Prediction of FinFET Current-Voltage and Capacitance-Voltage Curves Using Machine Learning With Autoencoder

Kashyap Mehta and Hiu-Yung Wong, *Senior Member, IEEE*

***Abstract*— In this letter, we demonstrated the possibility of predicting full transistor current-voltage (*IV*) and capacitance-voltage (*CV*) curves using machines trained by Technology Computer-Aided Design (TCAD) generated data. 3D FinFET I_DV_G and C_GV_G predictions are used as examples. The machine is constructed through manifold learning using Autoencoder (AE) to extract the latent variables which are then correlated to physical parameters through 3rd-order polynomial regression. No device physics domain expertise is required in the machine learning process because there is no need to extract device metrics such as transconductance (g_m) or Drain-Induced-Barrier-Lowering (*DIBL*) from the TCAD training data. We show that the machine can predict not just the *full IV/CV* curves but also g_m (1st derivative quantity) and DIBL (extracted from two machines trained by different data). Good results can be obtained even with < 50 training data. Our work shows that manifold learning is possible in device *IV* and *CV* to capture the complex physics and, thus, it is expected that it is possible to predict the *IV/CV* of novel devices using limited experimental data before the underlying physics is well-understood.**

***Index Terms*— Autoencoder, FinFET, machine learning, simulation, technology computer-aided design (TCAD).**

I. Introduction

RECENTLY, TCAD augmented Machine Learning (TCAD-ML) has gained increased attention [1]–[7]. Due to the scarcity of experimental data, TCAD has been proposed to generate appropriate data to train machines to find the source of defect and process variation of a given abnormal *IV* curve [1], [2]. It has been experimentally demonstrated that a TCAD-trained machine can be used to deduce the physical parameters (such as the effective contact workfunction) of a device based solely on its experimental *IV* curve [3], [4] for troubleshooting.

TCAD-ML has also been proposed to assist power device [5] and junctionless nanowire designs [6]. One of the main purposes of these studies is to avoid additional TCAD simulations after the machine is trained. However, usually, there are two limitations. Firstly, domain expertise is required. For example, relevant input features (e.g. ON/OFF state currents, I_{ON}, I_{OFF}) need to be extracted from the raw TCAD data for machine learning to predict the threshold voltage [6]. This precludes the rapid adoption of ML in novel devices before the underlying physics is well understood. Secondly, usually 1000-2000 TCAD simulations are required. This increases the cost and reduces the value of TCAD-ML, in particular when 3D simulations are required. Neural networks have also been used to predict *IV/CV* curves by training on TCAD or SPICE data [8]–[10]. However, large amount of training data [9], domain expertise and significant machine tuning are usually required.

Therefore, in this letter, we attempt to demonstrate the possibility of using about *1 order of magnitude less training data (<200)* to train machines to predict n-type FinFET *full IV/CV* curves, as an example, *without domain expertise*. Autoencoder (AE) [11], a type of manifold learning, is found to be effective in achieving the goals because of its capability in performing non-linear dimensionality reduction to latent variables. We demonstrated that the latent variables can be mapped to device parameters and thus full *IV/CV* can be deduced from unseen device structures/parameters. Note that this work is different from [7] in which AE was used to perform *p-i-n* diode inverse design.

II. TCAD Simulation and Data Generation

3D n-type FinFET is constructed in TCAD Sentaurus Process [12] and its I_DV_G ($V_D = 0.8V$ for saturation, or $V_D = 0.05V$ for linear) and C_GV_G ($V_S = V_D = 0V$) are simulated in Sentaurus Device [13] (Fig. 1). Essential process and device models are included in the simulations, including stress effect due to gate-last high-k metal gate and C-doped Si source/drain, the impact of crystal orientation and stress on mobility and bandgap, thin layer and high-k induced mobility degradation, and band-to-band tunneling (BTBT). Model details can be found in [14]. The FinFET is made realistic with corner rounding and tapered fin. The fin bottom width is set to 15nm. To generate the I_DV_G / C_GV_G data, the gate length (L_G), fin top width (W_{TOP}), and gate metal workfunction (WF) are varied randomly and independently in the ranges of 15nm-25nm, 5nm-15nm, and 4.4eV-4.7eV, respectively. V_G is swept from −0.6V to 0.8V. Each curve is discretized with 80 equal intervals and used as the input features for ML (Fig. 2 and Fig. 3a, b). 250 curves of each type (C_GV_G, linear I_DV_G, and saturation I_DV_G) are generated.

Manuscript received November 30, 2020; revised December 9, 2020 and December 12, 2020; accepted December 13, 2020. Date of publication December 15, 2020; date of current version January 27, 2021. The review of this letter was arranged by Editor V. Moroz. *(Corresponding author: Hiu-Yung Wong.)*

The authors are with the Department of Electrical Engineering, San Jose State University, San Jose, CA 95192 USA (e-mail: hiuyung.wong@sjsu.edu).

Color versions of one or more figures in this letter are available at https://doi.org/10.1109/LED.2020.3045064.

Digital Object Identifier 10.1109/LED.2020.3045064

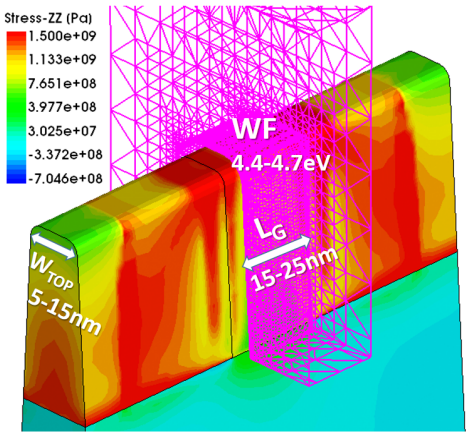

Fig. 1. FinFET structure created for the simulation. Channel direction stress distribution is displayed to show the complexity of the system. The 3 parameters varied and their ranges are identified. Only Silicon is shown for clarity. Pink mesh is the mesh of gate contact.

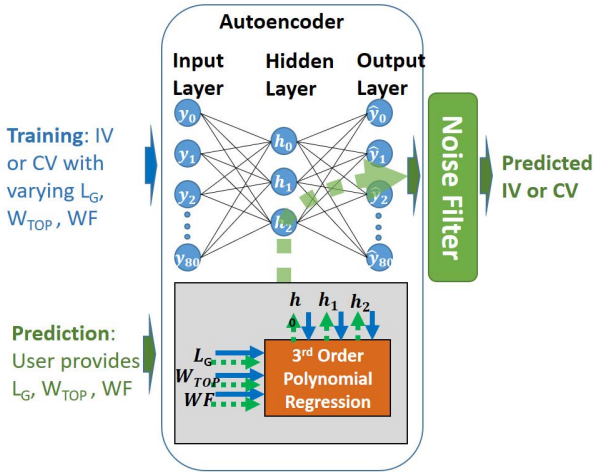

Fig. 2. The ML framework used in this study. Blue path represents the training flow during which AE and 3rd order PR are trained. Green path represents the prediction flow. Noise filter is noisy autoencoder + LPF, which is not needed when there are enough training data (e.g. AE200). *For clarity, only 3-layer AE is shown. In the experiment, actually 5-layer AE is used.*

III. Machine Learning Algorithm and IV Prediction

C_GV_G, linear I_DV_G, and saturation I_DV_G machines are trained by the corresponding TCAD curves for predictions. For each type (e.g. C_GV_G), 50 of the 250 TCAD curves, e.g. C_GV_G, are randomly selected and *set aside for testing,* and then three machines are trained, namely, AE200, AE50, and AE25 by 200, 50 and, 25 curves, e.g. C_GV_G, respectively, which are randomly selected from the remaining 200 curves. While the machines of the 3 types of curves are different, for simplicity and when there is no confusion, they are all called AE200, AE50, or AE25.

The ML framework is shown in Fig. 2. The training is represented by the blue path. It contains 3 major parts. Firstly, a 5-layer AE is trained. The inputs to the AE are the scaled (StandardScaler) logarithmic values of the drain current, y_i. The outputs are the corresponding predicted values, $\hat{y}_i$. The hidden layers use RELU for activation. The first and last hidden layers have 40 nodes and the middle hidden layer has 3 nodes (h_0, h_1, h_2). Adam algorithm is used for

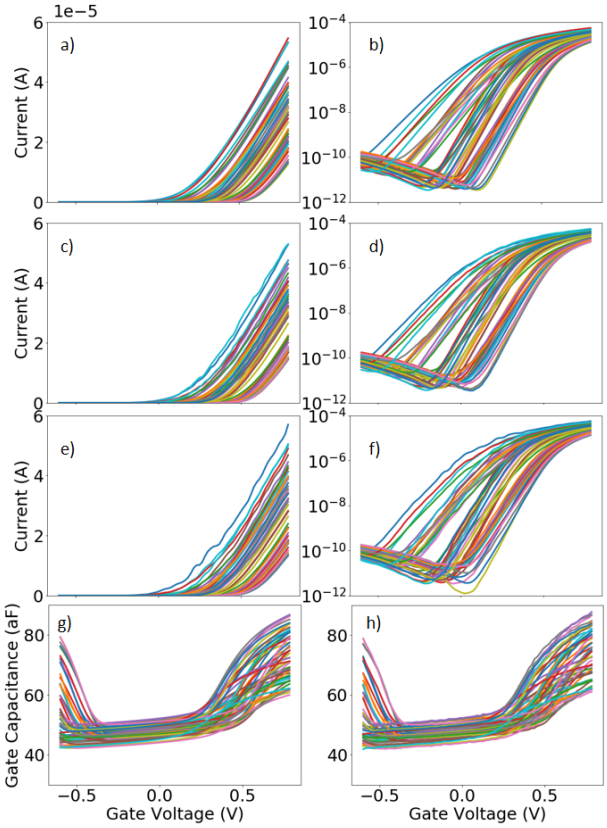

Fig. 3. The 50 test data I_DV_G's (a-f) in linear and logarithmic scales and their CV's (g-h). a) and b) are simulated by TCAD. c) and d) are predicted by AE200. e) and f) are predicted by AE50. g) are the CV simulated in TCAD. h) are the CV predicted by AE200.

optimization with 500 epochs. The performance metric is given by the Mean Squared Error (MSE) defined by

$$MSE = \frac{1}{n}\sum_{i=0}^{n-1}\left(y_i - \hat{y}_i\right)^2 \quad (1)$$

All AE training stops when MSE $\sim 10^{-3}$. Autoencoder is known to be able to perform efficient coding for signals [16]. Therefore, by using 3 hidden nodes in the middle hidden layer, we essentially encode a full *IV* or *CV* curve by 3 values (h_0, h_1, h_2). The number of hidden nodes is set to 3 because the variations in the curves are caused by 3 parameters (L_G, W_{TOP}, and WF). We then need to create a second machine to map (h_0, h_1, h_2) to (L_G, W_{TOP}, WF). However, due to the non-linear interaction between the parameters, (h_0, h_1, h_2) are not necessarily the linear combination of (L_G, W_{TOP}, WF) or vice versa. Indeed, we found that 3rd order polynomial regression is required to allow accurate prediction. Therefore, (h_0, h_1, h_2) is regressed against (L_G, W_{TOP}, WF) using 3rd order polynomial.

In the third part, to further improve the results, noise filtering is used (noisy autoencoder + Low Pass Filter, LPF).

As examples, Fig. 3a and b (Fig. 3g) show the 50 TCAD saturation I_DV_G's (C_GV_G) set aside for testing. Fig. 3c and d (Fig. 3h) show the saturation I_DV_G's ($C_GV_G's$) predicted by AE200. Although the result is not perfect, it is able to reconstruct the shape of the TCAD *IV* (*CV*) and capture various small and important features. In particular, it can capture the crossovers of various *IV*'s in the sub-threshold region

TABLE I
SATURATION I_D AND G_M PREDICTION ACCURACY (R^2) OF VARIOUS MACHINES

	SATURATION I_D @ V_G				G_{M2} @V_G
Machine	0.8V	0.4V	-0.2V	-0.6V	0.8V
AE200	0.97	0.94	0.98	0.85	0.78
AE50	0.97	0.98	0.91	0.66	0.66
AE25	0.95	0.84	0.83	0.68	0.58
AE50SD	0.93	0.86	0.93	0.69	0.51

TABLE II
OTHER METRICS PREDICTION ACCURACY (R^2) OF VARIOUS MACHINES

	CV Machine		IV Machine	
Machine	C_{high}[a]	C_{low}[b]	DIBL	SS[c]
AE200	0.99	1	0.98	0.97
AE50	0.84	0.93	0.84	0.87
AE25	0.38	0.83	0.43	0.75
AE50SD	0.89	0.94	0.90	0.81

[a]Gate capacitance at V_G=0.8V, [b]Gate capacitance at V_G=0V, [c]Subthreshold Slope defined in the region between 10^{-9}A to 10^{-6}A in saturation I_DV_G.

and the voltages at which BTBT tunneling starts dominating. And it also can capture the transition from accumulation to depletion to inversion regions in Fig. 3h. Fig. 4a-4d show the scatter plots of predicted I_{ON}, I_{OFF}, DIBL, and g_{m2}. It is worth noting that the coefficient of determination [17], R^2, higher than 0.85 can be achieved in g_{m2} (1st derivative of the curve) and I_{OFF} predictions, and R^2 is almost 1 for I_{ON}. By reducing the training data to only 50 (AE50), it still can capture the features mentioned earlier, although with larger errors (Fig. 3e and f). Fig. 4e-4h show the scatter plots of predicted I_{ON}, I_{OFF}, DIBL, and g_{m2} by AE50. It shows similar performance in predicting I_{ON} and I_{OFF} as AE200. For g_{m2}, it can achieve $R^2 = 0.61$. Note that since the machine takes milliseconds to perform one prediction, one may vary many combinations of (L_G, W_{TOP}, WF) to study the trend and effect of the parameters on the full IV to gain physical insight.

The machines trained by 25 data points, AE25, are also investigated. It still can capture the shape of the IV and CV but with more noise. It can predict the I_{ON}, I_{OFF}, C_{low}, and SS with R^2 of 0.82, 0.54, 0.83, and 0.75 respectively. Tables I and II show the R^2 of I_D and g_m predictions at various V_G and also other metrics important in the expertise domain.

IV. DISCUSSIONS

This study shows that AE can be used to capture the relationship between the design parameters and the IV/CV curves without learning any process/device physics *a priori* even with as few as 25-50 training curves. Our demonstration does not limit the training data to be TCAD curves. If we treat FinFET as a novel device, one can perform experiments with 4 wafers to generate the desired training curves (each wafer has different W_{TOP} while L_G and WF variations can be achieved through a special mask on the same wafer). Since in real experiment, there are other variables. To demonstrate that this approach is still possible when there are variations not known to the machine, the whole process is repeated with TCAD

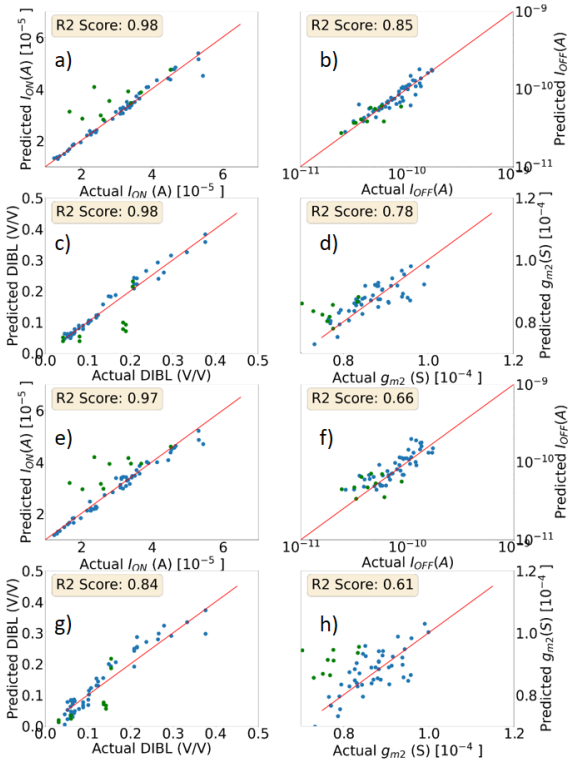

Fig. 4. Scatter plots (blue dots) showing the prediction of I_{ON}, I_{OFF}, DIBL, and g_{m2} in the test IV's by AE200, a) – d) and by AE50, e) – h). DIBL is calculated based on the difference between linear and saturation V_{TH}'s, where V_{TH} is defined as V_G@ $I_D = 10^{-7}$A. g_{m2} is defined at $V_G = 0.8$V in an 87.5mV interval. Green dots are the data with W_{TOP} = 3 or 4nm.

training curves generated with S/D carbon doping randomly varied by +/− 5%, which changes the strain distribution and currents. Tables I and II show that even with the extra variation, the machine (AE50SD) has a similar performance as AE50 *using the same set of 50 test curves*.

The purpose of the study is to train a machine with limited data in the parameter ranges of interest. It is found that we need to ensure the parameters do not concentrate at a certain corner (e.g. data with W_{TOP} = 5nm, 10nm, and 15nm is better than W_{TOP} = 10nm, 12nm, and 15nm). To further test if the machines can predict parameters outside of the training range (which is not the main purpose of this work), the machines are used to predict the IV's of W_{TOP} = 3nm and 4nm with various L_G and WF. The results are plotted as green dots in Fig. 4. It can be seen there are only a few outliers. It is also worthy to point out that DIBL is a quantity extracted from two different machines, each trained by saturation and linear IV's respectively and *separately*. The high R^2 (0.84 to 0.98) indicates that the physics is correctly captured in this methodology.

V. CONCLUSION

In this letter, we demonstrated the possibility of predicting full device IV and CV curves by training machines with limited training data (25-50) and minimal domain expertise. The machines are used to extract I_{ON}, I_{OFF}, g_m, DIBL, and SS with high R^2. It is expected the same methodology can be used to understand novel devices through training with limited experimental data before the underlying physics is understood.

REFERENCES

[1] Y. S. Bankapalli and H. Y. Wong, "TCAD augmented machine learning for semiconductor device failure troubleshooting and reverse engineering," in *Proc. Int. Conf. Simulation Semiconductor Processes Devices (SISPAD)*, Udine, Italy, Sep. 2019, pp. 1–4, doi: 10.1109/SISPAD.2019.8870467.

[2] C.-W. Teo, K. L. Low, V. Narang, and A. V.-Y. Thean, "TCAD-enabled machine learning defect prediction to accelerate advanced semiconductor device failure analysis," in *Proc. Int. Conf. Simulation Semiconductor Processes Devices (SISPAD)*, Udine, Italy, Sep. 2019, pp. 1–4, doi: 10.1109/SISPAD.2019.8870440.

[3] H. Y. Wong, M. Xiao, B. Wang, Y. K. Chiu, X. Yan, J. Ma, K. Sasaki, H. Wang, and Y. Zhang, "TCAD-machine learning framework for device variation and operating temperature analysis with experimental demonstration," *IEEE J. Electron Devices Soc.*, vol. 8, pp. 992–1000, 2020, doi: 10.1109/JEDS.2020.3024669.

[4] S. S. Raju, B. Wang, K. Mehta, M. Xiao, Y. Zhang, and H.-Y. Wong, "Application of noise to avoid overfitting in TCAD augmented machine learning," in *Proc. Int. Conf. Simulation Semiconductor Processes Devices (SISPAD)*, Kobe, Japan, Sep./Oct. 2020, pp. 351–354, doi: 10.23919/SISPAD49475.2020.9241654.

[5] J. Chen, M. B. Alawieh, Y. Lin, M. Zhang, J. Zhang, Y. Guo, and D. Z. Pan, "Powernet: SOI lateral power device breakdown prediction with deep neural networks," *IEEE Access*, vol. 8, pp. 25372–25382, 2020, doi: 10.1109/ACCESS.2020.2970966.

[6] H. Carrillo-Nuñez, N. Dimitrova, A. Asenov, and V. Georgiev, "Machine learning approach for predicting the effect of statistical variability in Si junctionless nanowire transistors," *IEEE Electron Device Lett.*, vol. 40, no. 9, pp. 1366–1369, Sep. 2019, doi: 10.1109/LED.2019.2931839.

[7] K. Mehta, S. S. Raju, M. Xiao, B. Wang, Y. Zhang, and H. Y. Wong, "Improvement of TCAD augmented machine learning using autoencoder for semiconductor variation identification and inverse design," *IEEE Access*, vol. 8, pp. 143519–143529, 2020, doi: 10.1109/ACCESS.2020.3014470.

[8] Z. Zhang, R. Wang, C. Chen, Q. Huang, Y. Wang, C. Hu, D. Wu, J. Wang, and R. Huang, "New-generation design-technology Co-optimization (DTCO): Machine-learning assisted modeling framework," in *Proc. Silicon Nanoelectron. Workshop (SNW)*, Kyoto, Japan, Jun. 2019, pp. 1–2, doi: 10.23919/SNW.2019.8782897.

[9] L. Zhang and M. Chan, "Artificial neural network design for compact modeling of generic transistors," *J. Comput. Electron.*, vol. 16, no. 3, pp. 825–832, 2017, doi: 10.1007/s10825-017-0984-9.

[10] J. Wang and W. Choi, "System and method for compact neural network modeling of transistors," U.S. Patent 0320366 A1, Oct. 8, 2020.

[11] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006, doi: 10.1126/science.1127647.

[12] *Sentaurus Process User Guide Version Q-2019.12*, Synop., Mountain View, CA, USA, Dec. 2019.

[13] *Sentaurus Device User Guide Version Q-2019.12*, Synop., Mountain View, CA, USA, Dec. 2019.

[14] *Application Note: Three-Dimensional Simulation of 14/16 nm FinFETs With Round Fin Corners and Tapered Fin Shape*, Synopsys, Mountain View, CA, USA, 2019.

[15] *Scikit-Learn Machine Learning in Python*. Accessed: Nov. 1, 2020. [Online]. Available: https://scikit-learn.org/stable/

[16] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE J.*, vol. 37, no. 2, pp. 233–243, Feb. 1991, doi: 10.1002/aic.690370209.

[17] Y. Dodge, "Coefficient of determination," in *The Concise Encyclopedia of Statistics*. New York, NY, USA: Springer, 2008, doi: 10.1007/978-0-387-32833-1_62.