POLITECNICO DI TORINO
Repository ISTITUZIONALE

Nonlinear Behavioral Modeling of FETs: Toward the Implementation of Deep Neural Networks Through Large Signal Data and EDA Tools

Original
Nonlinear Behavioral Modeling of FETs: Toward the Implementation of Deep Neural Networks Through Large Signal Data and EDA Tools / Kouhalvandi, L., DONATI GUERRIERI, S.. - ELETTRONICO. - (2024). (19th European Microwave Integrated Circuits Conference, EuMIC Paris (Fra) 23 -24 September 2024) [10.23919/EuMIC61603.2024.10732844].

Availability:
This version is available at: 11583/2993330 since: 2024-11-04T08:48:22Z

Publisher:
IEEE

Published
DOI:10.23919/EuMIC61603.2024.10732844

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2024 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

15 September 2026

Nonlinear Behavioral Modeling of FETs: Toward the Implementation of Deep Neural Networks through Large Signal Data and EDA Tools

Lida Kouhalvandi[#1], Simona Donati Guerrieri[^2],
[#]EEM, Dogus University, Turkiye
[^]DET, Politecnico di Torino, Italy
[1]lida.kouhalvandi@ieee.org, [2]simona.donati@polito.it

***Abstract*— Nonlinear behavioral Field-Effect Transistor (FET) models often rely on large look-up tables extracted from extensive load-pull characterization. Besides the numerical burden, these models have limited extrapolation capabilities and can hardly be made dependent on the device technology. In this paper, we demonstrate that a Long Short-Term Memory (LSTM)-based Deep Neural Network (DNN) is an effective alternative modeling approach. The DNN is trained with the load-pull data within a simulation platform where data exchange between an Electronic Design Automation (EDA) tool (such as PathWave ADS) and a programming platform (such as MATLAB) is exploited. As a test case, the DNN model has been extracted for an S-band MACOM 10W GaN power device, for which the Enhanced Poly-Harmonic Distortion (EPHD) behavioral model is also available in the ADS. The accuracy of DNN model is verified against the EPHD model in terms of output power, gain, efficiency, and dynamic load lines. Compared to other behavioral models, the DNN approach is expected to provide superior extrapolation capability and to be easily reconfigurable to add/combine heterogeneous device data e.g. from advanced characterization, including memory, and physical (TCAD, EM) simulations.**

***Keywords*— Behavioral model, Deep Neural Network (DNN), Field-Effect Transistor (FET), Large Signal (LS) model, Harmonic Balance (HB), load-pull, Long Short-Term Memory (LSTM), microwave device.**

I. Introduction

Wireless communication systems, including Fifth-generation/Sixth-generation (5G/6G) wireless networks, are fast developing and require high-performance yet accurate design tools [1]. Microwave power devices play the main role in enhancing the overall performance of communication systems, and in particular of the power amplifier.

Developing accurate Large Signal (LS) active device models is the main challenge [2], especially for the Gallium Nitride (GaN) technology, where the devices still suffer from significant dispersion effects due to traps and temperature effects. The accuracy of LS models is greatly increased when they are extracted directly from measured data, making it necessary to exploit a massive harmonic load/source pull characterization campaign. Among these approaches, X-Parameters and, more generally, advanced poly-harmonic distortion-based models [3], [4] have nowadays reached sufficient maturity to compete with traditional compact models [5]. Despite this, they suffer from limited extrapolation capabilities, and their validity is often restricted within the range of, bias, frequencies, and loads where the extraction has been made. Furthermore, they are often implemented as large look-up tables, which require massive interpolation and intensive numerical effort. Recently, various attempts have been devoted to developing Neural Network (NN) based device models (see [6] for a review): due to their extrapolation capability and the ease of accommodate additional data either by re-tuning the NN weights or by adding a limited number of neurons, NNs are expected to provide a good trade-off in terms of model accuracy and generality. For example, while behavioral models are entirely oblivious of the underneath device technology, NN models may be trained to incorporate a dependency on physical parameters, e.g. the device periphery or the statistical variability. Most NNs, though, are still limited to DC or small signal behavior and focus on extracting the parameter of equivalent circuits taking either the frequency or the gate/drain voltages as input neurons. Significant effort has been dedicated to simulating GaN High Electron Mobility Transistor (HEMT) memory effects [7]. On the other hand, there is a significant need to develop NN that can describe the device in terms of large signal input and reflected waves and work directly in the Harmonic Balance (HB) environment used in microwave circuit design. Recently, some studies present NNs that are based on port waves for transistor modeling. In [8] and in [9], the input layer limits to the available input port power and the incident waves of the other ports at the fundamental, making it difficult to use the model for harmonic loads. In [10], the input waves are taken at all harmonics but the NN structure is the Shallow Neural Network (SNN) (i.e., a network with one hidden layer).

This paper presents a new methodology for the nonlinear modeling of Field-Effect Transistor (FET) active devices through Long Short-Term Memory (LSTM)-based Deep Neural Network (DNN) that can be linked to Electronic Design Automation (EDA) tools such as ADS in terms of port waves. The proposed DNN is trained with load-pull data: incident waves available from harmonic load pull data are inserted into the DNN and the corresponding reflected waves are generated. Hence, the DNN model can be directly used as a replacement for LS device models in HB simulations. As a test case, the DNN model has been extracted for an S-band MACOM 10W

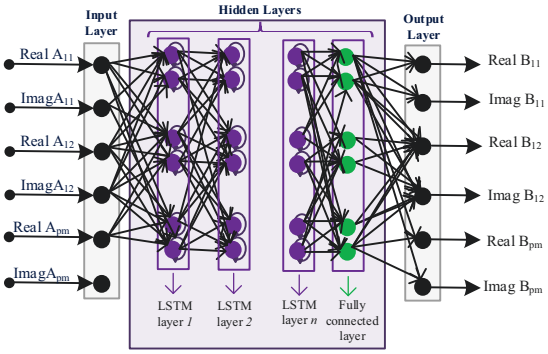

Fig. 1. Proposed LSTM-based DNN.

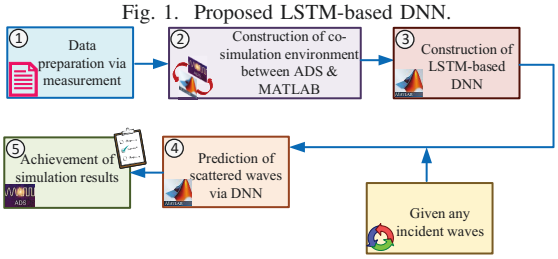

Fig. 2. General view of optimization method.

GaN power device, using a set of load-pull data for which the Enhanced Poly-Harmonic Distortion (EPHD) behavioral model is also available in ADS. The accuracy of proposed method is verified against the EPHD model in terms of output power (P_{out}), gain, power added efficiency (PAE), and dynamic load lines. The comparison is carried out with several loading conditions not used for the DNN model extraction, demonstrating the effectiveness of the proposed approach.

II. DNN Large Signal Model

The aim of this paper is to develop a DNN model that can be used as a replacement for a compact or behavioral model in HB simulations. As such, the developed model works directly with frequency-dependent, harmonic data. Figure 1 presents the general structure of LSTM-based DNN used to model the nonlinear active device. The input layer includes the real and imaginary incident port waves $A_{k,l}$ (where k is the port index and l the harmonic index) while the output layer returns the reflected waves.

The capability of the proposed DNN to deal with active device modeling has been already demonstrated in a previous work [11], where the incident and reflected waves used to train the model were obtained from LS and X-parameters data generated by an independent TCAD physical simulator [12], [13]. Here, we aim to link the DNN to EDA environments and, specifically, to PathWave ADS. The LSTM-based DNN is here trained directly through load-pull data, with multiple incident and reflected waves differing in terms of amplitude, port termination, and number of harmonics. As usual for active device modeling (i.e. for unmatched devices), extensive harmonic source and load pull data are required [4]. The general overview of the DNN optimization procedure is presented in Fig. 2.

The first step for training any DNN is to provide a suitable amount of data (Step-1). Multiple file formats of load-pull data are used according to the specific characterization set-up used (e.g. *.cst*, *.spl* etc.), some being proprietary. In order to make the DNN extraction general, the '*.mdf*' format is preferable, which is available within ADS (used in this work) as well as other EDA tools such as Cadence AWR. Since EDA tools (such as PathWave ADS) do not yet implement dedicated NNs built-in platforms, it is necessary to exploit an external numerical program, such as MATLAB, to extract and implement the DNN. Furthermore, a co-simulation environment allowing data exchange between ADS and MATLAB is necessary (Step-2). We exploit the dedicated ADS automated environment [14] in which ADS is working in the background and MATLAB is handling all the received/transferred data for optimizing and doing mathematical analysis [15]. In order to obtain input waves from the circuit simulator and feedback the DNN output to ADS the '.mdf' file is updated. In particular, a subset of the load-pull data is randomly selected for DNN training and testing (see next for details on a practical case study), leading to the optimized DNN topology and structure (Step-3). The extraction is done in MATLAB as shown in (1):

$$\text{net} = \text{trainNetwork}(X_{\text{Train}}, Y_{\text{Train}}, \text{layers}, \text{options}) \quad (1)$$

The required data includes input and output training data (X_{Train} and Y_{Train}). After training the DNN, it is critical to inspect the accuracy of the network using an independent set of input test data X_{Test} and monitoring the predicted output data Y_{Pred}, as shown in (2).

$$Y_{\text{Pred}} = \text{predict}(net, X_{\text{Test}}) \quad (2)$$

The topology of hidden layers is based on the LSTM structure in which the hyperparameters of DNN are provided by the 'rule of thumb' [15]. The activation function is the Rectified Linear Unit (ReLU), a function where the normalized Root Mean Square Error (RMSE) is determined for calculating the convergence of the LSTM-based DNN. The weights and biases of the DNN are updated through the Adam optimization algorithm and standard gradient descent algorithm in which the training options are set as a solver to 'Adam' and 'gradient threshold' to 1. Once the DNN is trained, it can be employed for predicting the scattered waves for any given incident waves (Step-4). These estimated outcomes are substituted into the file with the extension of '.mdf' in the ADS through MATLAB (see Fig. 5). The data generated from the previous step can be used in circuit simulation to generate various outputs in terms of input/output power, gain, and PAE (Step-5).

III. DNN Implementation

In this section, we present an example of the implementation of proposed method in a test case. We address the extraction of the DNN model for an S-band 10 W GaN HEMT power device from MACOM (CGH40010). The whole setup exploits a PC with Intel Core i7-4790, CPU @ 3.60 GHz and 32.0 GB RAM.

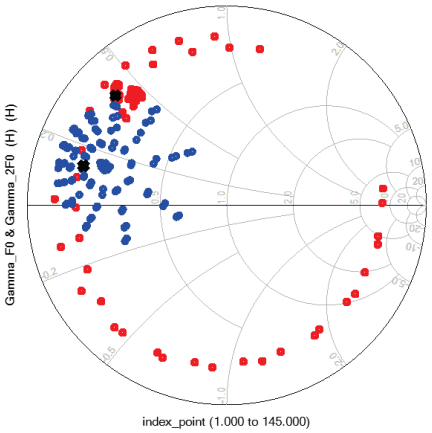

Fig. 3. Measured loads at fundamental frequency (blue circles) and 2nd harmonic (red squares).

```
! DATASOURCE NAME: MyEPHD_Meas_AB_3p743p95G.cst
! TRANSISTOR NAME:
! TRANSISTOR TYPE: Field effect
! FUNDAMENTAL FREQUENCY: 3.7 GHz
! HARMONICS: 1
VAR idx_bloc(1)=   1
BEGIN ISCRIVIA
%   INDEX   V1   I1   V2   I2   a1_r     a1_i     b1_r     b1_i     a2_r     a2_i     b2_r     b2_i
1   -2.60201   5.64657E-5  28.0   0.137675   -0.0830036  0.0356655   0.0610307   -0.0541931  -0.214273   0.0677698   0.3492   0.0112576
2   -2.60201   5.64657E-5  28.0   0.137675   7.92510E-5  5.54032E-5  1.71110E-4  -2.60825E-4 8.89947E-4  -0.00151202 -0.00222761 5.46995E-4
3   -2.60185   3.50433E-5  28.0   0.137357   -0.0980726  0.0377336   0.0648101   -0.0572803  -0.227133   0.0713289   0.369991   0.0125016
4   -2.60185   3.50433E-5  28.0   0.137357   9.12189E-5  6.42349E-5  1.97005E-4  -2.90456E-4 0.00107886  -0.00168888 -0.00250243 5.85634E-4
5   -2.60199   -1.35143E-5 28.0   0.1371   -0.0928259  0.0397964   0.0682689   -0.0604334  -0.239467   0.0753459   0.390199   0.0130347
6   -2.60199   -1.35143E-5 28.0   0.1371   9.29169E-5  6.46746E-5  2.02183E-4  -3.32754E-4 0.00120017  -0.00191939 -0.00282849 6.45024E-4
7   -2.60213   -3.35510E-5 28.0   0.136742   -0.0981910  0.0420744   0.0722575   -0.063883   -0.253276   0.0796553   0.41256  0.0139032
8   -2.60213   -3.35510E-5 28.0   0.136742   1.13275E-4  7.68244E-5  2.38558E-4  -3.72591E-4 0.00140136  -0.00214561 -0.00318376 6.89206E-4
9   -2.60179   -2.88778E-5 28.0   0.136386   -0.103705   0.0444438   0.0763025   -0.0674844  -0.267421   0.0840113   0.435721   0.0147068
10  -2.60179   -2.88778E-5 28.0   0.136386   1.22276E-4  8.53065E-5  2.54636E-4  -4.20248E-4 0.0015989   -0.00238288 -0.00357696 7.47308E-4
```

Fig. 4. Example of the '.mdf' load-pull data file used in harmonic balance simulations in ADS, including DC voltages and currents and real and imaginary parts of the port waves. Various blocks refer to independent load conditions.

By referring to the steps in the previous section, the first step corresponds to gaining measurement data for the GaN active device. Here, we leverage a dataset included within an ADS test workspace provided by AMCAD [16]. The characterization data is already in the '.mdf' format, and it is carried out at the gate bias of -2.6 V, drain bias of 28 V, and two fundamental frequencies (3.7 GHz and 3.95 GHz) with two harmonics. The equivalent loads are shown in Fig. 3 and also an example of the data file is presented in Fig. 4. The same dataset is also used by AMCAD to develop an EPHD [16] behavioral model which will also be used here to further validate the extracted DNN. Load pull simulations are carried out in ADS with load tuners as sketched in Fig. 5.

To build the LSTM-based DNN, the dataset is separated into three sub-sets as X_{Train}, X_{Val}, and X_{Test} with a ratio of 70%, 15%, and 15%, respectively. The partition is done by randomly selecting the indexes corresponding to the various loads and wave amplitude in the data file. The validation data are not used for the DNN build-up and are used for the validation results presented hereafter. In total, 2336 data is used for training the LSTM-based DNN and Fig. 6 presents the accuracy of the constructed DNN. As it is clear by increasing the number of hidden layers, the accuracy of the network is enhanced in comparison with the SNN. To limit the complexity, the optimum DNN includes 4 hidden layers with 120 fully connected neurons is selected in which the RMSE of 0.098 at the 120^{th} neuron is achieved. Figure 7 presents the loss performance of the trained network in terms of iteration. Totally, the consumed time for training and executing the DNN lasts around 3 hours and 30 minutes.

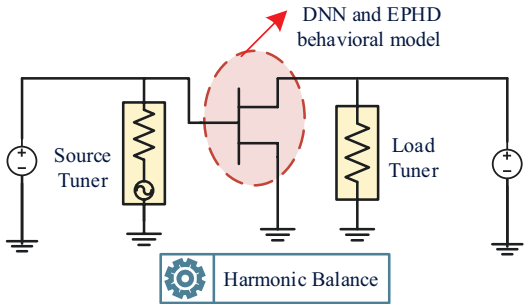

Fig. 5. Set-up environment in ADS for testing the DNN in Load-pull simulations. The EPHD behavioral model from AMCAD is also used as a referenc

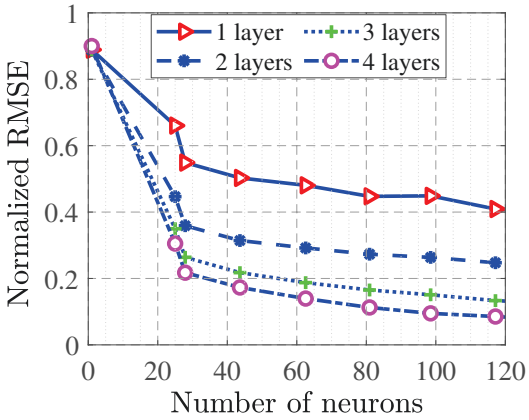

Fig. 6. Accuracy of the proposed DNN in comparison with the SNN.

To demonstrate the effectiveness of the extracted LSTM-based DNN, a load-pull analysis is carried to compare the EPHD and modeled DNN for various loads and input power that are not used for the DNN training. Notice that the proposed model, being trained on load-pull data, is at this moment entirely memory-less, hence the model validation is limited to single-tone analysis. Figure 8 describes an example of the obtained accuracy in terms of Gain, PAE, and P_{out}. The loading conditions are reported in Fig. 3 with black marks. Similar accuracy has been obtained for all the inspected loads. Time domain waveforms cannot be directly investigated by the characterization data. In order to test the DNN behavior in the time domain, we compare the predictions of the HB analysis comparing the DNN with the available (and already optimized) EPHD AMCAD model, taken here as a reference.

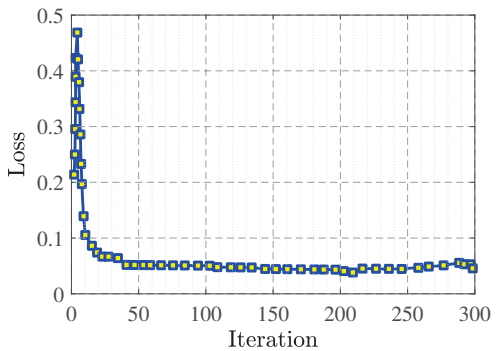

Fig. 7. Loss outcome of constructed LSTM-based DNN over the iterations.

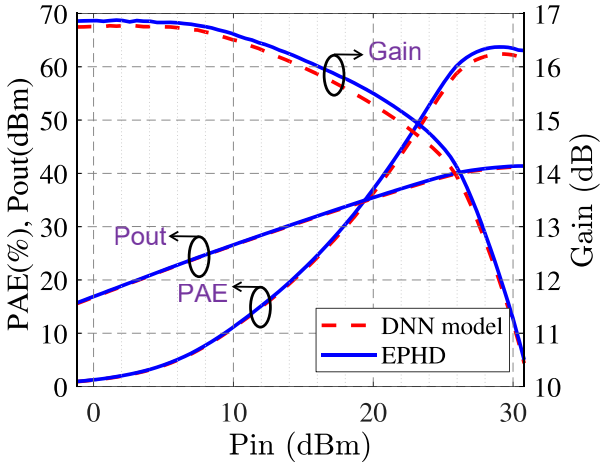

Fig. 8. Comparison of DNN and EPHD data: Gain, PAE, and P_{out} for the load marked in Fig. 3.

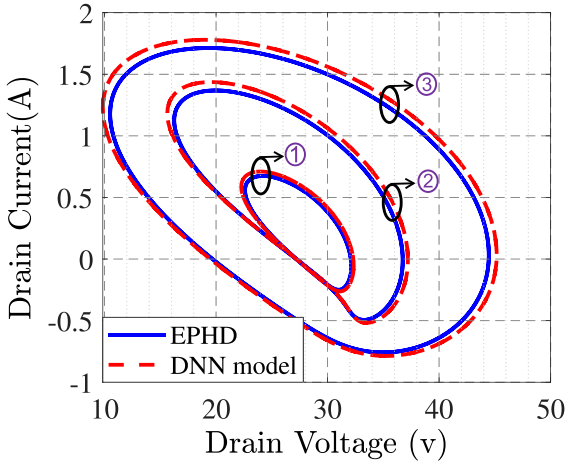

Fig. 9. DNN vs. EPHD dynamic load lines for three operating conditions.

The dynamic load-lines on three different operating conditions are reported in Fig. 9, where 1 and 2 refer to the same load and different input power, whereas 3 refers to different loading conditions. As it is illustrated, excellent agreement is observed.

IV. Conclusion

We have demonstrated a new approach for DNN-based nonlinear active device modeling using incident and reflected waves. The developed simulation environment allows to exploitation of the DNN into HB ADS simulations by exploiting intermediate data files. The direct implementation of the DNN in EDA tools would be beneficial to exploit the DNN speed and the circuit simulator capabilities. While the proposed model still performs similarly to other behavioral models extracted on the same characterization data, the DNN approach is promising for its flexibility. DNNs are expected to provide superior extrapolation capability and to be easily reconfigurable to add/combine heterogeneous device data from advanced characterization and physical (TCAD, EM) simulations, e.g. to include memory or to make them dependent on relevant technological ad layout parameters.

Acknowledgement

The authors would like to thank AMCAD Engineering for providing reference measurement results and behavioral model.

References

[1] C.-X. Wang, X. You, X. Gao, *et al.*, "On the road to 6G: Visions, requirements, key technologies, and testbeds," *IEEE Communications Surveys Tutorials*, vol. 25, no. 2, pp. 905–974, 2023. DOI: 10.1109/COMST.2023.3249835.

[2] R. P. Martinez, M. Iwamoto, J. Xu, *et al.*, "Assessment and comparison of measurement-based large-signal FET models for GaN HEMTs," *IEEE Transactions on Microwave Theory and Techniques*, pp. 1–0, 2024. DOI: 10.1109/TMTT.2023.3349172.

[3] D. Root, J. Horn, J. Verspecht, and M. Marcu, *X-Parameters*. Cambridge University Press, 2013. DOI: 10.1017/cbo9781139042970.

[4] C. Maziere, D. Gapillout, A. Xiong, and T. Gasseling, "High efficiency Doherty power amplifier design using enhanced poly-harmonic distortion model," in *2015 IEEE International Conference on Microwaves, Communications, Antennas and Electronic Systems (COMCAS)*, 2015, pp. 1–5. DOI: 10.1109/COMCAS.2015.7360453.

[5] A. Dasgupta, S. Ghosh, Y. S. Chauhan, and S. Khandelwal, "ASM-HEMT: Compact model for GaN HEMTs," in *2015 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, 2015, pp. 495–498. DOI: 10.1109/EDSSC.2015.7285159.

[6] L. Kouhalvandi and S. Donati Guerrieri, "Modeling of HEMT devices through neural networks: Headway for future remedies," in *2023 10th International Conference on Electrical and Electronics Engineering (ICEEE)*, 2023, pp. 261–267. DOI: 10.1109/ICEEE59925.2023.00054.

[7] Z. Zhu, G. Bosi, A. Raffo, G. Crupi, and J. Cai, "Accurate modeling of GaN HEMTs oriented to analysis of kink effects in S22 and H21: An effective machine learning approach," *IEEE Journal of the Electron Devices Society*, vol. 12, pp. 201–210, 2024. DOI: 10.1109/JEDS.2024.3364809.

[8] G. Avolio, A. Raffo, M. Marchetti, G. Bosi, V. Vadalà, and G. Vannini, "GaN FET load-pull data in circuit simulators: A comparative study," pp. 80–83, 2019. DOI: 10.23919/EuMIC.2019.8909451.

[9] J. Louro, C. Belchior, D. R. Barros, *et al.*, "New transistor behavioral model formulation suitable for doherty PA design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 69, no. 4, pp. 2138–2147, 2021. DOI: 10.1109/TMTT.2021.3054645.

[10] M. Tian, J. Bell, E. Azad, R. Quaglia, and P. Tasker, "A novel cardiff model coefficients extraction process based on artificial neural network," in *2023 IEEE Topical Conference on RF/Microwave Power Amplifiers for Radio and Wireless Applications*, 2023, pp. 1–3. DOI: 10.1109/PAWR56957.2023.10046221.

[11] L. Kouhalvandi, E. Catoggio, and S. Donati Guerrieri, "Synergic exploitation of TCAD and deep neural networks for nonlinear FinFET modeling," in *IEEE EUROCON 2023 - 20th International Conference on Smart Technologies*, 2023, pp. 542–546. DOI: 10.1109/EUROCON56442.2023.10198982.

[12] F. Bonani, S. Donati Guerrieri, F. Filicori, G. Ghione, and M. Pirola, "Physics-based large-signal sensitivity analysis of microwave circuits using technological parametric sensitivity from multidimensional semiconductor device models," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 5, pp. 846–855, 1997. DOI: 10.1109/22.575609.

[13] S. Donati Guerrieri, F. Bonani, and G. Ghione, "Linking X parameters to physical simulations for design-oriented large-signal device variability modeling," in *2019 IEEE MTT-S International Microwave Symposium (IMS)*, 2019, pp. 204–207. DOI: 10.1109/MWSYM.2019.8700869.

[14] *Keysight Advanced Design System (ADS) to Matlab interface*, http://nl.mathworks.com/matlabcentral\/fileexchange/76183-keysight-advanced-design-system-ads-to-matlab-interface, Accessed: 2024-03-15.

[15] L. Kouhalvandi, O. Ceylan, and S. Ozoguz, "Automated deep neural learning-based optimization for high performance high power amplifier designs," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 12, pp. 4420–4433, 2020. DOI: 10.1109/TCSI.2020.3008947.

[16] *Enhanced Poly Harmonic Distortion (EPHD) modeling*, https://www.amcad-engineering.com/load-pull/, Accessed: 2024-03-03.