Multidimensional signal-noise neural network model

F. Güneş
H. Torpi
F. Gürgen

Indexing terms: Neural network model, Microwave transistors, Signal parameters, Noise parameters

Abstract: Signal and noise behaviours of microwave transistors are modelled through the neural network approach for the whole operating ranges including frequency, bias and configuration types. Here, the device is modelled by a black box whose small-signal and noise parameters are evaluated through a neural network based upon the fitting of both of these parameters for multiple bias and configuration. The concurrent modelling procedure does not require the solving of device physics equations repeatedly during optimisation, and by this type of modelling the signal (S) and noise (N) parameters can be predicted not only at a single operation frequency around the chosen bias condition for a configuration, but at the same time for the whole operation frequency band for the same operating conditions, with good agreement compared to the measurements.

1 Description of the work

In this work the signal-noise neural network in [1] is extended to include bias condition (V_{DS}, I_{DS}) and configuration type (CT) so that performance parameters of the device can be generalised not only at a single operation frequency around the trained bias condition, which may be named single frequency generalisation (SFG), but at the same time in the whole operation frequency band around an untrained bias condition, which may be named as whole frequency band generalisation (WFBG). The same performance measures as [1] are also utilised for this model. The literature for the transistor modelling is given extensively in [1]. Applications of the neural networks in the microwave circuits reported in literature include automatic impedance matching [2], microstrip circuit design [3], microwave circuit analysis and optimisation [4] and, most recently, modelling of monolithic microwave integrated circuit (MMIC) passive elements [5] and simulation and optimisation of interconnects in high-speed VLSI circuits. The multiple bias-configuration signal-noise neural network is described in the second Section in detail, and worked examples and conclusion are given in the last Section.

IEE Proceedings online no. 19981712

Paper first received 28th February and in revised form 3rd October 1997

F. Güneş and H. Torpi are with the Yıldız Technical University, Electrics and Electronics Faculty, Electronics and Communication Engineering Department, 80750 Beşiktaş-Istanbul, Turkey

F. Gürgen is with Boğaziçi University, Computer Engineering Department, 80815, Bebek-Istabul, Turkey

2 Multiple bias and configuration signal-noise neural network

2.1 Structure of the network

A neural network is a simplified mathematical model of a biological neural network. It consists of a collection of interconnected neurons. Let:

$$x = [x_1 \quad x_2 \quad \cdots \quad x_n]^t \tag{1}$$

$$y = [y_1 \quad y_2 \quad \cdots \quad y_p]^t \tag{2}$$

respectively, be input and output vectors of n and p dimensions, in the signal-noise neural network, x is a 4-dimensional vector containing frequency f_k, bias condition (V_{DS}, I_{DS}) and configuration type (CT), y is a 12-dimensional vector which gives S- and N-parameters. The relationship between x and y is multidimensional and nonlinear. To model such a multidimensional nonlinear relationship, a three-layer neural network is employed, which has n processing nodes (PN) in the input layer, p PNs in the output layer, and q PNs in the hidden layer, as shown in Fig. 1. Let:

$$a^{(k)} = \left[a_1^{(k)} \quad a_2^{(k)} \quad \cdots \quad a_n^{(k)}\right]^t \tag{3}$$

$$b^{(k)} = \left[b_1^{(k)} \quad b_2^{(k)} \quad \cdots \quad b_p^{(k)}\right]^t \tag{4}$$

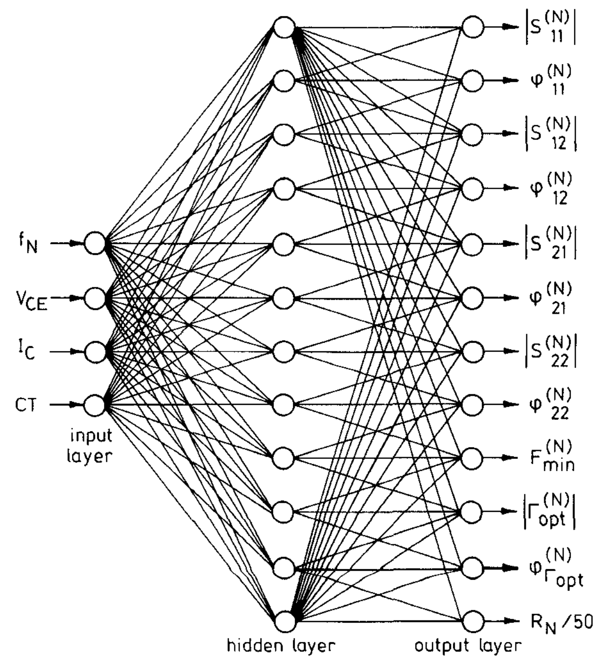

Fig. 1 *Multibias and configuration signal-noise neural network*

be vectors representing the kth sample of the input and output, respectively, $k = 1, 2, \ldots, N_s$ where N_s is the total number of the data samples. The weighting matrix between the hidden and input layer is W, and between the hidden and output layer is V, which can be expressed as:

$$W = [\,W_1 \quad W_2 \quad \cdots \quad W_h \quad \cdots \quad W_q\,] \tag{5}$$

$$V = [\,V_1 \quad V_2 \quad \cdots \quad V_h \quad \cdots \quad V_p\,] \tag{6}$$

where W_h vector is the weighting vector between the hth hidden node and the input layer:

$$W_h = [\,W_{1h} \quad W_{2h} \quad \cdots \quad W_{ih} \quad \cdots \quad W_{nh}\,]^t \tag{7}$$

V_j is the weighting vector between the jth output node and the hidden layer:

$$V_j = [\,V_{1j} \quad V_{2j} \quad \cdots \quad V_{qj}\,]^t \tag{8}$$

The signal resulting from the hidden layer to the jth output node can be expressed in the form of:

$$\Phi_j(x, W, V_j, \theta) = \sum_{h=1}^{q} V_{hj}\, g_h(x, W_h, \theta_h) \tag{9}$$

and the net output of the ith output node is obtained as follows:

$$y_j(x, W, V_j, \theta, T_j, \Theta_j) = T_j f_j(\Phi_j + \Theta_j) \tag{10}$$

where g_h and f_j are the basis functions for the hth hidden node and the jth output node, respectively, which are the sigmoid type of nonlinear functions in our application, and $g_h(x, W_h, \theta_h)$ can be expressed in the following form:

$$g_h(x, W_h, \theta_h) = \frac{1}{1 + \exp\left(-\left(\sum_{i=1}^{n} x_i W_{ih}\right) - \theta_h\right)} \tag{11}$$

In eqns. 14 and 15, Θ_j and θ_h are the thresholds of the jth output and hth hidden nodes, respectively.

2.2 Determination of the network matrix P

If parameters of the network architecture are denoted by the matrix P, the network parameter matrix P will have $n \times q + p \times q + q + p$ elements which consist of weighting factors between the input and hidden layers and the hidden and output layers, the local memories of the hidden and output nodes, respectively. The training process can be defined as computation of the network parameter matrix P so that the error function which is:

$$E(P) = \sum_{k=1}^{N_S} E^{(k)} = \sum_{k=1}^{N_S} \left[\frac{1}{2} \sum_{j=1}^{p} \left(y_j^{(k)} - b_j^{(k)} \right)^2 \right] \tag{12}$$

is minimised, where $y_j^{(k)}$ and $b_j^{(k)}$ are, respectively, the predicted and measured values of the jth output node at the training frequency f_k. This type of training process is also called backpropagation, which is available in the two types of algorithms: 'on-line' training where neural network parameters are updated after each sample presentation, and 'off-line' training where neural network parameters are updated after all samples are presented. In this work, we chose the 'on-line' training approach, since it is more efficient in most cases. The update equations for V_{hj}, W_{ih}, θ_h can be given as follows:

$$V_{hj}^{(k+1)} = V_{hj}^{(k)} - \eta \frac{\partial E^{(k)}}{\partial V_{hj}} + \alpha \left(V_{hj}^{(k)} - V_{hj}^{(k-1)} \right) \tag{13}$$

$$W_{ih}^{(k+1)} = W_{ih}^{(k)} - \eta \frac{\partial E^{(k)}}{\partial W_{ih}} + \alpha \left(W_{ih}^{(k)} - W_{ih}^{(k-1)} \right) \tag{14}$$

$$\theta_h^{(k+1)} = \theta_h^{(k)} - \eta \frac{\partial E^{(k)}}{\partial \theta_h} + \alpha \left(\theta_h^{(k)} - \theta_h^{(k-1)} \right) \tag{15}$$

and similar equations can be written for Θ_j and T_j. In eqns. 23 and 24 η and α are positive valued between 0 and 1, learning rate and momentum, respectively. In this work, a simple program is used to search their optimum values depending on the type of the worked transistor where the strategy is to determine optimum values for α and η, to ensure rapid convergence to the satisfied performance measures of the model. Thus we start with any set for the network parameters and then repeatedly change each parameter by an amount proportional to the related sensitivity terms such as

$$\frac{\partial E^{(k)}}{\partial V_{hj}}, \quad \frac{\partial E^{(k)}}{\partial W_{ih}}, \quad \frac{\partial E^{(k)}}{\partial \theta_h}$$

according to updated equations and assume that the training is completed when the error fails to decrease any further. The network parameters are then the final set of values of V_{hj}, W_{ih}, θ_h, T_j and Θ_j. By defining the sensitivities of the network with respect to V_{hj}, W_{ih}, θ_h can be given as follows, after letting Z_h and F_j be defined as:

$$Z_h = g_h(x, W_h, \theta_h) = \frac{1}{1 + \exp\left(-\left(\sum_{j=1}^{n} x_i W_{ih}\right) - \theta_h\right)} \tag{16}$$

$$F_j = f_j(\Phi_j + \Theta_j) = \frac{1}{1 + \exp(-\Phi_j - \Theta_j)} \tag{17}$$

$$\frac{\partial E^{(k)}}{\partial V_{hj}} = \frac{\partial \left[\frac{1}{2} \sum_{j=1}^{p} (y_j - b_j)^2 \right]}{\partial V_{hj}}$$

$$\frac{\partial E^{(k)}}{\partial V_{hj}} = (y_j - b_j) T_j \frac{\partial f_f}{\partial \Phi_j} \frac{\partial \Phi_j}{\partial V_{hj}}$$

$$\frac{\partial E^{(k)}}{\partial V_{hj}} = (y_j - b_j) T_j F_j (1 - F_j) Z_h = \delta_j^{(3)} Z_h \tag{18}$$

$$\frac{\partial E^{(k)}}{\partial W_{ih}} = \sum_{j=1}^{p} \frac{\partial E^{(k)}}{\partial y_j} \frac{dy_j}{dF_j} \frac{dF_j}{d\Phi_j} \frac{\partial \Phi_j}{\partial Z_h} \frac{\partial Z_h}{\partial W_{ih}}$$

$$\frac{\partial E^{(k)}}{\partial W_{ih}} = \left[\sum_{j=1}^{p} (y_j - b_j) T_j F_j (1 - F_j) V_{hj} \right] Z_h (1 - Z_h) a_i^{(k)}$$

$$\frac{\partial E^{(k)}}{\partial W_{ih}} = \sum_{j=1}^{N_0} \delta_j^{(3)} V_{hj} Z_h (1 - Z_h) a_i^{(k)} = \delta_h^{(2)} a_i^{(k)} \tag{19}$$

$$\frac{\partial E^{(k)}}{\partial \theta_h} = \sum_{j=1}^{p} \frac{\partial E^{(k)}}{\partial y_j} \frac{dy_j}{dF_j} \frac{dF_j}{d\Phi_j} \frac{\partial \Phi_j}{\partial Z_h} \frac{\partial Z_h}{\partial \theta_h}$$

$$\frac{\partial E^{(k)}}{\partial \theta_h} = \sum_{j=1}^{p} (y_j - b_j) T_j F_j (1 - F_j) V_{hj} Z_h (1 - Z_h)$$

$$\frac{\partial E^{(k)}}{\partial \theta_h} = \sum_{j=1}^{N_0} \delta_j^{(3)} V_{hj} Z_h (1 - Z_h) = \delta_h^{(2)} \tag{20}$$

$$\frac{\partial E^{(k)}}{\partial T_j} = \frac{\partial E^{(k)}}{\partial y_j^{(k)}} \frac{\partial y_j^{(k)}}{\partial T_j} = (y_j - b_j)F_j$$

$$\frac{\partial E^{(k)}}{\partial T_j} = \delta_j^{(3)} T_j^{-1}(1 - F_j)^{-1} \quad (21)$$

$$\frac{\partial E^{(k)}}{\partial \Theta_j} = \frac{\partial E^{(k)}}{\partial y_j} \frac{\partial y_j}{\partial F_j} \frac{\partial F_j}{\partial \Theta_j} = (y_j - b_j)T_j F_j(1 - F_j)$$

$$\frac{\partial E^{(k)}}{\partial \Theta_j} = \delta_j^{(3)} \quad (22)$$

where $\delta_h^{(2)}$ and $\delta_j^{(3)}$ represent local gradients at individual nodes in the second and third layers, respectively, and using (9.3) and (9.5) they can be expressed as:

$$\delta_h^{(2)} = \sum_{j=1}^{N} \delta_j^{(3)} V_{hj} Z_h (1 - Z_h) \quad (23)$$

$$\delta_j^{(3)} = (y_j - b_j)T_j F_j (1 - F_j) \quad (24)$$

3 Worked examples and conclusion

In the foregoing Sections a neural-based technique is described for modelling of signal and noise behaviour of microwave transistors over their operation frequency band, biasing ranges and configuration types. In the work, this multiple bias-configuration neural network model has been applied to many transistors for which the manufacturer's characterisation data is used as the training data. The results show that the predicted parameters are generally in good agreement with the desired parameters.

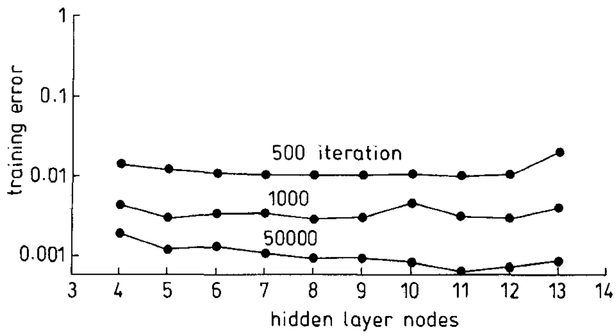

Fig. 2 *Variation of neural network with training error*
Iteration number is taken as parameter

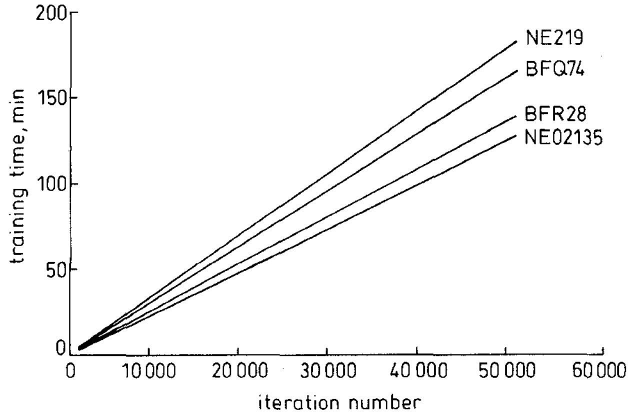

Fig. 3 *Variation of training time with respect to iteration number*

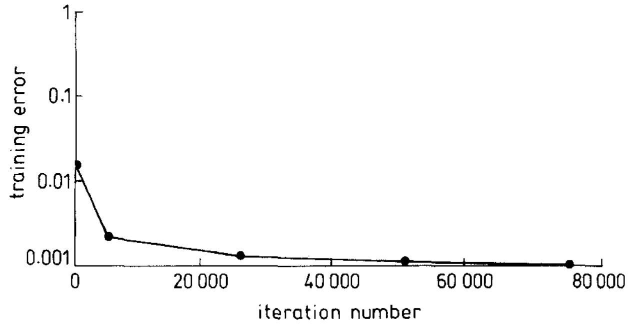

Fig. 4 *Variation of training error with respect to iteration number for NE02135*

In the following subsections, two worked examples are presented (Tables 1 and 2) with the performance measures of the neural network models, where amount of the data used for the training is optimised for each bias condition against the error and iteration number. A worked example is also given in [1]. Fig. 2 gives variations of the hidden nodes of the neural network against the training error, the iteration number taken as the parameter, in our model, which is taken to be equal to the number of output nodes, which is almost optimum. The graphs in Fig. 2 also show how rapid the training error converges, since the error settles down in the range of 0,01 within 500 iterations. Figs. 3 and 4 give variation of the training time with respect to the iteration number for the various types of transistor,

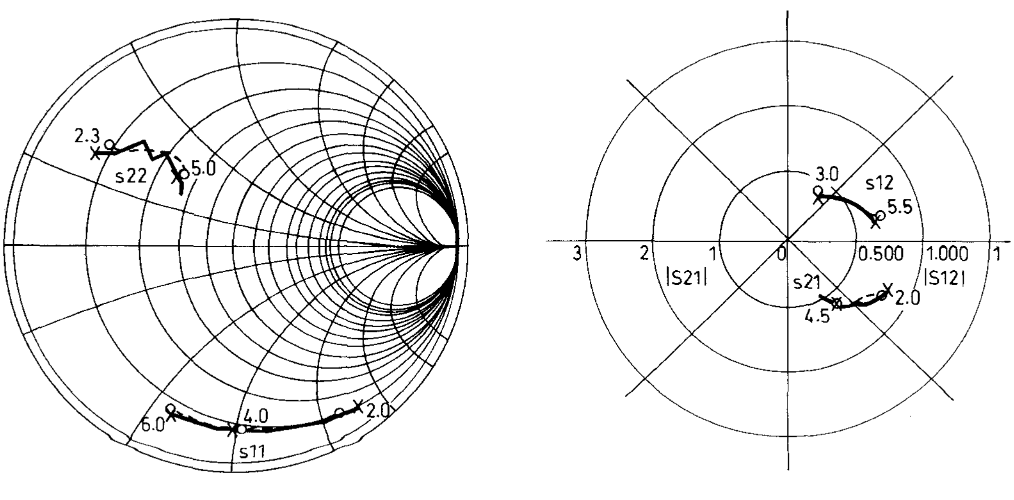

Fig. 5 *S-parameter–frequency variations for V_{CE} = 8 V, I_C = 20 mA at the common collector configuration of the transistor NE219 (WFBG)*
E_{11} = 0.019730, E_{21} = 0.035988, E_{12} = 0.049394, E_{22} = 0.071714, E_t = 0.044207

Table 1: Multiple bias and configuration signal neural network model for the transistor NE219 common collector and common emitter configuration

Operation frequency bandwidth: 2–6GHz

Operation bias conditions

Configuration	Bias voltage V_{CE}, V	Bias currents I_C, mA		
	8	10	20	30
		SFG	WFBG	SFG
CC		E_t = 0.050105	E_t = 0.044207	E_t = 0.032068
		E_{11} = 0.027857	E_{11} = 0.019730	E_{11} = 0.016128
		E_{21} = 0.037930	E_{21} = 0.035988	E_{21} = 0.036308
		E_{12} = 0.037051	E_{12} = 0.049394	E_{12} = 0.029213
		E_{22} = 0.097582	E_{22} = 0.071714	E_{22} = 0.046625
		Fig. 7	Fig. 5	not given
CE		E_t = 0.73243	E_t = 0.065140	E_t = 0.058195
		E_{11} = 0.045472	E_{11} = 0.039773	E_{11} = 0.028880
		E_{21} = 0.034876	E_{21} = 0.044730	E_{21} = 0.032119
		E_{12} = 0.061781	E_{12} = 0.060726	E_{12} = 0.042287
		E_{22} = 0.150743	E_{22} = 0.115331	E_{22} = 0.0129495
		not given	not given	Fig. 8

Table 2: Multiple bias and signal-noise neural network model for the transistor NE02135 at the common emitter configuration

Operation frequency bandwidth: 0.1–4GHz

Operation bias conditions

Bias voltage V_{CE}, V	Bias currents I_C, mA			
10.0	5.0	10.0	20.0	30.0
	SFG	WFBG	SFG	SFG
	E_t = 0.041287	E_t = 0.079736	E_t = 0.037003	E_t = 0.044565
	E_{11} = 0.033850	E_{11} = 0.034377	E_{11} = 0.023597	E_{11} = 0.018405
	E_{21} = 0.044067	E_{21} = 0.056279	E_{21} = 0.043421	E_{21} = 0.049997
	E_{12} = 0.043720	E_{12} = 0.097377	E_{12} = 0.046626	E_{12} = 0.067227
	E_{22} = 0.043512	E_{22} = 0.130912	E_{22} = 0.034370	E_{22} = 0.042631
	F_1 = 0.030823	F_1 = NMDA	F_1 = 0.023949	F_1 = 0.01918
	F_2 = 0.102794	F_2 = NMDA	F_2 = 0.052824	F_2 = 0.07685
	F_3 = 0.123793	F_3 = NMDA	F_3 = 0.096364	F_3 = 0.0659
	F_t = 0.085804	F_t = NMDA	F_t = 0.057712	F_t = 0.0539
	Fig. 9	Fig. 11	not given	Fig. 12

Extrapolated values of noise parameter for V_{CE} = 10V, I_C = 30mA are used. NMDA = no measured data available

where it should be noted that, once the neural network has been trained, it responses at once to the desired input. The generalisation process can be considered in two categories.

3.1 Single frequency generalisation (SFG)

This can be defined as the performance parameter prediction at a single operation frequency of the trained bias condition. In the work [1], the interpolation and extrapolation capabilities of the network are demonstrated in the error-frequency planes over a fairly large operation bandwidth at a single bias condition. In this work, the SFG procedure is applied to a lot of bias conditions for the different configuration types, and the same conclusion as [1] is reached, which is that the network has a high capability to interpolate between the data points used for learning.

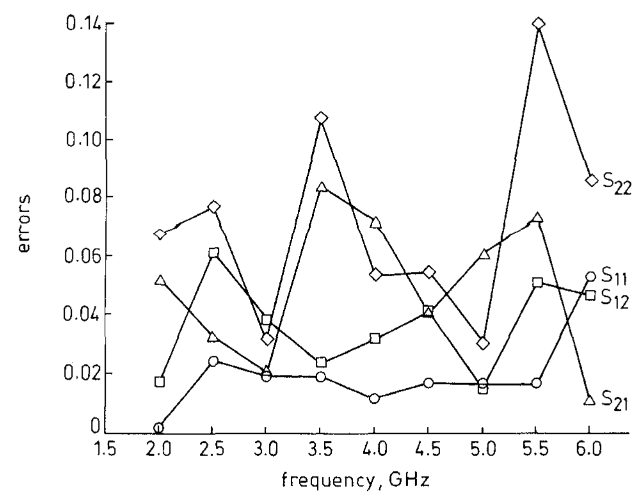

Fig. 6 *Error-frequency distributions for V_{CE} = 8V, I_C = 20mA at the common collector configuration of the transistor NE219 (WFBG)*

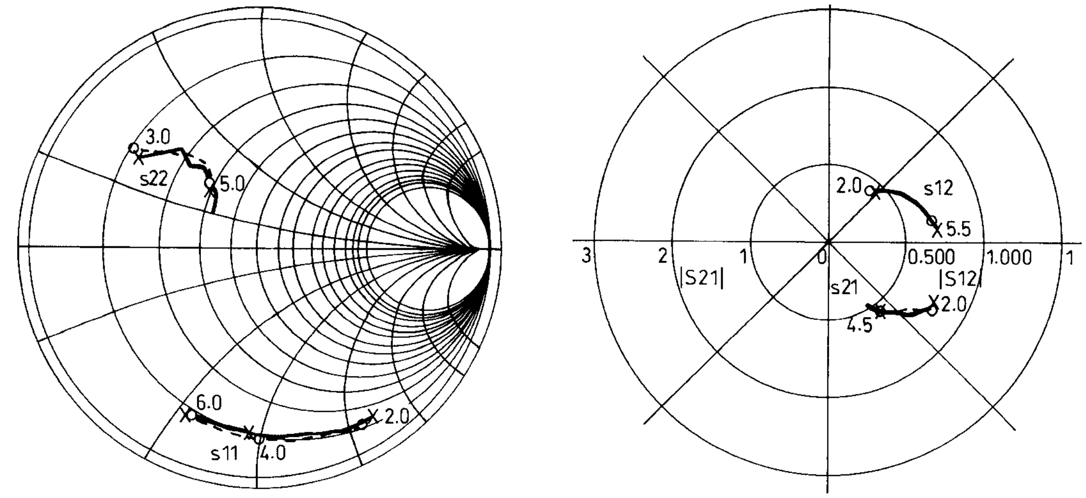

Fig. 7 *S-parameter–frequency variations for V_{CE} = 8.0 V, I_C = 10 mA at the common collector configuration of transistor NE219*
E_{11} = 0.027857, E_{21} = 0.037930, E_{12} = 0.037051, E_{22} = 0.097582, E_t = 0.050105

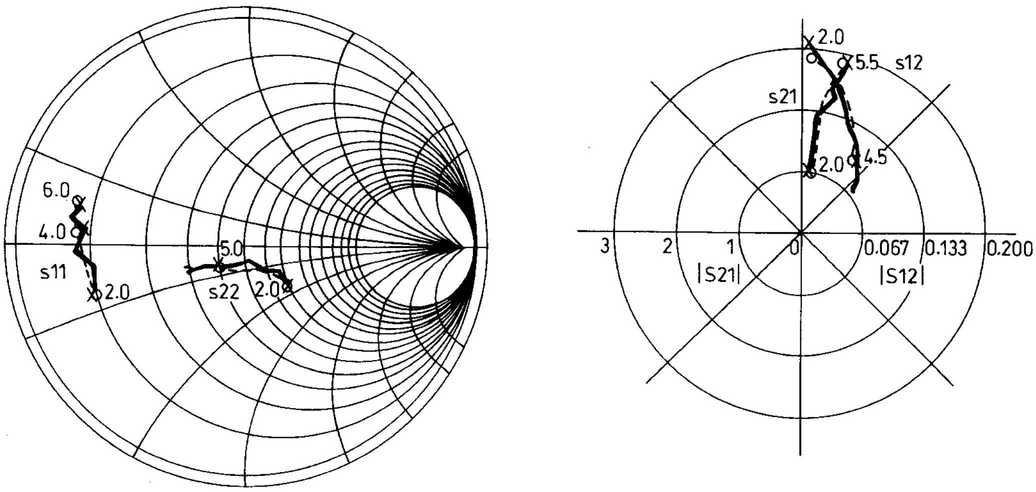

Fig. 8 *S-parameter–frequency variations for V_{CE} = 8 V, I_C = 30 mA at common emitter configuration of transistor NE219*
E_{11} = 0.028880 E_{21} = 0.032119 E_{12} = 0.042287 E_{22} = 0.129495 E_t = 0.058195

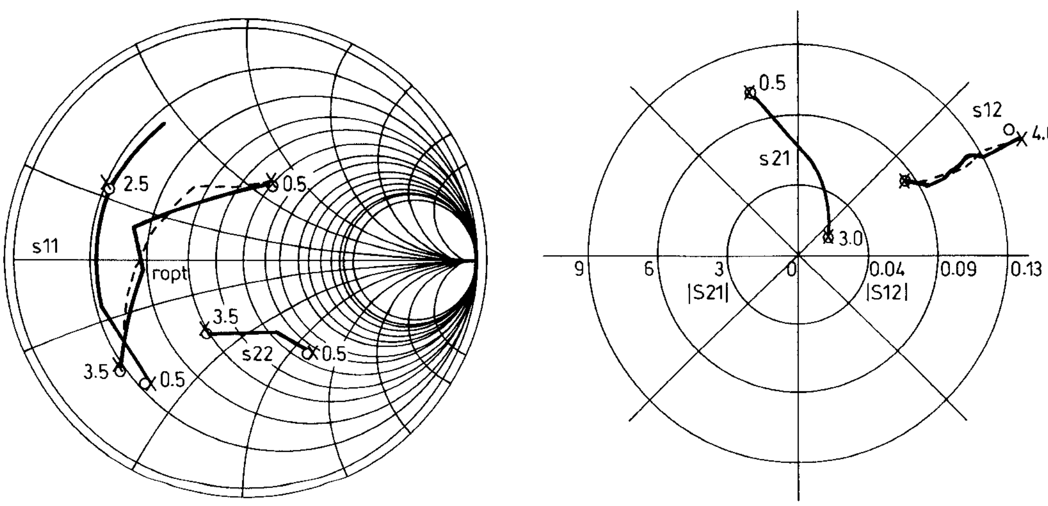

Fig. 9 *S- and N-parameter–frequency variation for V_{CE} = 10 V, I_C = 5 mA at the common emitter configuration of transistor NE02135 (SFG)*

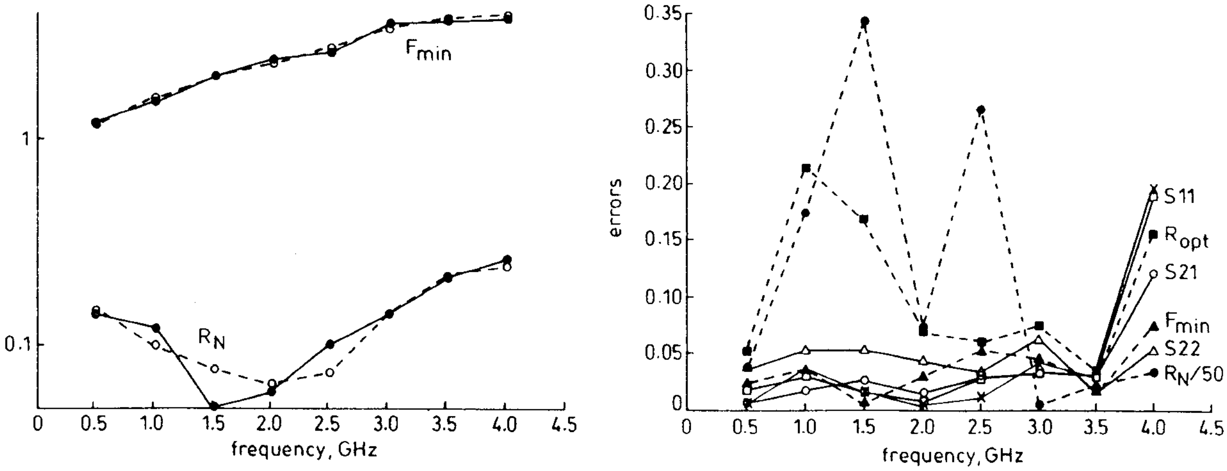

Fig. 10 *Error-frequency distributions for V_{CE} = 10 V, I_C = 5mA at the common emitter configuration of transistor NE02135 (SFG)*

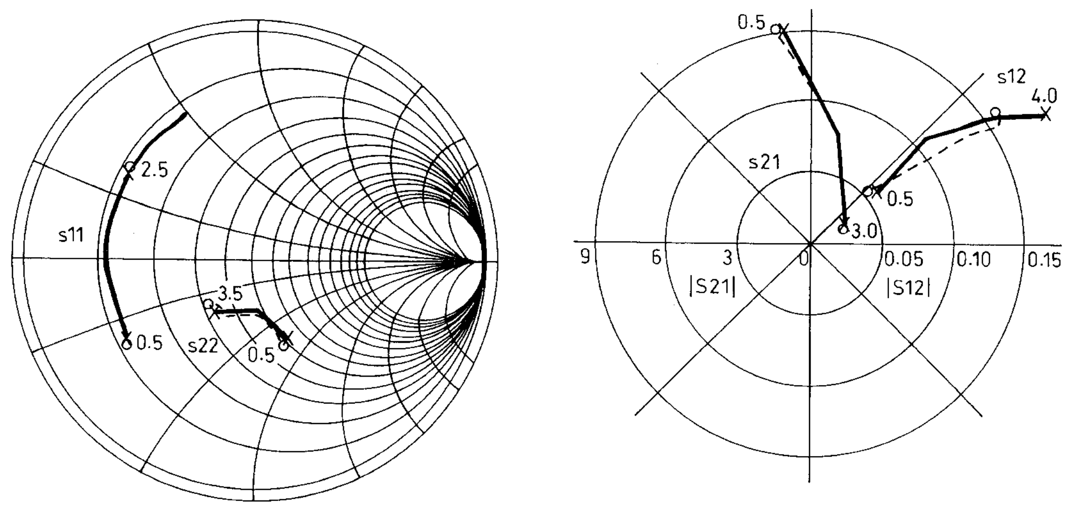

Fig. 11 *S-parameter–frequency variations for V_{CE} = 10 V, I_C = 10mA at the common emitter configuration of transistor NE02135 (WFBG)*

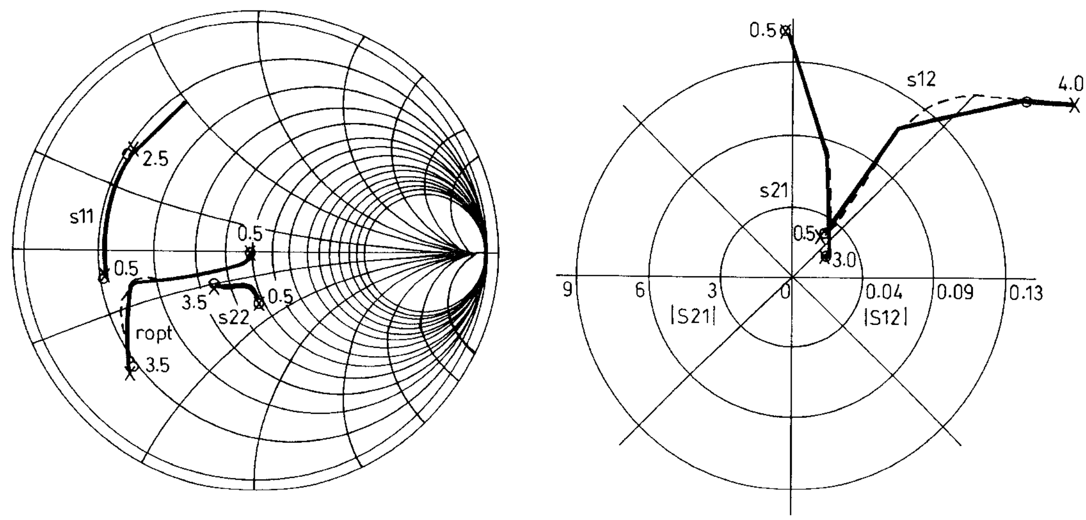

Fig. 12 *S-parameter–frequency variations for V_{CE} = 10 V, I_C = 30mA at the common emitter configuration of transistor NE02135 (SFG)*

3.2 Whole frequency band generalisation (WFBG)

This can be defined as prediction of the unknown multidimensional performance parameter function for the whole operation frequency band, which has eight signal dimensions and four noise dimensions given by eqn. 2, at an untrained bias condition. Fig. 6 gives the frequency distribution of error for the whole frequency band generalisation (WFBG) of the scattering parameters at the bias condition V_{CE} = 8 V, I_C = 10mA for the common collector configuration given in Fig. 5. From the examination of the WFBG processes, one can conclude that the network can predict the multidimensional performance parameter functions of the transistor between the bias conditions.

In the Figs. 7–12 some predicted and target variations of the signal and noise parameters for the transistors NE219 and NE02135 at the multiple bias-configuration are given between the starting and stopping frequencies supplied by the manufacturers, for the purpose of comparison. In all the Figures, the curves with dashed lines give the predicted variations, while the solid curves are the target variations.

In this work, we have presented a nontraditional approach to modelling of microwave transistors, so that a multiple bias/configuration signal-noise neural network model is obtained, which is shown to be capable of giving results in a good agreement with the measured values. By exploiting the flexibility and generality of the neural network model, we have demonstrated its use for the device characterisation for its signal and noise performance over its whole operation ranges including frequency, bias conditions of the possible configuration types. At the same time, results of our work have demonstrated the feasibility and efficiency of using neural networks in the practical interactive design and optimisation methods for the physics-based device modelling.

4 References

1 GÜNEŞ, F., GÜRGEN, F., and TORPI, H.: 'Signal-noise neural network model for active microwave devices', *IEE Proc., Circuit Devices Syst.*, 1996, **143**, (1), pp. 1–8

2 MANKUAN, V., and SHEILA, P.: 'Automatic impedance matching with a neural network', *IEEE Microwave Guided Wave Lett.*, 1993, **3**, (10), pp. 353–354

3 HORNG, T., WANG, C., and ALEXOPOULOS, N.G.: 'Microstrip circuit design using neural networks', *MTT-S Int. Microwave Symp. Dig.*, 1993, pp. 413–416

4 ZAABAB, A.H., ZHANG, Q.J., and NAKHLA, M.: 'A neural network modeling approach to circuit optimisation and statistical design', *IEEE Trans. Microw. Theory Tech.*, 1995, **43**, (6), pp. 1349–1358

5 CREECH, G.L., PAUL, J.B., LESNIAK, D.C., JENKINS, J.T., and CALCETERA, C.M.: 'Artificial neural networks for fast and accurate EM-CAD of microwave circuits', *IEEE Trans. Microw. Theory Tech.*, 1997, **45**, (5), pp. 794–802

6 VELUSWAMI, A., NAKHLA, S.M., and ZHANG, Q.J.: 'The applications of neural networks to EM-based simulation and optimisation of interconnects in high-speed VLSI circuits', *IEEE Trans. Microw. Theory Tech.*, 1997, **45**, (1), pp. 712–723