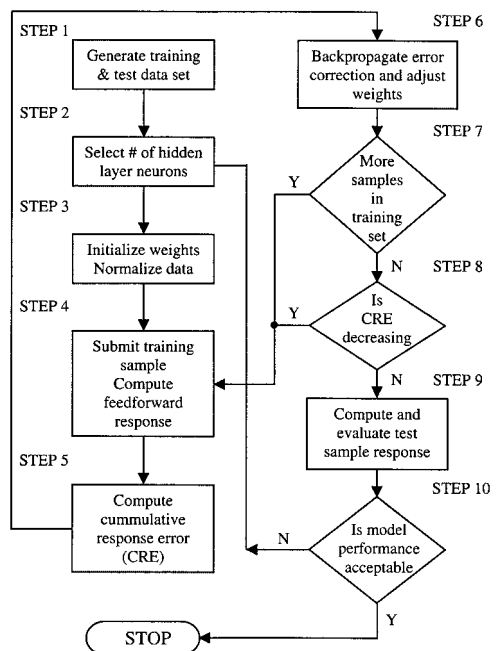
### MLPNN Process

In [27], fast and accurate neural network models of a spiral inductor were developed. The neural network structure used was the conventional MLP network, and the model development process was

termed the MLPNN process. The flow diagram of MLPNN process presented in this work (see Fig. 5) is noteworthy. The MLPNN process is an algorithm, however, human interaction is necessary in several steps. Although it is not a completely automated methodology, it gives an insight to the subject of neural modeling automation. The flow diagram indicates that there is a scope for automation of the actions such as selection of number of hidden layer neurons and addition of more samples to the training data set.

### Multistage Training with Adaptive Controller

Recently, a multistage training algorithm to address highly nonlinear and nonsmooth modeling problems, e.g., microwave filters [49], was proposed [30]. The multistage algorithm decomposes the original complicated microwave modeling problem (i.e., microwave responses to be modeled) into simpler subtasks. In each training stage, different neural network structure and training algorithms are suitably used to model a particular subtask, resulting in a subneural model. In the end, the submodels from various stages are com-

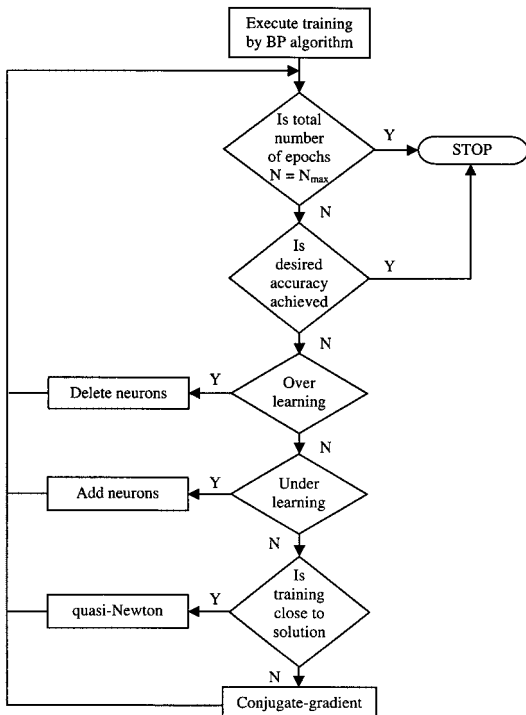


**Figure 5.** Flow diagram of the MLPNN model development process [27].

bined to produce the final neural model that represents the overall behavior. The multistage algorithm has a built-in adaptive controller subroutine, the flowchart of which is shown in Figure 6. The adaptive controller is an automation technique that reduces human interaction in each stage of training. It offers a robust start when far from solution, faster convergence when closer to solution, reasonable judgment for neural network size, and detection and escape from traps of local minimum.

How does the adaptive controller work? The training process starts with a three-layer MLP neural network and an initial estimate of number of hidden neurons. During the training process, the adaptive controller periodically taps the information such as training error, validation error, gradient of the training error, etc., and executes one of the following actions as needed:

- continue training
- change from one training algorithm to another
- escape traps of local minima by randomly perturbing neural network weights
- add or delete neurons
- stop training



**Figure 6.** Flowchart of the adaptive controller subroutine used in multistage training algorithm [30].

## Adaptive Sampling

Data generation is a crucial step toward developing accurate and reliable neural models. A large set of training data ensures good model accuracy but is expensive, while a smaller set yields a nonreliable model. A conventional grid sampling may lead to unnecessarily large number of samples in smooth portions and inadequate number of samples in highly nonlinear portions of the response to be modeled. Several data sampling [102, 103] and data exploration [104, 105] techniques were previously studied.

Recently, a neural network training-driven adaptive sampling algorithm [9] has been proposed. The sampling algorithm starts with small amounts of training and test data. A reasonable neural network structure is trained and tested. If the resulting neural model is not satisfactory, the sampling algorithm determines the additional number of training and test samples required and the values of the input parameters at which these additional samples need to be generated. The algorithm dynamically drives simulator or measurement setup to generate these additional samples.

A MESFET neural model was developed using the sampling algorithm [9]. The input space  $x$  contains gate-length, gate-source voltage, and drain-source voltage. Drain current was the only neural network output  $y$ . A three-layer MLP with 18 hidden layer neurons was used. The algorithm dynamically drives the *OS490* simulator [97] during the neural model development. Table 2 shows that the accuracy of the models from the adaptive sampling algorithm is better as compared to models trained from the same amount of training data using conventional grid sampling. Contrary to the conventional grid sampling, the adaptive sampling algorithm generates more samples in nonlinear regions and fewer samples in smooth regions of the input space as shown in Figure 7.

## NEURAL MODELS FOR NONLINEAR DEVICES/CIRCUITS

Neural modeling of nonlinear devices and dynamic responses of microwave circuits is a recent thrust in the microwave-ANN area. In this section, we review the neural network approaches to modeling of transistor DC, small-signal, and large-signal behaviors. Neural modeling of non-

**TABLE 2. Model Accuracy Comparison Between Conventional Grid Distribution and the Adaptive Sampling Algorithm for the MESFET Example<sup>a</sup>**

Adaptive Sampling Algorithm			Conventional Grid Distribution	
User-Desired Accuracy ( $E_d$ ) (%)	No. of Training Samples as Determined by the Algorithm	Model Accuracy ( $E_l$ ) (%)	No. of Training Samples Used	Model Accuracy ( $E_l$ ) (%)
2.00	35	1.84	36	5.14
1.75	44	1.56	45	4.75
1.50	53	1.45	54	3.72
1.25	62	1.22	64	1.44
1.00	71	0.88	75	1.05

<sup>a</sup> The advantage of the sampling algorithm is more significant when less training data is available.

linear circuit behaviors using recurrent neural networks is also presented.

## DC Models

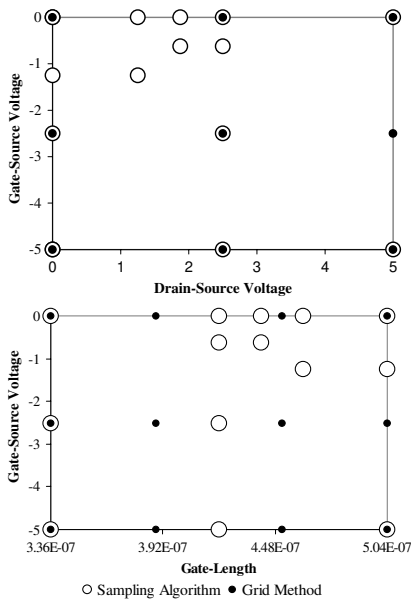
DC characteristics of transistors can be obtained either from simulation or measurement. Neural networks, e.g., MLP, can be trained using such data to produce fast and accurate DC neural models [15]. The inputs to the neural network include bias inputs (e.g., gate voltage, drain voltage), transistor geometrical parameters (e.g., gate-length, gate-width, channel thickness), physical/material parameters (e.g., doping density,

substrate permittivity), and frequency. In general, the output of the DC neural models is drain current. In [3], existing empirical formulas [106] were used to develop DC models using the concept of knowledge-based neural networks. The empirical functions were used as hidden neuron activation functions in place of conventional sigmoid neurons. The KBNN training needs fewer samples and lesser training time, and yet yields accurate models with better extrapolation capability as compared to MLP models.

## Small Signal S-Parameter Models

Neural networks can learn measured/simulated small-signal data of transistors accurately. The inputs to the neural network could include bias, geometrical, physical parameters, and frequency. The outputs of the neural models are the real/imaginary parts or magnitude/phase of two-port  $S$  or  $Y$  parameters. MLP neural networks have been used to develop  $S$ -parameter transistor models, e.g., MESFET [4], HBT [39]. In [4], a library of MESFET  $S$ -parameter models was developed using both conventional MLP approach and the hierarchical neural network approach. It was shown that the hierarchical approach [4, 80] required fewer data and shorter training time as compared to the MLP approach.

In [39], small-signal HBT neural models were developed.  $S$ -parameter data for a typical HBT was measured for various input combinations of bias current, bias voltage, and frequency, for five different transistor sizes. Three-layer perceptron neural networks (MLP3) were trained for each transistor size. The MLP3 networks take three inputs (collector bias current, bias voltage, and frequency) and give eight outputs (magnitudes and phases of  $S_{11}$ ,  $S_{12}$ ,  $S_{21}$  and  $S_{22}$ ). Average test



**Figure 7.** MESFET sample distribution of the conventional grid method and the adaptive sampling algorithm at a certain stage of training. As can be seen, the adaptive algorithm samples the nonlinear regions of the input space densely.

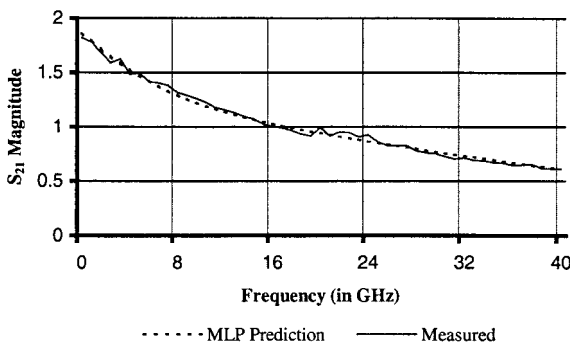
**TABLE 3. Average Test Errors of S-Parameter Neural Models for Different Transistor Sizes**

HBT Size	MLP (3-12-8) (%)	MLP (3-15-8) (%)	MLP (3-20-8) (%)	MLP (3-25-8) (%)
1	1.98	1.95	1.82	1.86
2	1.40	1.34	1.16	1.23
3	1.30	1.18	1.08	1.15
4	1.29	1.12	1.06	1.11
5	1.23	1.07	1.04	1.07

errors of the resulting neural models were observed to be well below 2% as shown in Table 3. The MLP3 with 20 hidden neurons (MLP 3-20-8) was found to give the best model accuracy for all the five transistor sizes. A good agreement of the neural model estimation of  $S_{21}$  magnitude with the measured data is shown in Figure 8.

### Large-Signal Models

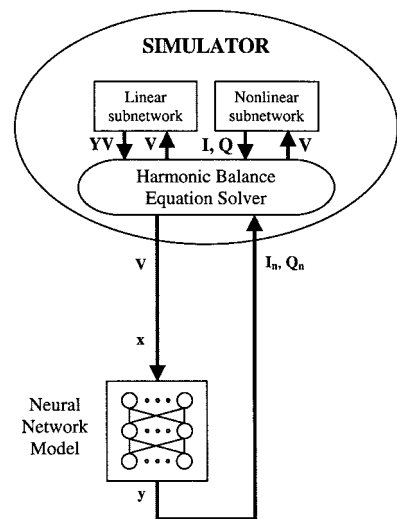
Neural network models to represent the large-signal behavior of transistors can be developed. In [35], terminal currents and charges for different configurations of MESFET were simulated at a number of bias points using *OS490* [97] with *Khatibzadeh* and *Trew* models [107]. The neural network model has six inputs, namely gate-length, gate-width, channel thickness, doping density, gate voltage, and drain voltage. The terminal currents and charges on the drain, gate, and source electrodes are the model outputs (six outputs). Neural networks were trained and the resulting large-signal neural models were plugged into a circuit simulator as shown in Figure 9. The large-signal neural model was used to satisfactorily perform DC, small-signal, and harmonic balance (HB) simulations.



**Figure 8.** Comparison of MLP (3-20-8) prediction and measured  $S_{21}$  magnitude.

### Neural Modeling Using Known Equivalent Circuit

As mentioned earlier, the commonly used transistor modeling approach is the lumped equivalent circuit approach. Developing such models requires experience and involves a trial-and-error process to determine the exact topology. Moreover, an equivalent model may not have any link with the physical/geometrical parameters of the device under consideration. Empirical formulas for such relations exist, and these relations can be easily learned by neural networks. A hybrid approach that utilizes existing knowledge in the form of known equivalent circuit and empirical formulas, together with the powerful learning and generalization abilities of neural networks was demonstrated for MESFET [37] and HEMT [41, 42]. In [37], MLP3 neural networks learned the bias-dependent lumped elements of the MESFET. The neural models were then combined



**Figure 9.** Incorporating the large signal neural network models of a transistor into harmonic balance simulator [35].

with the FDTD marching time algorithm leading to a first-order approximation of the device–wave interaction and a large-signal analysis. In [41], five-layer MLP neural networks (MLP5) were trained to learn the bias-dependent small-signal equivalent circuit elements.

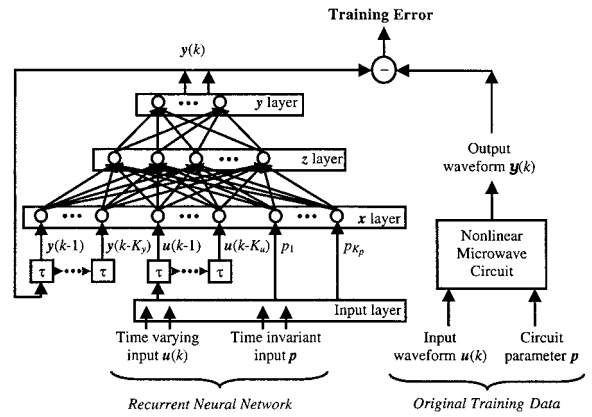
## Volterra Series Approach

The Volterra series approach reduces to a first-order kernel in case of short memory devices [108]. This significantly reduces the limitations of the classical Volterra series and avoids tedious task of measuring higher order kernels. Most electron devices including transistors satisfy the short memory condition. In [53], DC characteristics and small-signal  $S$  parameters of a transistor were measured. A group of 10 neural networks were trained to learn the drain-to-source current, gate-to-source current, and the real and imaginary parts of the two-port  $S$  parameters. The resulting neural models were used following the Volterra series approach to design a 13.6-GHz amplifier.

## Nonlinear Circuit Modeling Using Recurrent Neural Networks

Conventional feed-forward neural networks are well known for their learning and generalization capabilities. However, they can only map input–output relationships statically. To model nonlinear circuit responses in time domain, a neural network that can include temporal information is necessary. Recurrent neural network has been found to be a suitable candidate to accomplish this job. In the past, RNN were successfully used in engineering applications such as control [109, 110], speech recognition [111], etc.

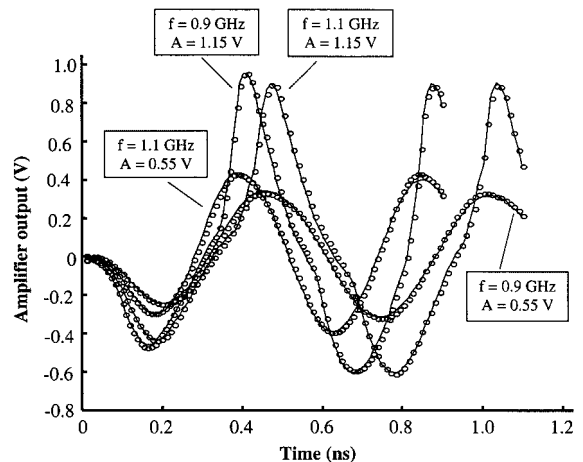
The structure of a typical RNN is shown in Figure 10. The input layer includes time-varying inputs  $\mathbf{u}$  and time-invariant inputs  $\mathbf{p}$ . The neural network outputs are denoted by  $\mathbf{y}$ . The first hidden layer of RNN denoted by  $\mathbf{x}$  contains buffered (time-delayed) history of  $\mathbf{y}$  fed back from the output layer, buffered history of  $\mathbf{u}$ , and  $\mathbf{p}$ . The second hidden layer  $\mathbf{z}$  contains sigmoid neurons. The RNN has to be trained to learn the dynamic characteristics of the nonlinear microwave circuit. The training data is a set of input and output waveforms of the original nonlinear circuit. Since the present outputs of the model not only depend



**Figure 10.** Structure of a typical recurrent neural network model [54].

upon the present inputs but also on previous inputs and outputs, a novel BP training scheme called backpropagation-through-time needs to be used [54].

An RFIC power amplifier was modeled using the RNN approach [54]. The amplifier contains eight NPN BJT's modeled by two internal HP-ADS [112] nonlinear models Q34 and Q37. Inputs to RNN are the voltage waveform of the amplifier input and the sampling cycle. Output of RNN is the voltage waveform of the amplifier output. The accuracy of the resulting RNN model can be seen in Figure 11. The RNN model was found to be much faster than the original simulation.



**Figure 11.** Comparison between output waveforms from original amplifier simulation ( $\circ$ ) and that from a RNN model ( $-$ ).

## CONCLUSIONS

The state-of-the-art technology of neural-network-based microwave modeling has been reviewed, with special attention to model development issues. Automation techniques for neural modeling that involve reduced human intervention have been discussed. A brief overview of the neural-network-based nonlinear device/circuit modeling has been presented. Practical microwave examples are used to illustrate the reviewed techniques. Research and development efforts are required to extract the full potential of neural networks, and there is a definite need for automated neural modeling CAD tools.

## ACKNOWLEDGMENT

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, in part by Micronet: a Canadian Network Centres of Excellence on Microelectronic Devices, Circuits and Systems, and in part by Nortel Networks.

## REFERENCES

1. Q.J. Zhang and K.C. Gupta, *Neural networks for RF and microwave design*, Artech House, Norwood, MA, 2000.
2. K.C. Gupta, Emerging trends in millimeter-wave CAD, *IEEE Trans Microwave Theory Tech* 46 (1998), 747–755.
3. F. Wang and Q.J. Zhang, Knowledge-based neural models for microwave design, *IEEE Trans Microwave Theory Tech* 45 (1997), 2333–2343.
4. F. Wang, V.K. Devabhaktuni, and Q.J. Zhang, A hierarchical neural network approach to the development of a library of neural models for microwave design, *IEEE Trans Microwave Theory Tech* 46 (1998), 2391–2403.
5. A. Patnaik, R.K. Mishra, G.K. Patra, and S.K. Dash, An artificial neural network model for effective dielectric constant of microstripline, *IEEE Trans Antennas Propagat* 45 (1997), 1697.
6. G. Antonini and A. Orlandi, Gradient evaluation for neural-networks-based electromagnetic optimization procedures, *IEEE Trans Microwave Theory Tech* 48 (2000), 874–876.
7. Q.J. Zhang, F. Wang, and M.S. Nakhla, Optimization of high-speed VLSI interconnects: A review, *Int J Microwave Millimeter-Wave CAE* 7 (1997), 83–107.
8. T. Horng, C. Wang, and N.G. Alexopoulos, Microstrip circuit design using neural networks, *IEEE MTT-S Int Microwave Symp Dig*, Atlanta, GA, 1993, pp. 413–416.
9. V.K. Devabhaktuni and Q.J. Zhang, Neural network training-driven adaptive sampling algorithm for microwave modeling, *Proc 30th European Microwave Conf*, Paris, France, Oct. 2000, pp. 222–225.
10. R.L. Mahajan, Design and optimization through physical/neural network models, *IEEE MTT-S Int Microwave Symp Workshop on Appl of ANN to Microwave Design*, Denver, CO, June 1997, pp. 1–16.
11. Q.J. Zhang, A neural network paradigm for microwave modeling, simulation and optimization, *IEEE MTT-S Int Microwave Symp Workshop on Appl of ANN to Microwave Design*, Denver, CO, June 1997, pp. 87–107.
12. P.M. Watson, M.Y. Mah, and L.L. Liou, Input variable space reduction using dimensional analysis for artificial neural network modeling, *IEEE MTT-S Int Microwave Symp Dig*, Anaheim, CA, June 1999, pp. 269–272.
13. F. Wang and Q.J. Zhang, Incorporating functional knowledge into neural networks, *Proc IEEE Int Conf Neural Networks*, Houston, TX, June 1997, pp. 266–269.
14. M. Nakhla, Application of neural networks to simulation and optimization of high-speed interconnects, *IEEE MTT-S Int Microwave Symp Workshop on Appl of ANN to Microwave Design*, Denver, CO, June 1997, pp. 49–66.
15. A.H. Zaabab, Q.J. Zhang, and M.S. Nakhla, Device and circuit level modeling using neural networks with faster training based on network sparsity, *IEEE Trans Microwave Theory Tech* 45 (1997), 1696–1704.
16. C. Cho and K. Gupta, EM-ANN modeling of overlapping open-ends in multilayer microstrip lines for design of bandpass filters, *IEEE APS Int Symp Dig*, Orlando, FL, July 1999, pp. 2592–2595.
17. J. Bandler, M. Ismail, J. Rayas-Sanchez, and Q. Zhang, New directions in model development for RF/microwave components utilizing artificial neural networks and space mapping, *IEEE APS Int Symp Dig*, Orlando, FL, July 1999, pp. 2572–2575.
18. J.W. Bandler, M.A. Ismail, J.E. Rayas-Sanchez, and Q.J. Zhang, Neuromodeling of microwave circuits exploiting space mapping technology, *IEEE MTT-S Int Microwave Symp Dig*, Anaheim, CA, June 1999, pp. 149–152.
19. P.M. Watson and K.C. Gupta, EM-ANN models for microstrip vias and interconnects in dataset circuits, *IEEE Trans Microwave Theory Tech* 44 (1996), 2495–2503.

20. P.M. Watson, K.C. Gupta, and R.L. Mahajan, Development of knowledge-based artificial neural network models for microwave components, IEEE MTT-S Int Microwave Symp Dig, Baltimore, MD, June 1998, pp. 9–12.
21. P.M. Watson, K.C. Gupta, and R.L. Mahajan, Applications of knowledge-based artificial neural network modeling to microwave components, Int J RF Microwave CAE 9 (May 1999), 254–260.
22. K.C. Gupta, EM-ANN models for microwave and millimeter-wave components, IEEE MTT-S Int Microwave Symp Workshop on Appl of ANN to Microwave Design, Denver, CO, June 1997, pp. 17–47.
23. P. Watson and K.C. Gupta, EM-ANN models for via interconnects in microstrip circuits, IEEE MTT-S Int Microwave Symp Dig, San Francisco, CA, June 1996, pp. 1819–1822.
24. P.M. Watson and K.C. Gupta, Design and optimization of CPW circuits using EM-ANN models for CPW components, IEEE Trans Microwave Theory Tech 45 (1997), 2515–2523.
25. P. Watson, G. Creech, and K. Gupta, Knowledge based EM-ANN models for the design of wide bandwidth CPW patch/slot antennas, IEEE APS Int Symp Dig, Orlando, FL, July 1999, pp. 2588–2591.
26. G.L. Creech, B. Paul, C. Lesniak, T. Jenkins, R. Lee, and M. Calcaterra, Artificial neural networks for accurate microwave CAD applications, IEEE MTT-S Int Microwave Symp Dig, San Francisco, CA, June 1996, pp. 733–736.
27. G.L. Creech, B.J. Paul, C.D. Lesniak, T.J. Jenkins, and M.C. Calcaterra, Artificial neural networks for fast and accurate EM-CAD of microwave circuits, IEEE Trans Microwave Theory Tech 45 (1997), 794–802.
28. Y. Harkouss, J. Rousset, H. Chehade, E. Ngoya, D. Barataud, and J.P. Teyssier, Modeling microwave devices and circuits for telecommunications system design, Proc IEEE Int Conf Neural Networks, Anchorage, Alaska, May 1998, pp. 128–133.
29. A.H. Zaabab, Q.J. Zhang, and M. Nakhla, Analysis and optimization of microwave circuits & devices using neural network models, IEEE MTT-S Int Microwave Symp Dig, San Diego, CA, May 1994, pp. 393–396.
30. V.K. Devabhaktuni, C. Xi, F. Wang, and Q.J. Zhang, Robust training of microwave neural models, IEEE MTT-S Int Microwave Symp Dig, Anaheim, CA, June 1999, pp. 145–148.
31. G. Kothapalli, Artificial neural networks as aids in circuit design, Microelectronics J 26 (1995), 569–578.
32. V.B. Litovski, J.I. Radjenovic, Z.M. Mrcarica, and S.L. Milenkovic, MOS transistor modeling using neural network, Elect Lett 28 (1992), 1766–1768.
33. G.L. Creech and J.M. Zurada, Neural network modeling of GaAs IC material and MESFET device characteristics, Int J RF and Microwave CAE 9 (May 1999), 241–253.
34. J. Rousset, Y. Harkouss, J.M. Collantes, and M. Campovecchio, An accurate neural network model of FET intermodulation and power analysis, Proc 26th European Microwave Conf, Prague, Czech Republic, Sept. 1996, pp. 16–19.
35. A.H. Zaabab, Q.J. Zhang, and M. Nakhla, A neural network modeling approach to circuit optimization and statistical design, IEEE Trans Microwave Theory Tech 43 (1995), 1349–1358.
36. J.A. Garcia, A.T. Puente, A.M. Sanchez, I. Santamaria, M. Lazaro, C.J. Pantaleon, and J.C. Pedro, Modeling MESFETs and HEMTs intermodulation distortion behavior using a generalized radial basis function network, Int J RF and Microwave CAE 9 (May 1999), 261–276.
37. S. Goasguen, S.M. Hammadi, and S.M. El-Ghazaly, A global modeling approach using artificial neural network, IEEE MTT-S Int Microwave Symp Dig, Anaheim, CA, June 1999, pp. 153–156.
38. G.L. Creech, Neural networks for the design and fabrication of integrated circuits, IEEE MTT-S Int Microwave Symp Workshop on Appl of ANN to Microwave Design, Denver, CO, June 1997, pp. 67–86.
39. V.K. Devabhaktuni, C. Xi, and Q.J. Zhang, A neural network approach to the modeling of heterojunction bipolar transistors from S-parameter data, Proc 28th European Microwave Conf, Amsterdam, Netherlands, Oct. 1998, pp. 306–311.
40. M. Vai and S. Prasad, Qualitative modeling heterojunction bipolar transistors for optimization: A neural network approach, Proc IEEE/Cornell Conf Adv Concepts in High Speed Semiconductor Dev and Circuits, 1993, pp. 219–227.
41. K. Shirakawa, M. Shimizu, N. Okubo, and Y. Daido, A large-signal characterization of an HEMT using a multilayered network, IEEE Trans Microwave Theory Tech 45 (1997), 1630–1633.
42. K. Shirakawa, M. Shimizu, N. Okubo, and Y. Daido, Structural determination of multilayered large-signal neural-network HEMT model, IEEE Trans Microwave Theory Tech 46 (1998), 1367–1375.
43. G. Fedi, S. Manetti, G. Pelosi, and S. Selleri, Design of cylindrical posts in rectangular waveguide by neural network approach, IEEE APS Int Symp Dig, Salt Lake City, UT, July 2000, pp. 1054–1057.
44. Q.J. Zhang, G. Wilson, R. Venkatachalam, A. Sarangan, J. Williamson, and F. Wang, Ultra fast neural models for analysis of electro/optical interconnects, Proc IEEE Electronic Components and Tech Conf, San Jose, CA, May 1997, pp. 1134–1137.

45. M.H. Bakr, J.W. Bandler, M.A. Ismail, J.E. Rayas-Sanchez, and Q.J. Zhang, Neural space mapping EM optimization of microwave structures, *IEEE MTT-S Int Microwave Symp Dig*, Boston, MA, June 2000, pp. 879–882.
46. P. Burrascano, M. Dionigi, C. Fancelli, and M. Mongiardo, A neural network model for CAD and optimization of microwave filters, *IEEE MTT-S Int Microwave Symp Dig*, Baltimore, MD, June 1998, pp. 13–16.
47. G. Fedi, A. Gaggelli, S. Manetti, and G. Pelosi, Direct-coupled cavity filters design using a hybrid feedforward neural network–finite elements procedure, *Int J RF and Microwave CAE* 9 (May 1999), 287–296.
48. S. Bila, Y. Harkouss, M. Ibrahim, J. Rousset, E. N’Goya, D. Billargeat, S. Verdeyme, M. Auborg, and P. Guillon, An accurate wavelet neural-network-based model for electromagnetic optimization of microwave circuits, *Int J RF and Microwave CAE* 9 (May 1999), 297–306.
49. S. Verdeyme, D. Billargeat, S. Bila, S. Moraud, H. Blondeaux, M. Aubourg, and P. Guillon, Finite element CAD for microwave filters, *Proc 28th EuMC Workshop*, Amsterdam, Netherlands, Oct. 1998, pp. 12–22.
50. P. Burrascano, S. Fiori, and M. Mongiardo, A review of artificial neural networks applications in microwave computer-aided design, *Int J RF and Microwave CAE* 9 (May 1999), 158–174.
51. S. Wang, F. Wang, V.K. Devabhaktuni, and Q.J. Zhang, A hybrid neural and circuit-based model structure for microwave modeling, *Proc 29th European Microwave Conf*, Munich, Germany, Oct. 1999, pp. 174–177.
52. M. Vai, S. Wu, B. Li, and S. Prasad, Creating neural network based microwave circuit models for analysis and synthesis, *Proc Asia Pacific Microwave Conf*, Hong Kong, Dec. 1997, pp. 853–856.
53. Y. Harkouss, J. Rousset, H. Chehade, E. Ngoya, D. Barataud, and J.P. Teyssier, The use of artificial neural networks in nonlinear microwave devices and circuits modeling: An application to telecommunication system design, *Int J RF and Microwave CAE* 9 (May 1999), 198–215.
54. Y. Fang, M.C.E. Yagoub, F. Wang, and Q.J. Zhang, A new macromodeling approach for nonlinear microwave circuits based on recurrent neural networks, *IEEE MTT-S Int Microwave Symp Dig*, Boston, MA, June 2000, pp. 883–886.
55. G. Fedi, R. Giomi, M. Giustini, R. Nesti, G. Pelosi, and S. Selleri, An artificial neural network for the analysis of profiled corrugated circular horns, *IEEE APS Int Symp Dig*, Orlando, FL, July 1999, pp. 2604–2607.
56. C. Christodoulou, A. El Zooghby, and M. Georgiopoulos, Neural network processing for adaptive array antennas, *IEEE APS Int Symp Dig*, Orlando, FL, July 1999, pp. 2584–2587.
57. R. Mishra and A. Patnaik, Neurospectral analysis of coaxial fed rectangular patch antenna, *IEEE APS Int Symp Dig*, Salt Lake City, UT, July 2000, pp. 1062–1065.
58. G. Washington, Aperture antenna shape prediction by feedforward neural networks, *IEEE Trans Antennas Propagat* 45 (1997), 683–688.
59. R.K. Mishra and A. Patnaik, Neural network based CAD model for the design of square-patch antennas, *IEEE Trans Antennas Propagat* 46 (1998), 1890–1891.
60. A.H. El Zooghby, C.G. Christodoulou, and M. Georgiopoulos, Neural network-based adaptive beamforming for one- and two-dimensional antenna arrays, *IEEE Trans Antennas Propagat* 46 (1998), 1891–1893.
61. E. Charpentier and J.J. Laurin, An implementation of a direction-finding antenna for mobile communications using a neural network, *IEEE Trans Antennas Propagat* 47 (1999), 1152–1159.
62. S. Sagiroglu, K. Guney, and M. Erler, Calculation of bandwidth for electrically thin and thick rectangular microstrip antennas with the use of multilayered perceptrons, *Int J RF and Microwave CAE* 9 (May 1999), 277–286.
63. S. El-Khamy, M. Aboul-Dahab, and K. Hijjah, Sidelobes reduction and steerable nulling of antenna arrays using neural networks, *IEEE APS Int Symp Dig*, Orlando, FL, July 1999, pp. 2600–2603.
64. G. Castaldi, V. Pierro, and I.M. Pinto, Neural net aided fault diagnostics of large antenna arrays, *IEEE APS Int Symp Dig*, Orlando, FL, July 1999, pp. 2608–2611.
65. P.R. Chang, W.H. Yang, and K.K. Chan, A neural network approach to MVDR beamforming problem, *IEEE Trans Antennas Propagat* 40 (1992), 313–322.
66. H.L. Southall, J.A. Simmers, and T.H. O’Donnell, Direction finding in phased arrays with a neural network beamformer, *IEEE Trans Antennas Propagat* 43 (1995), 1369–1374.
67. K. Watabe, K. Shimizu, K. Mizuno, and M. Yoneyama, Millimeter-wave imaging using neural networks for object recognition, *IEEE MTT-S Int Microwave Symp Dig*, San Francisco, CA, June 1996, pp. 1135–1138.
68. K. Yoshitomi, A. Ishimaru, J.N. Hwang, and J.S. Chen, Surface roughness determination using spectral correlations of scattered intensities and an artificial neural network technique, *IEEE Trans Antennas Propagat* 41 (1993), 498–502.



69. H. Yang, C. He, W. Song, and H. Zhu, Using artificial neural network approach to predict rain attenuation on earth-space path, IEEE APS Int Symp Dig, Salt Lake City, UT, July 2000, pp. 1058–1061.
70. J. Lee and A. Lai, Function-based and physics-based hybrid modular neural network for radio wave propagation modeling, IEEE APS Int Symp Dig, Salt Lake City, UT, July 2000, pp. 446–449.
71. M. Vai and S. Prasad, Automatic impedance matching with a neural network, IEEE Microwave Guided Wave Lett 3 (1993), 353–354.
72. M. Vai and S. Prasad, Microwave circuit analysis and design by a massively distributed computing network, IEEE Trans Microwave Theory Tech 43 (1995), 1087–1094.
73. M. Vai and S. Prasad, Applications of neural networks optimized by the genetic algorithm to microwave systems, IEEE APS Int Symp Dig, Orlando, FL, July 1999, pp. 2580–2583.
74. E. Bermami, S. Caorsi, and M. Raffetto, Time domain inverse scattering by means of neural networks, IEEE APS Int Symp Dig, Salt Lake City, UT, July 2000, pp. 1760–1763.
75. M.M. Vai, S. Wu, B. Li, and S. Prasad, Reverse modeling of microwave circuits with bidirectional neural network models, IEEE Trans Microwave Theory Tech 46 (1998), 1492–1494.
76. M. Vai and S. Prasad, Neural networks in microwave circuit design—Beyond black box models, Int J RF and Microwave CAE 9 (May 1999), 187–197.
77. P.M. Watson, C. Cho, and K.C. Gupta, Electromagnetic-artificial neural network model for synthesis of physical dimensions for multilayer asymmetric coupled transmission structures, Int J RF and Microwave CAE 9 (May 1999), 175–186.
78. R. Biernacki, J.W. Bandler, J. Song, and Q.J. Zhang, Efficient quadratic approximation for statistical design, IEEE Trans Circuits Syst CAS-36 (1989), 1449–1454.
79. P. Meijer, Fast and smooth highly nonlinear multidimensional table models for device modeling, IEEE Trans Circuits Syst 37 (1990), 335–346.
80. F. Wang, V.K. Devabhaktuni, C. Xi, and Q.J. Zhang, Neural network structures and training algorithms for RF and microwave applications, Int J RF and Microwave CAE 9 (May 1999), 216–240.
81. C. Xi, F. Wang, V.K. Devabhaktuni, and Q.J. Zhang, Huber optimization of neural networks: A robust training method, Int Joint Conf Neural Networks, Washington, DC, July 1999, pp. 1639–1642.
82. S.M. Sze, Physics of semiconductor devices, Wiley, New York, 1981.
83. W.R. Curtice, GaAs MESFET modeling and nonlinear CAD, IEEE Trans Microwave Theory Tech 36 (1988), 220–230.
84. M. Hirose, Y. Kitaura, and N. Uchitomi, A large-signal model of self-aligned gate GaAs FET's for high-efficiency power-amplifier design, IEEE Trans Microwave Theory Tech 47 (1999), 2375–2381.
85. D. Heo, E. Gebara, E. Chen, S. Yoo, M. Hamai, and J. Laskar, An improved deep sub-micron MOSFET RF nonlinear model with new breakdown current model and drain to substrate nonlinear coupling, IEEE MTT-S Int Microwave Symp Dig, Boston, MA, June 2000, pp. 745–748.
86. Q. Zhang, F. Wang, and V. Devabhaktuni, Neural network structures for EM/microwave modeling, IEEE APS Int Symp Dig, Orlando, FL, July 1999, pp. 2576–2579.
87. S. Haykin, Neural networks: A comprehensive foundation, Prentice Hall, Upper Saddle River, NJ, 1994.
88. D.E. Rumelhart, G.E. Hinton, and R.J. Williams, “Parallel distributed processing,” Vol. I, D.E. Rumelhart and J.L. McClelland (Editors), MIT Press, Cambridge, MA, 1986, pp. 318–362.
89. W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, Numerical recipes: The art of scientific computing, Cambridge University Press, Cambridge, MA, 1986.
90. T.R. Cuthbert Jr., in Optimization using Personal Computers, Wiley, New York, 1987, pp. 233–314.
91. A.J. Shepherd, in Second-order methods for neural networks, Springer, Berlin, 1997, pp. 43–72.
92. J.W. Bandler, W. Kellermann, and K. Madsen, A nonlinear  $l_1$  optimization algorithm for design, modeling and diagnosis of networks, IEEE Trans Circuits Syst CAS-34 (1987), 174–181.
93. P. Huber, Robust Statistics, Wiley, New York, 1981.
94. G. Li and K. Madsen, Numerical analysis, D.F. Griffiths and G.A. Watson (Editors), Pitman Research Notes in Mathematics Series 170, Longman, Harlow, Essex, UK, 1988, pp. 176–191.
95. J.W. Bandler, S.H. Chen, R.M. Biernacki, L. Gao, K. Madsen, and H. Yu, Huber optimization of circuits: A robust approach, IEEE Trans Microwave Theory Tech 41 (1993), 2279–2287.
96. J.W. Bandler, S.H. Chen, R.M. Biernacki, and K. Madsen, The Huber concept in device modeling, circuit diagnosis and design centering, IEEE Int Symp Circuit Syst Dig, London, England, 1994, pp. 129–132.
97. OSA90, Optimization Systems Associates, P.O. Box 8083, Dundas, Canada, L9H 5E7, now HP EEsof, 1400 Fountaingrove Parkway, Santa Rosa, CA 95403, 1994.
98. Q.J. Zhang, NeuroModeler, Department of Electronics, Carleton University, 1125 Colonel By Drive, Ottawa, Canada, K1S 5B6, 1999.

99. R. Reed, Pruning algorithms—A survey, *IEEE Trans Neural Networks* 4 (1993), 740–747.
100. T.Y. Kwok and D.Y. Yeung, Constructive algorithms for structure learning in feedforward neural networks for regression problems, *IEEE Trans Neural Networks* 8 (1997), 630–645.
101. A. Krzyzak and T. Linder, Radial basis function networks and complexity regularization in function learning, *IEEE Trans Neural Networks* 9 (1998), 247–256.
102. J.D. Geest, T. Dhaene, N. Fache, and D.D. Zutter, Adaptive sampling algorithm for accurate modeling of general interconnection structures, *Proc 29th European Microwave Conf, Munich, Germany, Oct. 1999*, pp. 178–181.
103. P.B.L. Meijer, Fast and smooth highly nonlinear multidimensional table models for device modeling, *IEEE Trans Circuits Syst* 37 (1990), 335–346.
104. D.C. Montgomery, *The design and analysis of experiments*, Wiley, New York, 1997.
105. U. Beyer and F. Smieja, Data exploration with reflective adaptive models, *Comp Statistics and Data Anal* 22 (1996), 193–211.
106. P.H. Ladbrooke, *MMIC Design: GaAs FET's and HEMT's*, Artech House, Boston, 1989.
107. M.A. Khatibzadeh and R.J. Trew, A large-signal, analytical model for the GaAs MESFET, *IEEE Trans Microwave Theory Tech* 36 (1988), 231–238.
108. F. Filicori, G. Vannini, and V.A. Monaco, A non-linear integral model of electron devices for HB circuit analysis, *IEEE Trans Microwave Theory Tech* 40 (1992), 1456–1465.
109. J. Aweya, Q.J. Zhang, and D.Y. Montuno, A direct adaptive neural controller for flow control in computer networks, *Proc IEEE Int Conf Neural Networks, Anchorage, Alaska, May 1998*, pp. 140–145.
110. K.S. Narendra and K. Parthasarathy, Identification and control of dynamical systems using neural networks, *IEEE Trans Neural Networks* 1 (1990), 4–27.
111. A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K.J. Lang, Phoneme recognition using time-delay neural networks, *IEEE Trans Acoust Speech Signal Processing* 37 (1989), 328–339.
112. HP-ADS, Agilent Technologies, 1400 Fountaingrove Parkway, Santa Rosa, CA 95403, 1999.

## BIOGRAPHIES



**Vijaya Kumar Devabhaktuni** (S'97) received the B.Eng. degree in Electrical and Electronics and the M.Sc. degree in Physics, both from the Birla Institute of Technology and Science, Pilani, Rajasthan, India, in 1996. He is currently a Ph.D. candidate and sessional lecturer in the Department of Electronics, Carleton University, Ottawa, Ontario, Canada. He is a two-time recipient of the Ontario Graduate Scholarship (OGS) for the academic years 1999–2000 and 2000–2001, from the Ministry of Education and Training, Ontario, Canada. He is the winner of one of the best student research exhibit prizes awarded by Nortel in 1999. His research interests include neural networks, microwave device and circuit modeling, and computer-aided design of VLSI circuits.



**Mustapha C. E. Yagoub** (M'00) received the Diplôme d'Ingénieur degree in Electronics and the *Magister* degree in Telecommunications, both from the Ecole Nationale Polytechnique, Algiers, Algeria, in 1979 and 1987 respectively. In 1994, he obtained his Ph.D. degree from the Institut National Polytechnique, Toulouse, France. He was with the Institute of Electronics of the Université des Sciences et de la Technologie Houari Boumédiène, Algiers, Algeria, first as an assistant during 1983–1991 and then as an assistant professor during 1994–1999. Starting August 1999, he has been a visit-

ing research scholar at the Department of Electronics, Carleton University, Ottawa, Canada. His research interests include neural networks, linear and nonlinear modeling, and optimization of microwave devices and circuits. He has over 50 publications in these areas in international journals and conferences. He is the co-author of *Conception de circuits linéaires et non linéaires micro-ondes* (Cépadues, Toulouse, France, 2000).



**Yonghua Fang** (S'99) received the B.Eng. degree in Computer Science and Engineering from Xi'an Jiaotong University, Xi'an, China, in 1997. He is currently an M.Eng. student in the Department of Electronics, Carleton University, Ottawa, Canada. His research interests include computer-aided design of VLSI circuits and application of artificial intelligence to microwave device and circuit modeling.



**Jianjun Xu** (S'00) received the B.Eng. degree in Electrical and Electronics Engineering from Tianjin University, Tianjin, China, in 1998. He is currently an M.Eng. student in the Department of Electronics, Carleton University, Ottawa, Canada. His research interests include neural networks, microwave modeling, and CAD of electronics circuits.



**Qi-jun Zhang** (S'84, M'87, SM'95) received the B.Eng. degree from East China Engineering Institute, Nanjing, China, in 1982, and the Ph.D. degree in Electrical Engineering from McMaster University, Hamilton, Canada, in 1987.

He was with the Systems Engineering Institute, Tianjin University, Tianjin, China, during 1982–1983. From 1988 to 1990, he was with the Optimization Systems Associates Inc. (OSA), Dundas, Ontario, Canada, developing advanced microwave optimization software. He joined the Department of Electronics, Carleton University, Ottawa, Canada, in 1990, where he is currently a professor. His research interests are neural networks and optimization methods for high-speed/high-frequency circuit design. He is a co-author of *Neural*

*Networks for RF and Microwave Design* (Artech House, Boston, 2000), a co-editor of *Modeling and Simulation of High-Speed VLSI Interconnects* (Kluwer, Boston, 1994), a contributor to *Analog Methods for Computer-Aided Analysis and Diagnosis* (Marcel Dekker, New York, 1988), a guest co-editor for a Special Issue on High-Speed VLSI Interconnects for the *International Journal of Analog Integrated Circuits and Signal Processing* (Kluwer, Boston, 1994), a guest editor for the first Special Issue on Applications of ANN to RF and Microwave Design for *the International Journal of RF and Microwave CAE* (Wiley, New York, 1999), and a guest editor for the second Special Issue on Applications of ANN to RF and Microwave Design for *the International Journal of RF and Microwave CAE* (Wiley, New York, scheduled for 2001). Dr. Zhang is a member of the Association of Professional Engineers of the Province of Ontario, (Canada).