Exact Adjoint Sensitivity Analysis for Neural Based Microwave Modeling and Design

Jianjun Xu, Mustapha C.E. Yagoub, Runtao Ding[+], and Qi-Jun Zhang

Department of Electronics, Carleton University, 1125 Colonel By Drive, Ottawa, Canada, K1S 5B6
[+] School of Electronics and Information Engineering, Tianjin University, Tianjin, China

Abstract — **For the first time, an adjoint neural network method is introduced for sensitivity analysis in neural-based microwave modeling and design. Exact first and second order sensitivities are systematically calculated for generic microwave neural models including variety of knowledge based neural models embedding microwave empirical information. A new formulation allows the models to learn both the input/output behavior of the modeling problem and its derivative data simultaneously. Examples for passive and active microwave modeling and simulation are presented.**

I. INTRODUCTION

Neural networks have been recently recognized as a useful vehicle for RF and microwave modeling and design [1]. Neural networks can be trained from EM simulation or measurement data and subsequently used during circuit analysis and design. The models are fast and can represent EM/physics behaviors it learnt which otherwise are computationally expensive. Microwave researchers have recently demonstrated this approach in a variety of applications such as modeling and optimization of high-speed VLSI interconnects [2], CPW circuits [3], spiral inductors [4] and microwave FETs and amplifiers [5]-[6]. Knowledge based approaches with microwave empirical or equivalent circuit models embedded into the neural network learning process have also been studied [1].

This paper addresses a new task in this area, that is, neural based sensitivity analysis. Sensitivity information is very important in circuit optimization and modeling [7]. For neural networks, sensitivity analysis has been studied, for example, for multilayer perceptrons [8] and neural networks with binary responses for signal processing purposes [9]. However, to provide sensitivity information in a generic neural model with microwave functions, and to learn from sensitivity data remains an unsolved task.

For the first time, a general adjoint neural network sensitivity analysis technique is presented in this paper, which allows exact sensitivity to be calculated in a general neural model accommodating microwave empirical functions, equivalent circuit as well as conventional switch type neurons in an arbitrary neural network structure. Techniques for both first and second order derivative calculations are derived. Using the second order derivative, we are able to train a neural network model to learn not only device input/output data but also the derivative information, which is very useful in simultaneous DC/small-signal/large-signal device modeling [10]. The proposed sensitivity analysis technique is applied to high-speed VLSI interconnect modeling, large-signal FET modeling and 3-stage amplifier design examples.

II. PROPOSED ADJOINT NEURAL NETWORK APPROACH

A. The Adjoint Structure and Sensitivity Analysis

Two networks, one called the original neural network, and the other defined as the adjoint neural network, are utilized in the proposed sensitivity analysis technique. Each network consists of neurons and connections between neurons. Each neuron receives and processes stimuli (inputs) from other neurons and/or external inputs, and produces an output. Here we introduce a generic framework in which microwave empirical and equivalent models can be coherently represented in the neural network structures, and connections between neurons can be arbitrary allowing different types of microwave neural structures to be included.

Suppose for neuron i, the output is z_i and the external input is x_i. Let N be the total number of neurons in the original neural network and $\boldsymbol{z} = [z_1\ z_2\ \ldots\ z_N]^T$. In order to accommodate microwave empirical knowledge, we use a notation $f_i(\boldsymbol{z}, \boldsymbol{p}_i)$ to represent the processing function for neuron i where $\boldsymbol{p}_i$ could represent either the neuron connection weights or parameters in the empirical equivalent model. Assuming the neuron indices are numbered consecutively starting from the input neurons, through hidden neurons to the outputs neurons, the feedforward calculation of the original model can be defined as

$$z_i = f_i(\boldsymbol{z}, \boldsymbol{p}_i) + x_i \qquad (1)$$

calculated sequentially for $i = 1, 2, \ldots N$.

Let δ_{ij} be the Kronecker symbol (1 if $i = j$ and 0 if $i \neq j$). We introduce an adjoint neural model, which consists of N adjoint neurons. Let $\hat{z}_j$ be defined as the output of the *jth* adjoint neuron. The processing function for this adjoint neuron is a linear function defined as

$$\hat{z}_j = \sum_{i=j+1}^{N} \frac{\partial f_i}{\partial z_j} \cdot \hat{z}_i + \delta_{kj} \tag{2}$$

where $\partial f_i / \partial z_j$, which could be derivatives from microwave empirical functions, are the local derivatives of original neuron functions. To perform "feedforward" computation in the adjoint model, we first initialize the last several adjoint neurons by Kronecker functions, e.g., $\hat{z}_N = \delta_{kN}$, where k indicates the output neuron for which sensitivity is to be computed. Then we calculate (2) *backwards* according to the neuron sequence $j = N\text{-}1, N\text{-}2, \ldots, 1$. The first order derivatives of the outputs versus the inputs of the original neural model is $\partial z_k/\partial x_i = \hat{z}_i$.

In order to train the original neural model to learn input/output derivative data, we train the adjoint neural model, leading to the need of second order sensitivity analysis. The derivatives required to train the adjoint model is computed by,

$$\begin{aligned}\frac{\partial \hat{z}_k}{\partial \boldsymbol{p_i}} &= \sum_{j=1}^{i-1} \hat{\hat{z}}_j \cdot \frac{\partial^2 f_i}{\partial z_j \partial \boldsymbol{p_i}} \cdot \hat{z}_i \\ &+ \sum_{j=1}^{N-1} \sum_{m=j+1}^{N} \sum_{n=i}^{m-1} \hat{\hat{z}}_j \cdot \frac{\partial^2 f_m}{\partial z_j \partial z_n} \cdot \frac{\partial z_n}{\partial \boldsymbol{p_i}} \cdot \hat{z}_m\end{aligned} \tag{3}$$

where $\partial^2 f_i/\partial z_j \partial \boldsymbol{p_j}$ represents 2nd order derivative information in individual neurons, and $\hat{\hat{\boldsymbol{z}}}$ is solved from back-propagation in the adjoint neural model according to the neuron sequence $j = 2,3, \ldots N$ by initializing $\hat{\hat{z}}_1 = \delta_{k1}$ and

$$\hat{\hat{z}}_j = \sum_{m=1}^{j-1} \frac{\partial f_j}{\partial z_m} \cdot \hat{\hat{z}}_m + \delta_{kj} \tag{4}$$

B. Training

Let $\boldsymbol{d}$ and $\boldsymbol{d'}$ represent the training data for the original output $\boldsymbol{z}$ and its derivatives $d\boldsymbol{z}/d\boldsymbol{x}$, respectively. Let I, K and S be the index sets of input and output neurons, and samples in training data $\boldsymbol{d}$, respectively. We formulate the error function for training as,

$$E = \frac{1}{2} \sum_{s \in S} \left[w_1 \sum_{k \in K} (z_{ks} - d_{ks})^2 + w_2 \sum_{i \in I, k \in K} \left(\frac{dz_{ks}}{dx_{is}} - (d')_{kis} \right)^2 \right] \tag{5}$$

where subscripts i, k and s (used for x, z, d and d') indicate input neuron i, output neuron k and sample s, respectively, and w_1, w_2 are the weighting parameters. During training, both the original and the adjoint neural models share the same set of parameters $\boldsymbol{p}_i$, $i = 1, 2, \ldots, N$. Therefore training one model will also result in the other model being updated. There are three types of trainings. (i) Train original neural model using input/output data $\boldsymbol{d}$, and after training, the outputs of adjoint model automatically becomes derivative of original input/output. (ii) Train adjoint model only with derivative data $d\boldsymbol{z}/d\boldsymbol{x}$. The original model will then give original input/output (i.e., $\boldsymbol{x}$-$\boldsymbol{z}$) relationship, which has the effect of providing integration solution over derivative training data. (iii) Train both original and adjoint models together to learn $\boldsymbol{x}$-$\boldsymbol{z}$ and $d\boldsymbol{z}/d\boldsymbol{x}$ data, which will help the neural model to be trained more accurately and robustly.

III. EXAMPLES

A. Example 1

Fast and accurate sensitivity analysis of coupled transmission lines are important for high-speed VLSI interconnect optimization and statistical design. This example illustrates the proposed sensitivity technique for an arbitrary neural network structure where microstrip empirical formulas are used as part of a knowledge based neural network structure shown in Fig. 1.

After training with accurate EM based microstrip data (100 samples), we use the proposed method to provide exact derivatives of electrical parameters of the transmission line with respect to the physical-geometrical parameters needed in VLSI interconnect optimization. The sensitivity solution from the proposed method is verified with brute-force perturbation in Table I. Without neural model, such sensitivity would have been computed by perturbation in EM simulators. The computation time for the proposed method compared to EM perturbation solution is 3s versus 2660s for sensitivity analysis of 1000 microstrip models which are typically needed in optimization of a network of VLSI interconnects.

TABLE I
COMPARISON OF SENSITIVITY BETWEEN PERTURBATION TECHNIQUE AND ADJOINT TECHNIQUE

Sensitivity	Perturbation Technique	Adjoint Technique	Difference (%)
dL_{12}/dw_1	-0.1440	-0.1435	0.354
dL_{12}/dw_2	0.0620	0.0616	0.645
dL_{12}/ds	-0.8462	-0.8514	0.610
dL_{12}/dh	0.5338	0.5337	0.018
$dL_{12}/d\,\varepsilon_r$	-0.0010	-0.0010	0.001
$dL_{12}/dfreq$	-0.0037	-0.0037	0.001

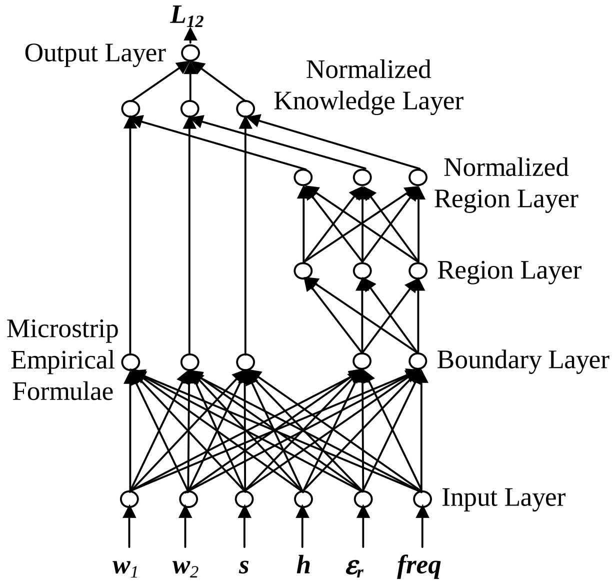

Fig. 1. Knowledge based coupled transmission line neural model of mutual inductance (L_{12}) for VLSI interconnect optimization. w_1, w_2, s, h, and ε_r are conductor widths, spacing, substrate thickness and dielectric constants, respectively.

B. Example 2

This example illustrates the integration effect of the adjoint neural model. We first train only the adjoint neural model to learn the nonlinear capacitor data, which is generated from Agilent-ADS. After training the adjoint model with 41 data samples, we then use the original neural model without re-training (with internal parameters updated according Section II.B) as a nonlinear charge-model (i.e., Q-model). The charge model is compared with analytical integration of ADS capacitor formula (Fig. 2). The good agreement in the figure verifies the integration effect of training the adjoint neural model. This example shows an interesting solution to one of the frequently encountered obstacles in developing a charge model for nonlinear capacitors required for harmonic balance simulators with only capacitor data available.

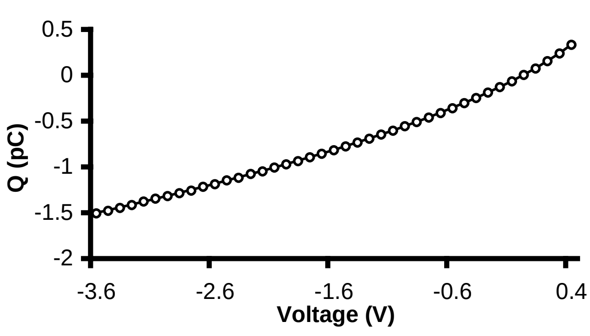

Fig. 2. Comparison of Q from adjoint neural model (o) with Q from analytical integration (—).

C. Example 3

This example shows large-signal device modeling using DC and small-signal training data. The model used is a knowledge based approach where existing equivalent circuit model is combined with neural net learning. In practice, manually creating formulas for the nonlinear current and charge sources in a FET model could be very time-consuming. Here we use neural networks to automatically learn the unknown relationship of gate-source charge Q_{gs} and drain current I_d as nonlinear functions of gate and drain voltages. However we do not have explicitly the charge data Q_{gs} and dynamic current data I_d for training the model. The available training data is the DC and bias-dependent S-parameters of the *overall* FET, which in our example is generated using Agilent-

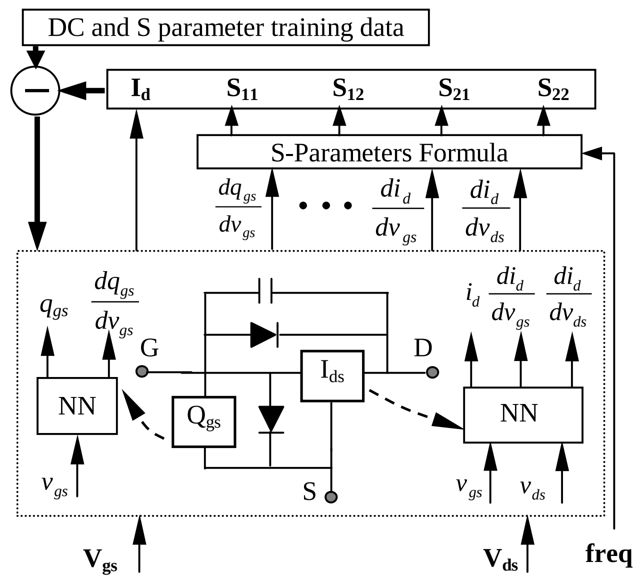

Fig. 3: Large-signal FET modeling with combined equivalent circuit and adjoint neural networks trained by DC and bias-dependent S-parameters.

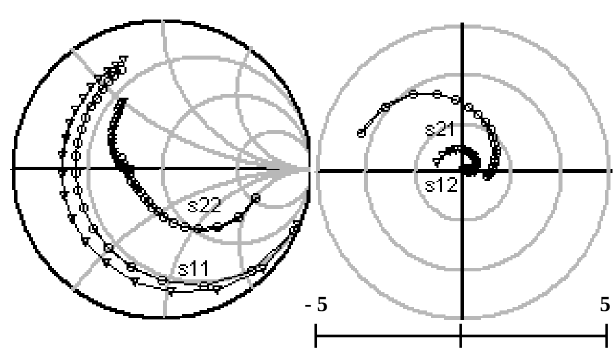

Fig.4: Comparison between S-parameters of the ADS Statz model (—) and our complete neural FET model (o), (Δ) at two of the ninety bias points : V_{ds} = 3.26 V, V_{gs} = -0.6 V and V_{ds} = 0.26 V, V_{gs} = -0.6 V respectively.

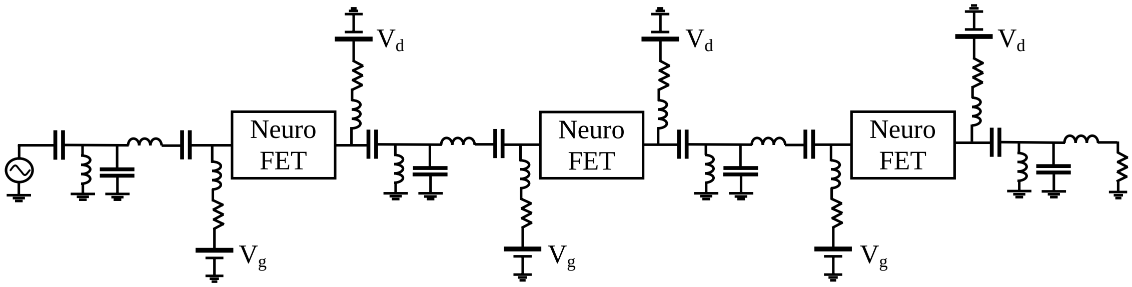

Fig. 5: The 3-stage amplifier where the FET models used are neural models trained from the proposed method following Fig. 3.

ADS with Statz Model. Therefore the neural network and the rest of the FET equivalent circuit are combined into a knowledge based model and they together are trained to learn the training data, shown in Fig. 3. Notice that learning S-parameters means learning the derivative information of the large-signal model. After training, a good agreement of small signal responses at all the 90 bias points between our neural model and those given by the ADS solution is observed, as shown in Fig. 4.

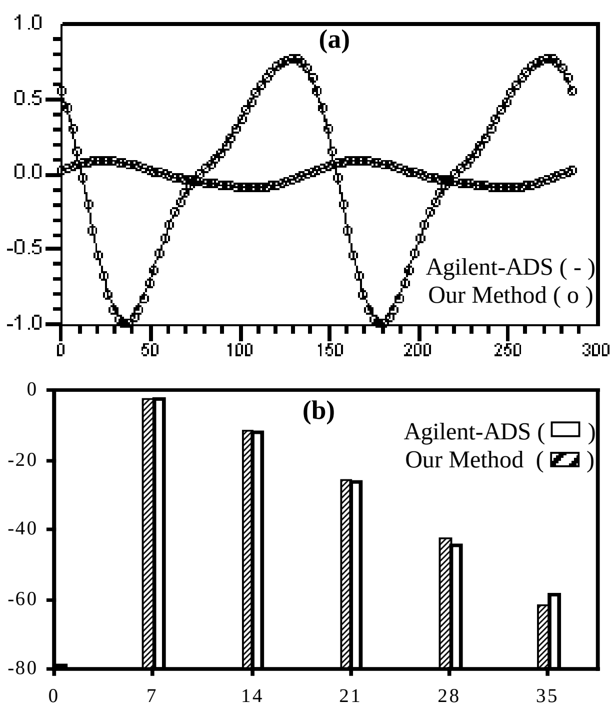

Fig. 6: Comparison of the amplifier simulation using the FET model trained by our approach and that by original Agilent-ADS model: (a) Time domain amplifier input and outputs. (b) Frequency domain harmonic solutions of the amplifier output.

We then used our complete knowledge based FET neural model in a three-stage power amplifier shown in Fig. 5 for large-signal simulation. The large-signal response of the amplifier using our model agrees well with that using original *ADS* model (Fig. 6). This example demonstrates that the proposed method can be used for efficient generation of nonlinear device models for use in large-signal simulation and design.

REFERENCES

[1] Q.J. Zhang and K.C. Gupta, *Neural Networks for RF and Microwave Design*, Norwood, MA: Artech House, 2000.

[2] A. Veluswami, M.S. Nakhla and Q.J. Zhang, "The application of neural networks to EM-based simulation and optimization of interconnects in high-speed VLSI circuits", *IEEE Trans. Microwave Theory Tech.*, vol. 45, pp. 712-723, 1997.

[3] 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.*, vol. 45, pp. 2515-2523, 1997.

[4] G. L. Creech, B.J. Paul, C.D. Lesniak, T.J. Jenkins, and M.C. Calcatera "Artificial neural networks for fast and accurate EM-CAD of microwave circuits," *IEEE Trans. Microwave Theory Tech*., vol. 45, pp. 794-802, 1997.

[5] H. Zaabab, Q.J Zhang and M.S. Nakhla, "A neural network approach to circuit optimization and statistical design," *IEEE Trans. Microwave Theory Tech.*, vol. 43, pp. 1349-1358, 1995.

[6] F. Wang and Q.J. Zhang, "Knowledge based neural models for microwave design," *IEEE Trans. Microwave Theory Tech.*, vol. 45, pp. 2333-2343, 1997.

[7] J.W. Bandler, Q.J. Zhang and R. Biernacki, "A unified theory for frequency-domain simulation and sensitivity analysis of linear and nonlinear circuits", *IEEE Trans. Microwave Theory Tech.*, vol. 36, pp. 1661-1669, 1988.

[8] G. Antonini and A. Orlandi, "Gradient evaluation for neural-networks-based electromagnetic optimization procedures," *IEEE Trans. Microwave Theory Tech.*, vol. 48, pp. 874-876, 2000.

[9] S. Haykin, *Neural Networks, A Comprehensive Foundation*, New Jersey: Prentice Hall, 1994.

[10] J.W. Bandler, R.M. Biernacki, S.H. Chen, J. Song, S. Ye and Q.J. Zhang, "Analytically unified DC/small-signal/large-signal circuit design", *IEEE Trans. Microwave Theory Tech.*, vol. 39, pp. 1076-1082, 1991.