micromachines

MDPI

Article

A Time Delay Neural Network Based Technique for Nonlinear Microwave Device Modeling

Wenyuan Liu [1], Lin Zhu [2], Feng Feng [3], Wei Zhang [3,*], Qi-Jun Zhang [3], Qian Lin [4] and Gaohua Liu [5]

[1] School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an 710021, China; liuwenyuan@sust.edu.cn
[2] School of Control and Mechanical Engineering, Tianjin Chengjian University, Tianjin 300384, China; zeal286@163.com
[3] Department of Electronics, Carleton University, Ottawa, ON K1S 5B6, Canada; fengfeng@doe.carleton.ca (F.F.); qjz@doe.carleton.ca (Q.-J.Z.)
[4] College of Physics and Electronic Information Engineer, Qinghai University for Nationalities, Xining 810007, China; linqian@tju.edu.cn
[5] School of Electronics and Information Engineering, Tianjin University, Tianjin 300072, China; suppig@126.com
* Correspondence: weizhang13@doe.carleton.ca

Received: 15 July 2020; Accepted: 28 August 2020; Published: 31 August 2020

Abstract: This paper presents a nonlinear microwave device modeling technique that is based on time delay neural network (TDNN). The proposed technique can accurately model the nonlinear microwave devices when compared to static neural network modeling method. A new formulation is developed to allow for the proposed TDNN model to be trained with DC, small-signal, and large signal data, which can enhance the generalization of the device model. An algorithm is formulated to train the proposed TDNN model efficiently. This proposed technique is verified by GaAs metal-semiconductor-field-effect transistor (MESFET), and GaAs high-electron mobility transistor (HEMT) examples. These two examples demonstrate that the proposed TDNN is an efficient and valid approach for modeling various types of nonlinear microwave devices.

Keywords: nonlinear device modeling; neural networks; optimization methods

1. Introduction

Artificial neural network (ANN) is a recognized tool for modeling and design optimization in RF and microwave computer-aided design (CAD) [1–9]. This technique has been successfully used in parametric modeling of microwave components [10–12], electromagnetic (EM) optimization [13,14], parasitic modeling [15], nonlinear device modeling [16–18], nonlinear microwave circuit optimization [19–22], power amplifier modeling [23–25], and more.

This paper addresses the nonlinear device modeling area. Nonlinear device modeling is an important area of CAD and a variety of device models have been built. With the rapid development of semiconductor industry, new devices constantly evolve. The existing models may not be accurate for the new devices. Therefore, there is an ongoing need for new models. The challenge for CAD researchers is not only to develop new models, but also to introduce new CAD methods.

Traditionally, the equivalent circuit modeling approach is a vital modeling technique for nonlinear device modeling. The existing equivalent circuit models need to be modified in order to fit for different devices. The parameters in the equivalent circuit need repetitively changes and sometimes the parameters are mutually contradictory. Especially, when it comes to a new device, it is time consuming to build a nonlinear model that is based on equivalent modeling technique. For an alternative approach, the ANN model can be an efficient trained and implemented model [1], which can be systematically developed by neural network training process. ANN technique recently can be used to approach device modeling problem with good accuracy. When the nonlinear and dynamic effects in the device become significant, we need more advanced neural network-based techniques to approach device modeling problem. Several different types of neural networks, such as dynamic neural networks (DNNs), real-valued time-delay neural networks (RVTDNNs), and recurrent neural networks (RNNs), have been used for nonlinear circuit modeling [20–22]. These neural network-based techniques are more flexible to build more general models. Recently, the dynamic neuro-space mapping technique [26] can also deal with the nonlinear device modeling problem well, which contains coarse model and neural network. However, building a proper coarse model needs repetitive changes of the parameters in equivalent circuits.

In this paper, we focus on directly modeling methods that can systematically establish models without building proper equivalent circuit models. In this paper, we propose a time delay neural network (TDNN) technique for nonlinear microwave device modeling using DC, small-signal, and large-signal information for the first time. A new formulation to train the proposed TDNN with DC, small-signal, and large signal data is proposed. An algorithm to train the proposed TDNN model is formulated. Examples of GaAs metal-semiconductor-field-effect transistor (MESFET) and GaAs high-electron mobility transistor (HEMT) modeling is used to demonstrate the validity of the proposed TDNN method.

2. Formulations of the Proposed Time Delay Neural Network (TDNN) Model

According to a nonlinear device, $\boldsymbol{u} = \begin{bmatrix} u_1 & u_2 & \dots & u_m \end{bmatrix}^T$ represents the vector of the input signals, while $\boldsymbol{o} = \begin{bmatrix} o_1 & o_2 & \dots & o_{N_o} \end{bmatrix}^T$ are the output signals, where m is the number of the inputs and N_o is the number of outputs. For example, $\boldsymbol{u} = \begin{bmatrix} v_g & v_d \end{bmatrix}^T$ and $\boldsymbol{o} = \begin{bmatrix} i_g & i_d \end{bmatrix}^T$ in the transistor example, where $m = 2$ and $N_o = 2$. In this example, v_g and v_d represent the gate voltage and the drain voltage, respectively. i_g and i_d are the gate current and the drain current, respectively. Let $\boldsymbol{f}_{ANN}$ represent the multilayer neural network. $\boldsymbol{w}$ represents the internal weight of the neural network. The general TDNN equation in time domain can be used in order to describe the original nonlinear device as

$$\boldsymbol{o}(t) = \boldsymbol{f}_{ANN}(\boldsymbol{u}(t), \boldsymbol{u}(t-\tau), \dots, \boldsymbol{u}(t-N_d\tau), \boldsymbol{w}) \tag{1}$$

where τ is a time delay parameter and N_d represents the total number of delay steps.

Suppose that the TDNN model contains one input and one output and the $\boldsymbol{f}_{ANN}$ is a three-layer multilayer perceptron (MLP) model. Therefore, Figure 1 shows the TDNN structure. In this figure, the TDNN structure contains external delay information compared with MLP model.

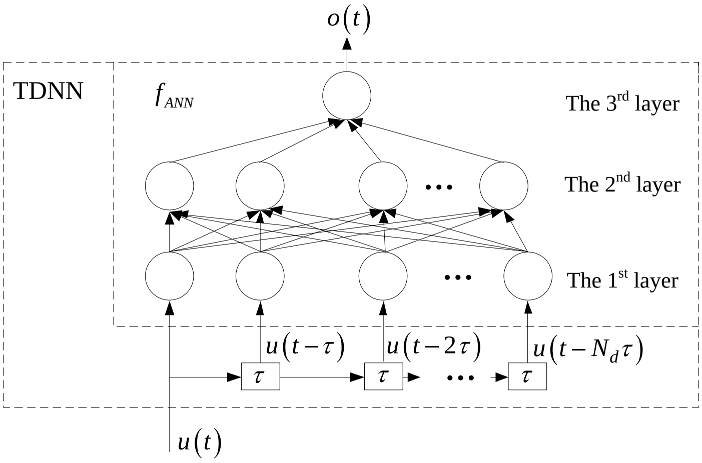


Figure 1. The structure for time delay neural network (TDNN).

In this paper, the f_{ANN} of the TDNN is a three-layer MLP. The first layer of the MLP is the input relay layer, the second layer is the hidden layer, and the third layer is the output layer. The sigmoid function is used as the activation function in the internal hidden layer.

After the neural network well trained by the device data, the TDNN model can be a good model. We can usually get DC, S-parameters, and harmonic data for nonlinear device modeling from measurement or simulation. Therefore, we propose an analytical formulation of TDNN for nonlinear device modeling using DC, bias-dependent S-parameter data, and large-signal harmonic balance (HB) data.

Let $\boldsymbol{U}$ represent the DC input signals and $\boldsymbol{O}$ be DC output signals. Therefore, the delayed signals of inputs in DC condition are all equal to $\boldsymbol{U}$. The output of TDNN in DC case is derived as

$$\boldsymbol{O} = f_{ANN}\left(\overbrace{\boldsymbol{U},\ \boldsymbol{U},\ \ldots,\ \boldsymbol{U}}^{N_d+1},\ \boldsymbol{w}\right) \tag{2}$$

Let $\boldsymbol{Y}$ represent the small signal transfer function of the system. In transistor example, matrix $\boldsymbol{Y}$ represents Y-parameters. Let $\boldsymbol{U}_{bias}$ denote the DC bias of $\boldsymbol{u}$. The small-signal S-parameters are derived through the Y-parameters of the TDNN model that are shown in Equation (3). In Equation (3), the derivative of $\boldsymbol{f}_{ANN}$ can be obtained using the adjoint neural network method [27], and k represents the index of delay buffers. The $\boldsymbol{Y}$ matrix, defined as the sum of products of $e^{-j\omega k\tau}$ and $\partial \boldsymbol{f}_{ANN}/\partial \boldsymbol{u}$ in (3), is frequency dependent due to the use of delayed signal in output function $\boldsymbol{f}_{ANN}$. Hence, the proposed TDNN model is a non-quasi static (NQS) model [28–31], when $N_d > 0$. In Equation (3), $j\omega = j2\pi f$, where f represents frequency.

$$\boldsymbol{Y} = \left(\sum_{k=0}^{N_d} e^{-j\omega k\tau} \cdot \left.\frac{\partial \boldsymbol{f}_{ANN}^{T}(\boldsymbol{u}(t),\ \boldsymbol{u}(t-\tau),\ \ldots,\ \boldsymbol{u}(t-N_d\tau),\ \boldsymbol{w})}{\partial \boldsymbol{u}(t-k\tau)}\right|_{\boldsymbol{u}(t)=\ \boldsymbol{u}(t-\tau)=\ \ldots=\boldsymbol{u}(t-N_d\tau)=\boldsymbol{U}_{bias}}\right)^{T} \tag{3}$$

In the large-signal case, suppose the generic harmonic frequency be ω_k, where the subscript k represents the index of harmonic frequency $k = 0, 1, 2, \ldots, N_H$. N_H is the number of harmonics that are considered in HB simulation. N_T represents the number of time points. Let $W_N(n, k)$ denote the Fourier coefficient for nth time sample and the kth harmonic frequency, where n = 1,2, … ,N_T and k = 1,2, … ,N_H. Let superscript* represent complex conjugate. Let $\boldsymbol{U}(\omega_k)$ and $\boldsymbol{O}(\omega_k)$ be the input and output signals in the frequency domain, respectively. Given input $\boldsymbol{U}(\omega_k)$ for all k, $\boldsymbol{u}(t_n - K_\tau)$ can be computed from Equation (4), where K = 0, 1, 2, … , N_d. The outputs $O(\omega_k)$ are computed as in Equation (5). The frequency domain delay functions $e^{-j\omega_k\tau}$, $e^{-j\omega_k 2\tau}$, … ,$e^{-j\omega_k N_d\tau}$ are introduced into the training equation. The proposed technique can accurately model the nonlinear behavior of the device by training the TDNN model with DC, S-parameter, and HB data.

$$\boldsymbol{u}(t_n - K\tau) = \sum_{k=0}^{N_H} \boldsymbol{U}(\omega_k) W_N^{*}(n,k) e^{-j\omega_k K\tau} \tag{4}$$

$$\boldsymbol{O}(\omega_k) \ = \frac{1}{N_T} \sum_{n=0}^{N_T-1} f_{ANN}(u(t_n),\ \ u(t_n-\tau)\ , \ldots, u(t-N_d\tau)) \ \cdot W_N(n,k) \tag{5}$$

We systematically described above the TDNN model equation used in DC, small-signal, and large-signal simulation. Because of the neural network universal approximation capability [1], such TDNN model can achieve satisfied accuracy.

3. An Algorithm for Training the Proposed TDNN Model

Our proposed TDNN model will be good after the neural networks being well trained by DC, S-parameters, and HB data of the nonlinear device. The training error is formulated as

$$\begin{aligned} E_{Tr}(w) &= \alpha E_{DC}(w) + \beta E_S(w) + \gamma E_{HB}(w) \\ &= \alpha\left(\frac{1}{2}\sum_{k\in T}\|\boldsymbol{O}(\boldsymbol{x}_k, w) - \boldsymbol{O}_k^{(d)}\|^2\right) + \beta\left(\frac{1}{2}\sum_{k\in T}\|\boldsymbol{S}(\boldsymbol{x}_k, w) - \boldsymbol{S}_k^{(d)}\|^2\right) + \gamma\left(\frac{1}{2}\sum_{k\in T}\|\boldsymbol{HB}(\boldsymbol{x}_k, w) - \boldsymbol{HB}_k^{(d)}\|^2\right) \end{aligned} \tag{6}$$

where E_{Tr} represents the total training error, E_{DC} represents the error between DC responses of the proposed TDNN model and the DC device data, E_S represents the error between small-signal responses of the proposed TDNN model and the small-signal device data, and E_{HB} represents the error between large-signal responses of the proposed TDNN model and the large-signal device data. α, β and γ represent the weighting factors for DC error E_{DC}, small-signal error E_S, and large-signal error E_{HB}, respectively. The weighting factors α, β and γ can be roughly determined by the value range of the training data and the number of DC data, small-signal data, and large-signal harmonic data. $\boldsymbol{O}(.)$, $\boldsymbol{S}(.)$ and $\boldsymbol{HB}(.)$ represent the DC, bias-dependent S-parameters and HB response of the proposed TDNN model, respectively. $\boldsymbol{O}_k^{(d)}$, $\boldsymbol{S}_k^{(d)}$ and $\boldsymbol{HB}_k^{(d)}$ represent the kth training data of DC, bias-dependent S-parameters and HB, respectively. T represents of training sets. We use real and imaginary types of the HB data for training in the proposed TDNN technique.

The first step for developing the proposed TDNN model is to generate DC, small-signal and large-signal device data used for training and testing. The range of the training data should cover the range of the testing data. After data preparation, we have to determine the structure of the proposed TDNN model, including the number of delay buffers and the number of hidden neurons. After these preparation works, we begin to train the proposed TDNN model. In the beginning, the number of delays buffers can be tried from 1, i.e., $N_d = 1$ and the hidden neurons can be tried with a smaller number. We first set α and β as constant that are roughly decided by the value range of the training data and the number of DC data, and small-signal data, and set γ equals 0. The proposed TDNN model can be trained with DC and small-signal data by adjust the neural network weights according to the error back propagation algorithm. After the first step training (it may need hundreds or thousands times of iteration, which is according to the practical problem), α, β and γ will be set as constants. Subsequently, the proposed TDNN model can be trained combined with DC, small-signal S-parameters, and large-signal harmonic data. After this step training, the training error will be calculated. When it is less than E_t (user defined error criteria), the process of the training will stop. After the overall training, a separate set of DC, small-signal and large-signal data called test data, which are never used in the training, is used to test the quality of the proposed TDNN model. The test error E_{Te} is defined as the error between the model responses and the test data. If the test error is also lower than the threshold error E_t, then the model training process terminates and the proposed TDNN model is ready to be used for high-level design. Otherwise, the overall model training process will go to the previous step being repeated with different numbers of hidden neurons or different numbers of delay buffers. Figure 2 shows the flowchart illustrating the overall development process of the proposed TDNN model.

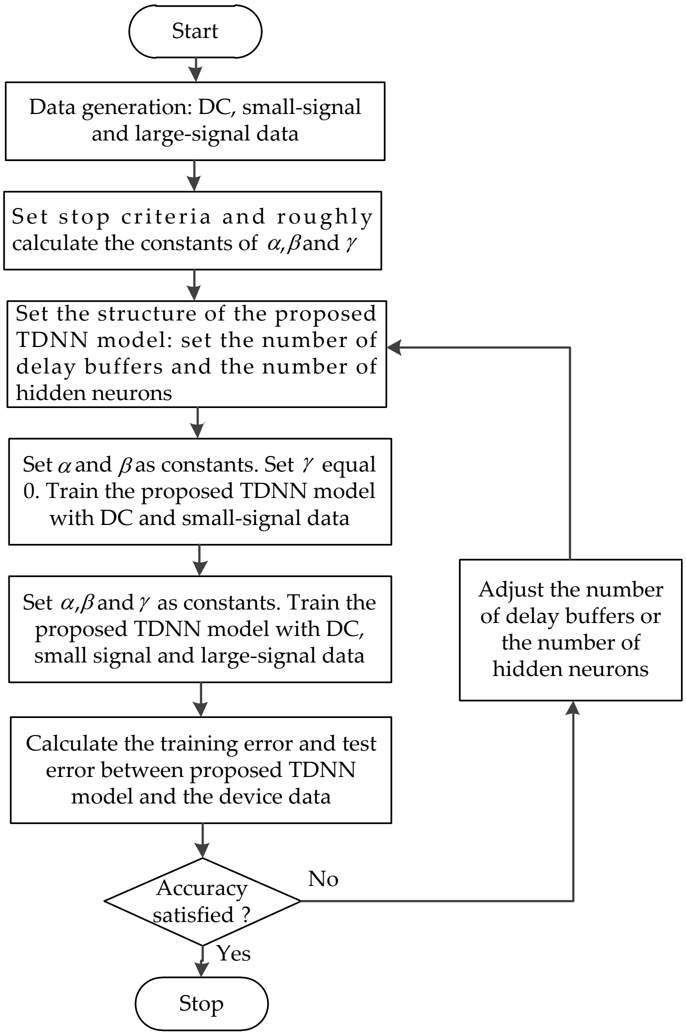

Start
Data generation: DC, small-signal and large-signal data
Set stop criteria and roughly calculate the constants of α,$\beta$and γ
Set the structure of the proposed TDNN model: set the number of delay buffers and the number of hidden neurons
Set $\alpha$and βas constants. Set γ equal 0. Train the proposed TDNN model with DC and small-signal data
Set α,$\beta$and γ as constants. Train the proposed TDNN model with DC, small signal and large-signal data
Calculate the training error and test error between proposed TDNN model and the device data
Accuracy satisfied ?
No
Yes
Adjust the number of delay buffers or the number of hidden neurons
Stop

Figure 2. The process for the proposed TDNN model development.

4. Examples

4.1. GaAs Metal-Semiconductor-Field-Effect Transistor (MESFET)

In this example, the TDNN method is used to model a Keysight advanced design system (ADS) [32] internal GaAs MESFET device [33] with the Statz model. Table 1 shows the parameters for the Statz model in ADS. We perform DC, small-signal, and large-signal training together in NeuroModelerPlus [34]. Training data includes DC data at different DC points, S-parameter data at different biases and large-signal harmonic data generated at different fundamental frequencies (1–6 GHz), input power levels (−5–7 dBm), and loads (40–60 Ohm), as seen in Table 2. The training data set and test data set are not randomly divided shown in Table 2. There are DC data at 162 different DC points, bias-dependent S-parameter data at 120 different biases, and harmonic data at a total of 936 combinations of input power, fundamental frequency, and load for training data. There are DC data at 130 different DC points, bias-dependent S-parameter data at 95 different biases, and harmonic data at a total of 120 combinations of input power, fundamental frequency, and load for test data. All of the training data was generated in ADS after performing DC simulation, S-parameter simulation, and harmonic balance simulation for getting DC, S-parameter, and harmonic data, respectively. The range of V_g and V_d in DC case can cover the range of V_g and V_d in small-signal S-parameter and harmonic cases. The frequency range of S-parameter data can cover the frequency range of harmonic data which is calculated by the fundamental frequency with the number of harmonics considered in the harmonic modeling process. The range of test data is within the range of training data. In this example, we choose the time delay parameter of the TDNN as 0.0045 ns. We perform the training for the proposed TDNN technique according to part 3. The proposed TDNN model is built after nearly 3000 iterations of DC and small-signal training and 300 iterations of DC, small-signal, and large-signal training. It takes roughly 1.5 h with the Intel core i9-9900 CPU at 3.60 GHz of the computing system. When the training is finished, we compare the accuracy of the proposed TDNN model at different training conditions shown in Table 3.

Table 1. Parameters for Statz model.

Parameter Name	Vaule	Parameter Name	Vaule
Cgs (F)	9.581×10^{13}	*Lambda* (1/V)	0.05
Cgd (F)	7.598×10^{14}	*Alpha* (1/V)	3.0
Cds (F)	1×10^{14}	*B* (none)	3.0
Crf (F)	1×10^{14}	*Rgd* (Ohm)	3
Vto (V)	0.5	*Rg* (Ohm)	1
Beta (A/V^2)	0.310	*Rd* (Ohm)	5
Vbi (V)	0.9	*Rs* (Ohm)	2

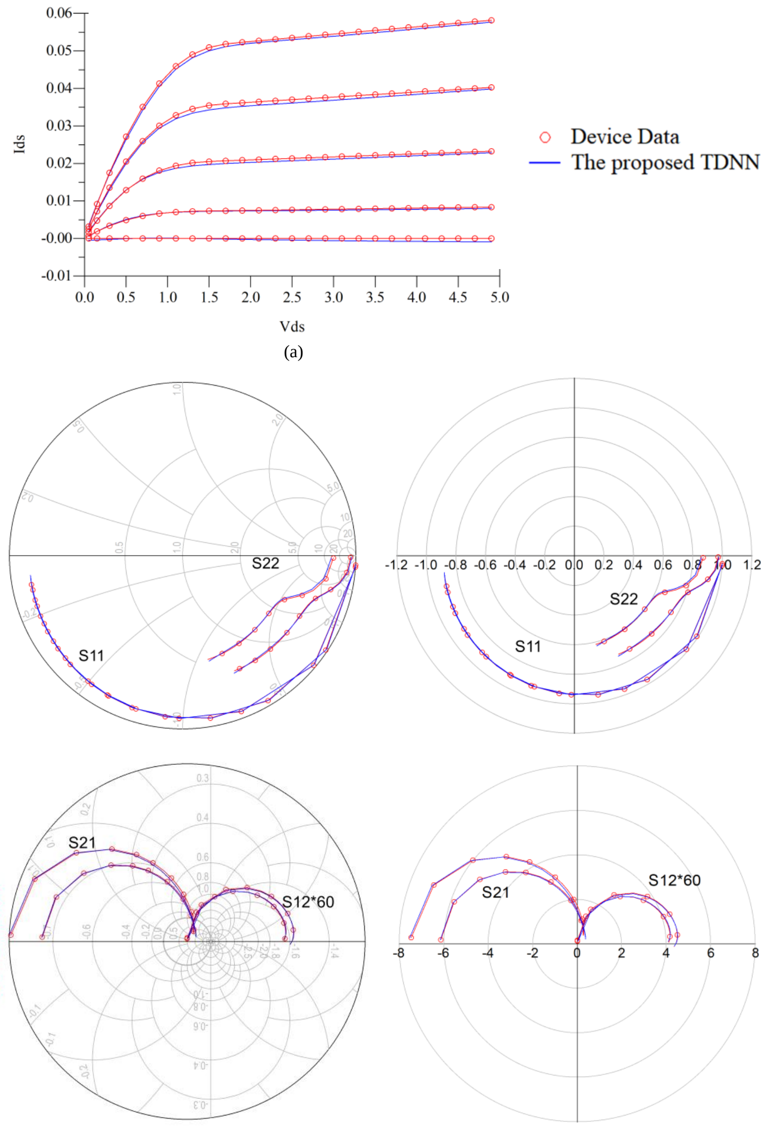
Table 2. Training data and test data for GaAs metal-semiconductor-field-effect transistor (MESFET).

Data Type	Parameter Name	Training Data			Test Data		
		Min	Max	Step	Min	Max	Step
DC data	V_g (V)	−0.6	0.4	0.2	−0.5	0.3	0.2
	V_d (V)	0 0.4	0.2 5	0.1 0.2	0.05 0.3	0.15 4.9	0.1 0.2
Small-signal data	V_g (V)	−0.6	0.4	0.2	−0.5	0.3	0.2
	V_d (V)	0 0.4 2.6	0.2 2.2 5	0.1 0.2 0.4	0.05 0.3 2.4	0.15 2.1 4.8	0.1 0.2 0.4
	f (GHz)	0.1	40.1	1	0.1	40.1	1
Large-signal data	V_g (V)	−0.2	−0.1	0.1	−0.15	−0.15	0
	V_d (V)	3.0	3.2	0.2	3.1	3.1	0
	Pin (dBm)	−5	7	1	−4.5	6.5	1
	freq (GHz)	1	6	1	1.5	5.5	1
	Load (Ohm)	40	60	10	45	55	10

Table 3. Accuracy comparison of two modeling approach at different conditions.

Approach	Training	Test
MLP	53.99%	51.14%
TDNN (N_d = 1)	6.16%	6.22%
TDNN (N_d = 2)	3.59%	2.75%
TDNN (N_d = 3)	2.95%	2.11%
TDNN (N_d = 4)	2.38%	1.88%

For comparison purpose, we also developed the static model using the MLP technique for this GaAs MESFET example. MLP is a feedforward neural network. The inputs of the MLP and TDNN both are V_g and V_d of the transistor, the outputs of the MLP and TDNN are both I_g and I_d of the transistor. For fairly comparison, we both use a three-layer MLP for MLP technique and the f_{ANN} of the TDNN technique, the activation functions are both the sigmoid function, the numbers of hidden neurons for these two techniques are both same, and the learning algorithm used in this paper is quasi-newton method. We compare the results from the MLP model and the proposed TDNN model that is shown in Table 3. In the case of DC, S-parameter at multiple biases, and HB training, the TDNN approach has accuracy advances over the static modeling technique, as seen in Table 3. This is because nonlinear devices usually contain dynamic effects, which is not adequate for device modeling by using the static modeling technique (MLP). However, when compared with MLP (only contains the present information), the proposed TDNN includes not only present information, but also the history information, which is necessary for nonlinear device modeling, especially when nonlinear device contains dynamic effects. When the number of delay buffers increases, the error of the proposed TDNN model when compared with device data decreases rapidly. We choose the condition (N_d = 4, training error = 2.38%, test error = 1.88%) in Table 3 to present the results of our proposed TDNN model. DC, S-parameters, and HB responses of the proposed TDNN model are shown in Figures 3 and 4. Finally, in this proposed TDNN model, the number of hidden neurons is 40, time delay parameter of the TDNN is 0.0045 ns, and the number of delay buffers is 4. From these figures, we can see the proposed TDNN model can accurately model the nonlinear microwave device.



(a)

(b)

Figure 3. For GaAs metal-semiconductor-field-effect transistor (MESFET) example, comparison of DC and S-parameters at multiple biases of the device data and the proposed TDNN model. (**a**) DC. (**b**) S-parameters at two test biases of (−0.3 V, 3.6 V) and (0.1 V, 2.1 V). The DC and S-parameters shown in the figure from proposed TDNN is test data which is never used in the training. The frequency range of S-parameters for this MESFET example is from 0.1 GHz to 40.1 GHz.

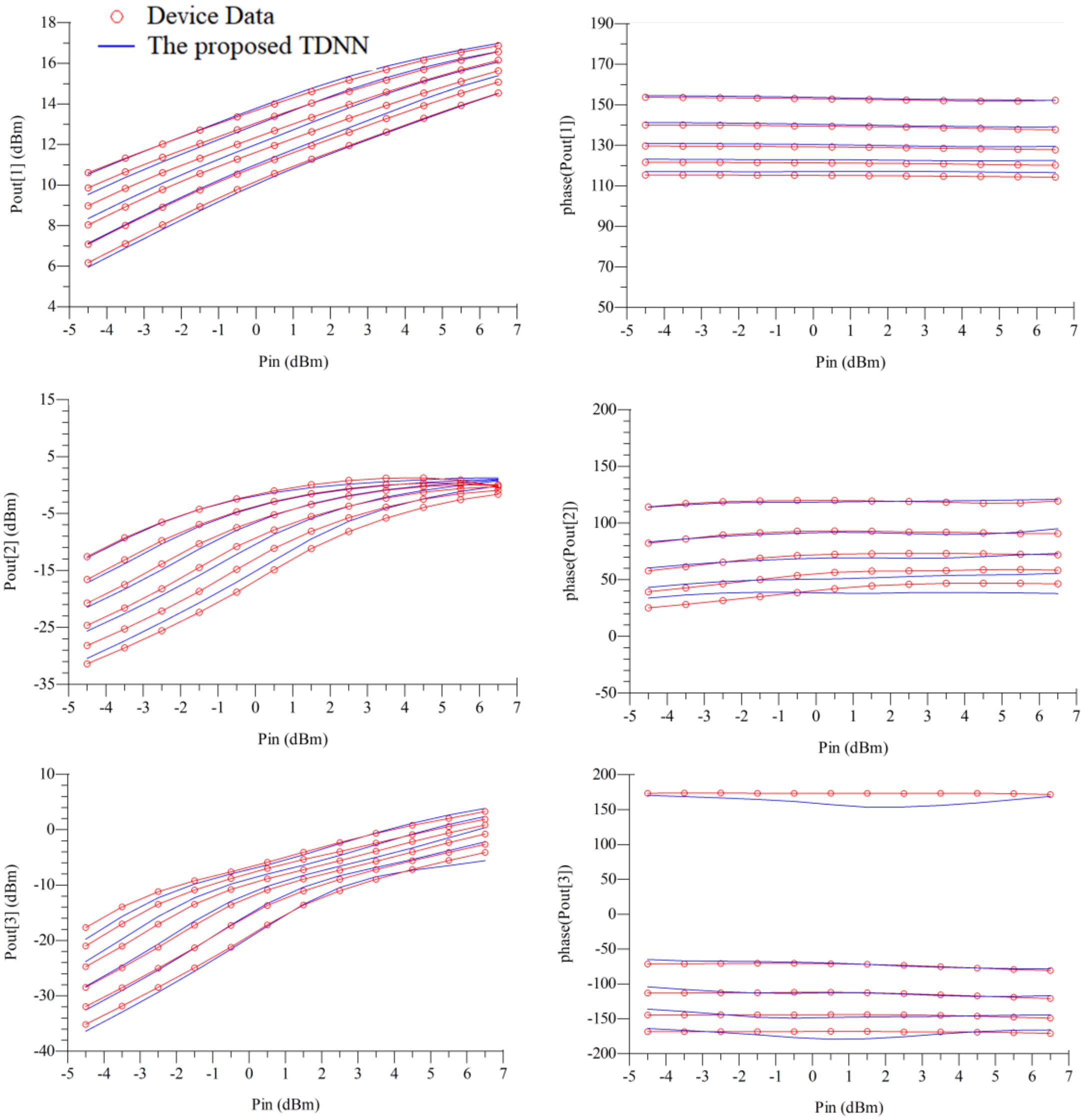


Figure 4. Comparison of the harmonic balance (HB) responses between the proposed TDNN model and the device data at the test load: 45 Ω, the fundamental frequency points: from 1.5 to 5.5 GHz, test bias: (V_g: −0.15 V, V_d: 3.1 V), and input power levels: from −4.5 to 6.5 dBm in the MESFET example.

4.2. GaAs High-Electron Mobility Transistor (HEMT)

In this example, the proposed TDNN method is used to model the GaAs HEMT device. Training and test data were generated from a five-layer GaAs-AlGaAs-InGaAs HEMT example given in a physics-based device simulator Medici [35]. The structure of the HEMT [36] used in setting up the physical-based simulator is shown in Figure 5. Table 4 shows the parameters for the HEMT device. We performed DC, small-signal, and large signal training of the proposed TDNN according to the algorithm in part 3 with NeuroModelerPlus [34]. Training data includes DC data at different DC points, S-parameter data at different biases and large-signal harmonic data generated at different fundamental frequencies (2–5 GHz) and input power levels (−20–10 dBm), as seen in Table 5. The static bias is chosen as: V_g: 0.2 V and V_d: 5 V. The training data set and test data set are not randomly divided shown in Table 5. There are DC data at 378 different DC points, bias-dependent S-parameter data at 138 different biases, and harmonic data at a total of 44 combinations of input power, fundamental frequency, and load for training data. There are DC data at 310 different DC points, bias-dependent S-parameter data at 110 different biases, and harmonic data at a total of 33 combinations of input power, fundamental frequency, and load for test data. All of the training data was generated in Medici after performing DC simulation, S-parameter simulation, and harmonic balance simulation for getting DC, S-parameter, and harmonic data, respectively. The range of V_g and V_d in DC case can cover the range of V_g and V_d in small-signal S-parameter and harmonic cases. The frequency range of S-parameter data can cover the frequency range of harmonic data, which is calculated by the fundamental frequency with the number of harmonics considered in the harmonic modeling process. The range of test data is within the range of training data. In this example, we choose time delay parameter of the TDNN as 0.005 ns. After nearly 2000–3000 iterations of DC and small-signal training and 300 iterations of DC, small-signal, and large-signal training, the proposed TDNN model is built. It takes roughly 1.5–2 h with the Intel core i9-9900 CPU at 3.60 GHz of the computing system. When the training is finished, we compare the accuracy of the proposed TDNN model at different training conditions shown in Table 6.

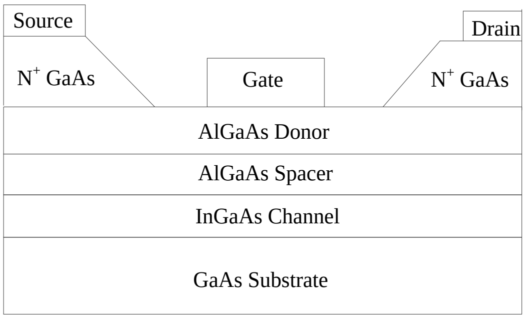


Figure 5. The structure of the high-electron mobility transistor (HEMT) device in Medici simulator used for data generation.

Table 4. Values of geometrical/physical parameters for high-electron mobility transistor (HEMT) device.

Parameter Name		**Value (um)**
Gate Length (um)		0.2
Gate Width (um)		100
Thickness (um)	AlGaAs Donor Layer	0.025
	AlGaAs Spacer Layer	0.01
	InGaAs Channel Layer	0.01
	GaAs Substrate	0.045
Doping Density (1/cm^3)	AlGaAs Donor Layer	1×10^{18}
	InGaAs Channel Layer	1×10^{2}
	Source N+	2×10^{20}
	Drain N+	2×10^{20}

Table 5. Training data and test data for GaAs HEMT.

Data Type	Parameter Name	Training Data			Test Data		
		Min	Max	Step	Min	Max	Step
DC data	V_g (V)	−0.2	0.8	0.2	−0.1	0.7	0.2
	V_d (V)	0	6.2	0.1	0.05	6.15	0.1
Small-signal data	V_g (V)	−0.2	0.8	0.2	−0.1	0.7	0.2
	V_d (V)	0 0.4 2.6	0.2 2.2 6.2	0.1 0.2 0.4	0.05 0.3 2.4	0.15 2.1 6.0	0.1 0.2 0.4
	f (GHz)	0.1	40.1	1	0.1	40.1	1
Large-signal data	Pin (dBm)	−20 −3	−5 9	5 2	−20 −3	−5 9	5 2
	freq (GHz)	2	5	1	2.5	4.5	1

Table 6. Accuracy comparison from different training conditions.

Approach	30 Hidden Neurons		40 Hidden Neurons	
	Training Error	Test Error	Training Error	Test Error
MLP	31.13%	33.98%	33.07%	34.11%
TDNN (N_d = 1)	6.41%	6.58%	6.24%	6.48%
TDNN (N_d = 3)	3.10%	3.32%	2.68%	2.88%
TDNN (N_d = 5)	2.44%	2.51%	2.16%	2.24%
TDNN (N_d = 7)	1.49%	1.86%	1.15%	1.9%

For comparison purpose, we have also developed MLP model for this GaAs HEMT example. The inputs of the MLP and TDNN are V_g and Vd of the transistor, the outputs of the MLP and TDNN are I_g and I_d of the transistor. For fairly comparison, we both use a three-layer MLP for MLP technique and the f_{ANN} of the TDNN technique, the activation functions are both the sigmoid function, the numbers of hidden neurons for these two techniques both are same, and the learning algorithm used in this paper is quasi-Newton method. In the complicated case, DC, S-parameter at multiple biases and HB training together, TDNN model has huge accuracy advantage over MLP model, as seen in Table 6. In this table, the error of the TDNN model compared with test data reduces as the number of delay buffers increases. When comparing the number of hidden neurons 30 and 40, we can see as the number of hidden neurons increases, the accuracy enhances slowly. We choose the condition (N_d = 7, 40 hidden neurons, training error = 1.15%, and test error = 1.9%) in Table 6 in order to present the results of our proposed TDNN model. The DC, S-parameters and HB responses of the proposed TDNN model are shown in Figures 6 and 7. Finally, in the proposed TDNN model for this GaAs HEMT example, the number of hidden neurons is 40, the time delay parameter of the TDNN is 0.005 and the number of delay buffers is 7. From these figures, we can see that the proposed TDNN technique can accurately model the GaAs HEMT example.

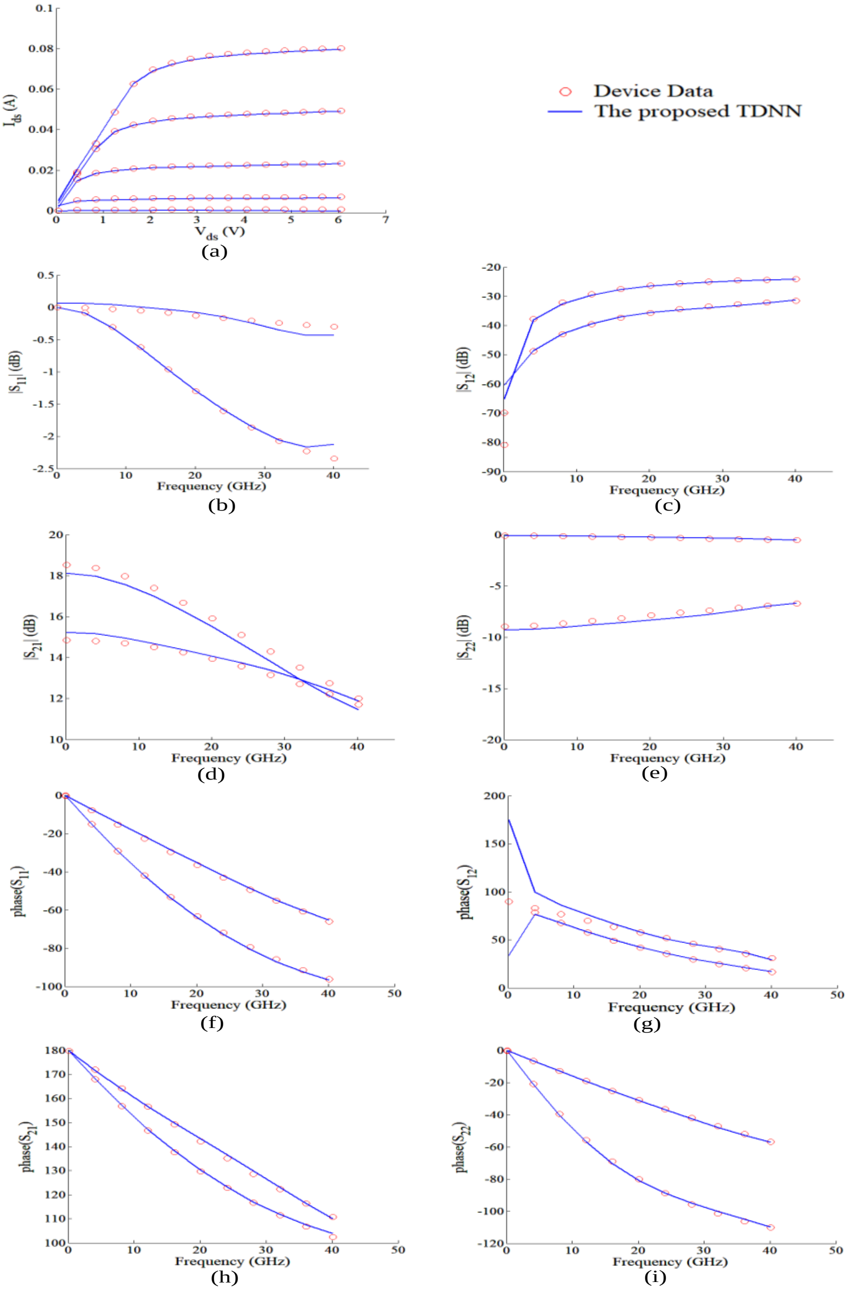


Figure 6. Comparison between the proposed TDNN model and the device data using DC and S-parameters at multiple biases for the HEMT example. (**a**) DC. (**b**–**i**) Magnitudes and Phases of S-parameters at two test biases of (V_g and V_d) at (0.1 V, 5.6 V) and (0.7 V, 2.1 V). The DC and S-parameters shown in the figure from proposed TDNN is test data which is never used in the training.

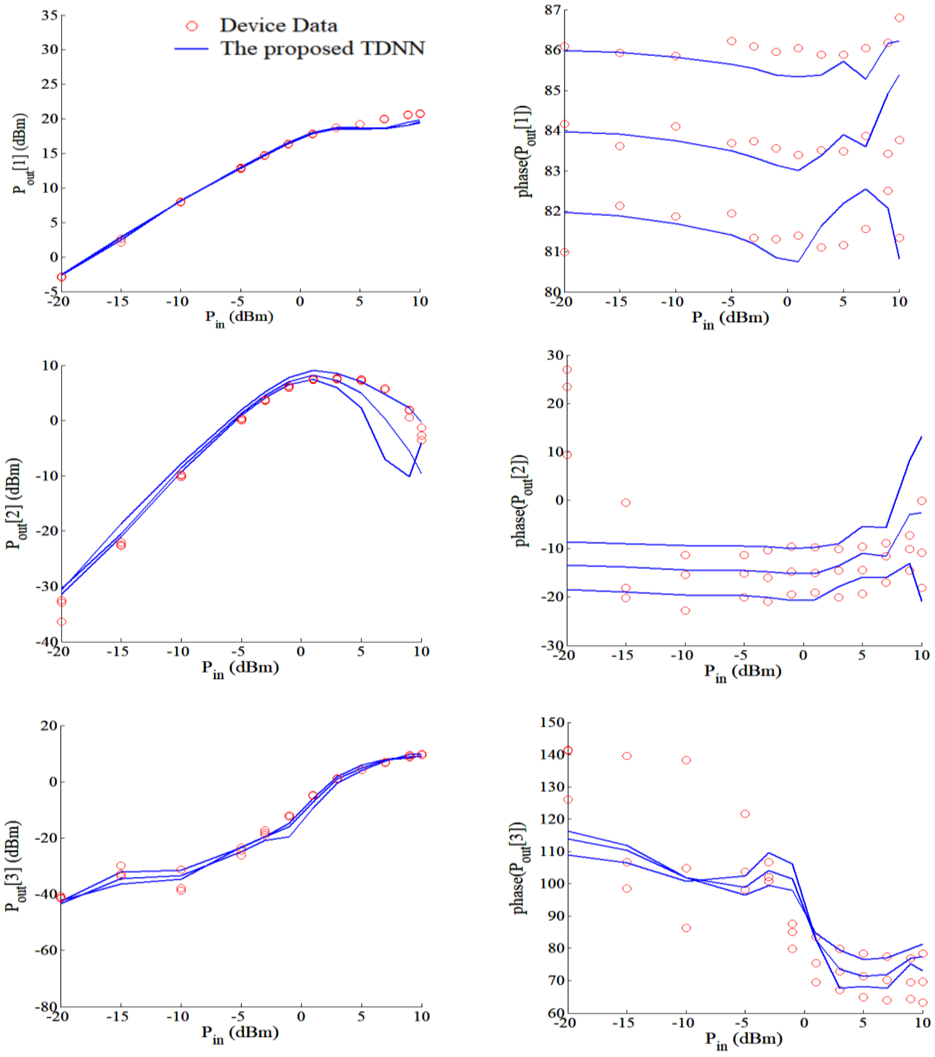


Figure 7. Comparison of magnitude and phase responses of the proposed TDNN model and device data at fundamental frequency 2.5 GHz, 3.5 GHz, and 4.5 GHz. Each blue line represents the magnitude or phase of output power along the input power at different fundamental frequencies.

5. Conclusions

In this paper, we have proposed a TDNN based technique for nonlinear microwave devices modeling. We have proposed a set of new formulations for training with DC, small-signal and large-signal data. We also have proposed an algorithm for the proposed TDNN model development. The modeling of GaAs MESFET and GaAs HEMT examples has successfully demonstrated that the TDNN based technique can accurately build nonlinear microwave device models. Using measurements to validate the comparison with real situation could be a useful direction. In the future direction, the thermal and trapping effects can be combined into the proposed TDNN. In the future, conventional device modeling method as compared with proposed TDNN can be a useful direction. As a potential future direction, the proposed TDNN technique can be investigated for other semiconductor technologies, such as Si and GaN based FETs. In the future, modeling and design microwave absorbers by the proposed technique can also be investigated.

Author Contributions: Conceptualization, W.L., L.Z. and Q.-J.Z.; methodology, W.L., L.Z. and F.F.; validation and writing—original draft preparation, W.L.; writing—review and editing, W.L., L.Z., W.Z. and F.F.; supervision, W.Z., F.F. and Q.-J.Z.; funding acquisition, Q.L. and G.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Basic Research Plan of Shaanxi Province (Grant 2020JQ-732), the National Natural Science Foundation of Tianjin (Grant no. 18JCQNJC00900), the National Natural Science Foundation of China (Grant no. 61601323 and no. 61841110), the Scientific Research Project of Tianjin Education Commission (Grant no. 2017KJ088 and 2016CJ13), and the Research Foundation of Shaanxi University of Science and Technology (Grant no. 2018BJ-01).

Acknowledgments: The authors thank the technical support from Zhihao Zhao, Tianjin University and Carleton University, and Weicong Na, Beijing University of Technology.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, Q.J.; Gupta, K.C.; Devabhaktuni, V.K. Artificial neural networks for RF and microwave design—from theory to practice. *IEEE Trans. Microw. Theory Tech.* **2003**, *51*, 1339–1350. [CrossRef]
2. Feng, F.; Na, W.; Liu, W.; Yan, S.; Zhu, L.; Ma, J.; Zhang, Q.-J. Multifeature-assisted neuro-transfer function surrogate-based EM optimization exploiting trust-region algorithms for microwave filter design. *IEEE Trans. Microw. Theory Tech.* **2020**, *68*, 531–542. [CrossRef]
3. Na, W.; Liu, W.; Zhu, L.; Feng, F.; Ma, J.; Zhang, Q.J. Advanced extrapolation technique for neural-based microwave modeling and design. *IEEE Trans. Microw. Theory Tech.* **2018**, *66*, 4397–4418. [CrossRef]
4. Kabir, H.; Zhang, L.; Kim, K. Automatic parametric model development technique for RFIC inductors with large modeling space. In Proceedings of the 2017 IEEE MTT-S International Microwave Symposium (IMS), Honololu, HI, USA, 4–9 June 2017; pp. 551–554.
5. Jin, J.; Zhang, C.; Feng, F.; Na, W.; Ma, J.; Zhang, Q. Deep Neural Network Technique for High-Dimensional Microwave Modeling and Applications to Parameter Extraction of Microwave Filters. *IEEE Trans. Microw. Theory Tech.* **2018**, *67*, 4140–4155. [CrossRef]
6. Du, X.; Helaoui, M.; Jarndal, A.; Liu, T.; Hu, B.; Hu, X.; Ghannouchi, F.M. ANN-based large-signal model of AlGaN/GaN HEMTs with accurate buffer-related trapping effects characterization. *IEEE Trans. Microw. Theory Tech.* **2020**, *68*, 3090–3099. [CrossRef]
7. Zhao, P.; Wu, K. Homotopy optimization of microwave and millimeter-wave filters based on neural network model. *IEEE Trans. Microw. Theory Tech.* **2020**, *68*, 1390–1400. [CrossRef]
8. Xiao, L.; Shao, W.; Jin, F.; Wang, B.; Joines, W.T.; Liu, Q.H. Semisupervised radial basis function neural network with an effective sampling strategy. *IEEE Trans. Microw. Theory Tech.* **2020**, *68*, 1260–1269. [CrossRef]
9. Li, S.; Wang, Y.; Yu, M.; Panariello, A. Efficient modeling of Ku-band high power dielectric resonator filter with applications of neural networks. *IEEE Trans. Microw. Theory Tech.* **2019**, *67*, 3427–3435. [CrossRef]
10. Feng, F.; Gongal, V.M.R.; Zhang, C.; Ma, J.G.; Zhang, Q.J. Parametric modeling of microwave components using adjoint neural networks and pole-residue transfer functions with EM sensitivity analysis. *IEEE Trans. Microw. Theory Tech.* **2017**, *65*, 1955–1975. [CrossRef]
11. Zhang, W.; Feng, F.; Gongal, V.M.R.; Zhang, J.; Yan, S.; Ma, J.; Zhang, Q.-J. Space Mapping Approach to Electromagnetic Centric Multiphysics Parametric Modeling of Microwave Components. *IEEE Trans. Microw. Theory. Tech.* **2018**, *66*, 3169–3185. [CrossRef]
12. Feng, F.; Na, W.; Liu, W.; Yan, S.; Zhu, L.; Zhang, Q. Parallel gradient-based EM optimization for microwave components using adjoint-sensitivity-based neuro-transfer function surrogate. *IEEE Trans. Microw. Theory Tech.* **2020**. [CrossRef]
13. Zhang, C.; Feng, F.; Gongal-Reddy, V.-M.-R.; Zhang, Q.J.; Bandler, J.W. Cognition-driven formulation of space mapping for equal-ripple optimization of microwave filters. *IEEE Trans. Microw. Theory Tech.* **2015**, *63*, 2154–2165. [CrossRef]
14. Zhang, J.; Zhang, C.; Feng, F.; Zhang, W.; Ma, J.; Zhang, Q. Polynomial Chaos-based approach to yield-driven EM optimization. *IEEE Trans. Microw. Theory Tech.* **2018**, *66*, 3186–3199. [CrossRef]
15. Sen, P.; Woods, W.H.; Sarkar, S.; Pratap, R.J.; Dufrene, B.M.; Mukhopadhyay, R.; Lee, C.; Mina, E.F.; Laskar, J. Neural-network-based parasitic modeling and extraction verification for RF/millimeter-wave integrated circuit design. *IEEE Trans. Microw. Theory Tech.* **2006**, *54*, 2604–2614. [CrossRef]
16. Root, D.E. Future device modeling trends. *IEEE Microw. Mag.* **2012**, *13*, 45–59. [CrossRef]
17. Liu, W.; Na, W.; Zhu, L.; Zhang, Q.J. A review of neural network based techniques for microwave device modeling. In Proceedings of the IEEE MTT-S International Conference Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO), Beijing, China, 27–29 July 2016; pp. 1–2.

18. Zhao, Z.; Zhang, L.; Feng, F.; Zhang, W.; Zhang, Q. Space mapping technique using decomposed mappings for GaN HEMT modeling. *IEEE Trans. Microw. Theory Tech.* **2020**, *68*, 3318–3341. [CrossRef]
19. Rizzoli, V.; Costanzo, A.; Masotti, D.; Lipparini, A.; Mastri, F. Computer-aided optimization of nonlinear microwave circuits with the aid of electromagnetic simulation. *IEEE Trans. Microw. Theory Tech.* **2004**, *52*, 362–377. [CrossRef]
20. Xu, J.J.; Yagoub, M.C.E.; Ding, R.T.; Zhang, Q.J. Neural based dynamic modeling of nonlinear microwave circuits. *IEEE Trans. Microw. Theory Tech.* **2002**, *59*, 913–923.
21. Liu, T.; Boumaiza, S.; Ghannouchi, F.M. Dynamic behavioral modeling of 3G power amplifier using real-valued time-delay neural networks. *IEEE Trans. Microw. Theory Tech.* **2004**, *52*, 1025–1033. [CrossRef]
22. Cao, Y.; Zhang, Q.J. A New Training Approach for robust recurrent neural-network modeling of nonlinear circuits. *IEEE Trans. Microw. Theory Tech.* **2009**, *57*, 1539–1553. [CrossRef]
23. Schreurs, D.; Droma, M.O.; Goacher, A.A.; Gadringer, M. *RF Power Amplifier Behavioral Modeling*; Cambridge University Press: New York, NY, USA, 2008.
24. Yan, S.; Zhang, C.; Zhang, Q.J. Recurrent neural network technique for behavioral modeling of power amplifier with memory effects. *Int. J. RF Microw. Comput.-Aided Eng.* **2015**, *25*, 289–298. [CrossRef]
25. Mkadem, F.; Boumaiza, S. Physically inspired neural network model for RF power amplifier behavioral modeling and digital predistortion. *IEEE Trans. Microw. Theory Tech.* **2011**, *52*, 2274–2284. [CrossRef]
26. Zhu, L.; Zhang, Q.J.; Liu, K.; Ma, Y.; Peng, B.; Yan, S. A Novel dynamic neuro-space mapping approach for nonlinear microwave device modeling. *IEEE Microw. Wirel. Compon. Lett.* **2016**, *26*, 131–133. [CrossRef]
27. Xu, J.J.; Yagoub, M.C.E.; Ding, R.T.; Zhang, Q.J. Exact adjoint sensitivity for neural based microwave modeling and design. *IEEE Trans. Microw. Theory Tech.* **2003**, *51*, 226–237.
28. Long, Y.S.; Guo, Y.X.; Zhong, Z. A 3-D table-based method for non-quasi-static microwave FET devices modeling. *IEEE Trans. Microw. Theory Tech.* **2012**, *60*, 3088–3095. [CrossRef]
29. Homayouni, S.M.; Schreurs, D.M.M.-P.; Crupi, G.; Nauwelaers, B.K.J.C. Technology-independent non-quasi-static table-based nonlinear model generation. *IEEE Trans. Microw. Theory Tech.* **2009**, *57*, 2845–2852. [CrossRef]
30. Zhang, L.; Rueda, H.; Kim, K.; Aaen, P. Non-quasi-static large-signal model for RF LDMOS power transistors. In Proceedings of the 2018 IEEE MTT-S International Microwave Symposium, Philadelphia, PA, USA, 10–15 June 2018.
31. Crupi, G.; Schreurs, D.M.M.-P.; Caddemi, A.; Raffo, A.; Vannini, G. Investigation on the non-quasi-static effect implementation for millimeter-wave FET models. *Int. J. RF Microw. Comput.-Aided Eng.* **2010**, *20*, 87–93. [CrossRef]
32. *Advanced Design System (ADS), 2013.06*; Keysight Technologies: Santa Rosa, CA, USA, 2013.
33. Statz, H.; Newman, P.; Smith, I.W.; Pucel, R.A.; Haus, H.A. GaAs FET device and circuit simulation in SPICE. *IEEE Trans. Electron. Device* **1987**, *34*, 160–169. [CrossRef]
34. Zhang, Q.J. *NeuroModelerPlus_V2.1E*; Carleton University: Ottawa, ON, Canada, 2008.
35. *Medici 2013. 12-0*; Synopsys Inc.: Mountain View, CA, USA, 2013.
36. Chang, C.Y.; Kai, F. *GaAs High-Speed Devices: Physics, Technology, and Circuit Applications*; Wiley: New York, NY, USA, 1994.