Deep-Learning-Assisted Physics-Driven MOSFET Current-Voltage Modeling

Ming-Yen Kao, H. Kam, and Chenming Hu, *Life Fellow, IEEE*

***Abstract*— In this work, we propose using deep learning to improve the accuracy of the partially-physics-based conventional MOSFET current-voltage model. The benefits of having some physics-driven features in the model are discussed. Using a portion of the Berkeley Short-channel IGFET Common-Multi-Gate (BSIM-CMG), the industry-standard FinFET and GAAFET compact model, as the physics model and a 3-layer neural network with 6 neurons per layer, the resultant model can well predict *IV*, output conductance, and transconductance of a TCAD-simulated gate-all-around transistor (GAAFET) with outstanding 3-sigma errors of 1.3%, 4.1%, and 2.9%, respectively. Implications for circuit simulation are also discussed.**

***Index Terms*— BSIM-CMG, MOSFET, deep learning, compact model**

I. Introduction

Accurate and fast current-voltage (*I*–*V*) models are critical for integrated circuit simulation. Compact models are traditionally derived from physics [1-3]; they involve solving nonlinear differential equations where closed-form solutions may not exist. Furthermore, as transistors are scaled, equations that account for nanoscale effects and non-idealities become more mathematically complicated. By sacrificing accuracy, semi-empirical models with fitting parameters are commonly used. To improve model accuracy, pure look-up-table [4] or deep-learning-based transistor models [5-11] have recently been proposed in which the measured *IV* data were tabulated or used for model training. For look-up-table, dense V_{DS} and V_{GS} sampling is required for accurate gradient approximation, which results in a large number of model parameters. Furthermore, interpolation in look-up table is prone to data noise. On the other hand, deep-learning based models can prevent overfitting by using well-developed techniques such as regularization. However, many deep-learning approaches result in non-physical behavior such as non-zero current at V_{DS} = 0V and/or asymmetric *IV* model. Furthermore, the modeling of variation is complex in the pure deep learning-based model.

To alleviate these, a deep-learning assisted, partially-physics-based, *IV* model that combines the advantages of both strategies is presented in this paper. The rest of this article is organized as follows. Section II sets up the model framework and discusses the physics-driven requirements. Then, sections III and IV present the neural network design and dataset, respectively. Using Berkeley short-channel insulated-gate field-effect transistor (IGFET) Common-Multi-Gate (BSIM-CMG) and the simulated *IV* characteristics of 12nm gate-length gate-all-around (GAAFET) technology as the physics model and the dataset, respectively, section V presents the modeling results. The implication of circuit simulations is also discussed. Finally, we conclude this paper in Section VI.

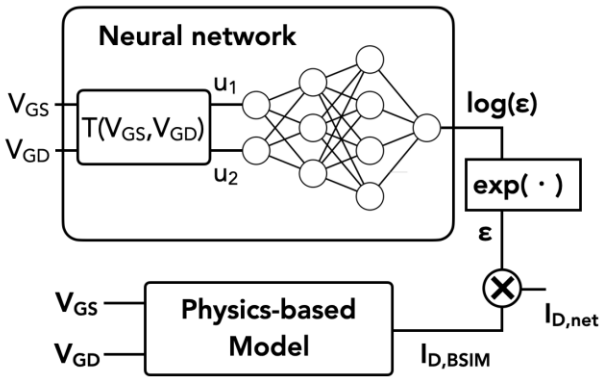
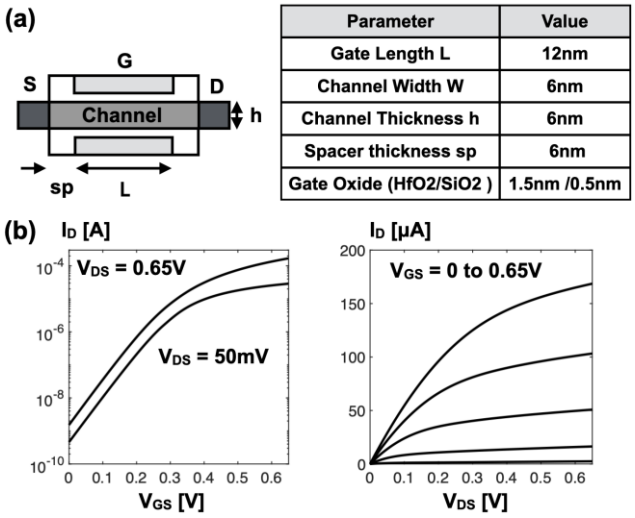


Fig 1. Deep-learning-assisted *IV* model architecture. Input (V_{GS},V_{GD}) is first transformed by *T* (Eq. 2) to satisfy the Gummel symmetry.



Parameter	Value
Gate Length L	12nm
Channel Width W	6nm
Channel Thickness h	6nm
Spacer thickness sp	6nm
Gate Oxide (HfO2/SiO2)	1.5nm /0.5nm

Fig 2. (a) Cross-sectional schematic of the simulated GAAFET structure and (b) simulated transfer and output *IV* characteristics.

This work was supported by the Berkeley Device Modeling Center, University of California at Berkeley, CA 94720 USA. (Corresponding author: Ming-Yen Kao)

M.-Y. Kao and C. Hu are with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: mingyenkao@berkeley.edu).

II. Deep-learning-assisted IV model

We select the core BSIM-CMG [1] ($I_{D,BSIM}$) as the starting physics-based model. The charge density model, the transport model, and the terminal charge model are included in the core model. Then, a bias-dependent correction function $\varepsilon(V_{GS},V_{GD})$ is introduced to account for the aforementioned non-idealities not included in the core model:

$$I_D(V_{GD},V_{GD}) = I_{D,BSIM}(V_{GS},V_{GD}) \times \varepsilon(V_{GS},V_{GD}) \quad (1)$$

in which ε is to be trained by deep learning. Before doing so, we first discuss the physics-driven requirements on I_D and ε.

Physics requires I_D to be zero when V_{DS}=0. Physics-based $I_{D,BSIM}$ and therefore (1) always satisfy this requirement.

Furthermore, since transistors are symmetric devices, the direction of the current flow change if we swap the source and drain voltage, i.e. $I_D(V_{GS},V_{GD}) = - I_D(V_{GD},V_{GS})$, substituting this condition into (1), we get:

$$I_{D,BSIM}(V_{GS},V_{GD}) \times \varepsilon(V_{GS},V_{GD}) = -I_{D,BSIM}(V_{GD},V_{GS}) \times \varepsilon(V_{GD},V_{GS})$$

Since BSIM-CMG is symmetric, i.e. $I_{D,BSIM}(V_{GS},V_{GD}) = - I_{D,BSIM}(V_{GD},V_{GS})$, the correction function must be symmetric: $\varepsilon(V_{GS},V_{DS}) = \varepsilon(V_{GD},V_{GS})$

This condition can be satisfied if we transform input (V_{GS},V_{GD}) by T before feeding it into the neural network:

$$(u_1,u_2) = T(V_{GS},V_{GD})$$
$$\text{where } u_1 = V_{GS} + V_{GD} \text{ and } u_2 = (V_{GS} - V_{GD})^2 \quad (2)$$

To prove that T is symmetric, we simply swap V_D and V_S:

$$(u_1^*,u_2^*) = T(V_{GD},V_{GS})$$
$$\text{where } u_1^* = V_{GD} + V_{GS} \text{ and } u_2^* = (V_{GD} - V_{GS})^2$$

Therefore $u_1^* = u_1$, $u_2^* = u_2$ and $T(V_{GD},V_{GS}) = T(V_{GS},V_{GD})$.

Finally, I_D must be infinitely differentiable with respect to V_{GS} and V_{DS}. The core model is physics-based and already meets this requirement. As a result, only the correction function, i.e., the activation function, must be infinitely differentiable. In addition, analog circuit applications also require accurate prediction for the output conductance *gds* and transconductance *gm*.

Another advantage of having a physics-based core model in our approach is that the effects of many transistor design parameters (e.g., gate length *L*, width *W*, channel thickness *h*, doping concentration, etc.) can be assessed. This insight is valuable for first order manufacturing process variations modeling and circuit design. With this, we will discuss the neural network design in the next section.

III. Neural network design

Fig. 1 shows the neural network design used in this work. Input (V_{GS},V_{GD}) is first transformed with T before feeding into the neural network. After comparing several neural network designs, a 3-hidden-layer network with 6 neurons per layer is found to provide good model accuracy and fast training and is selected. Hyperbolic tangent function *tanh* is used as the activation function for the input and hidden layers due to its infinite differentiability. Since ε is always positive, the network is designed to train $log(\varepsilon)$ for improved convergence. The resulting ε is multiplied by the physics model $I_{D,BSIM}$ to obtain the final $I_{D,net}$, where the subscript *net* denotes the network output.

The cost function J is defined as the average root mean square relative errors in I_D, *gds*, and *gm*:

$$J = \frac{1}{3}\left[\sqrt{\frac{1}{m}\sum_{i=1}^{m} RE^2\left(I_{D,net}^{(i)}, I_{D,data}^{(i)}\right)} + \sqrt{\frac{1}{m}\sum_{i=1}^{m} RE^2\left(gds_{net}^{(i)}, gds_{data}^{(i)}\right)} + \sqrt{\frac{1}{m}\sum_{i=1}^{m} RE^2\left(gm_{net}^{(i)}, gm_{data}^{(i)}\right)}\right] \quad (3)$$

where m is the training set sample size, i is the *ith* sample, and subscript *data* denotes measured data. *RE* is the relative error:

$$RE(x_{net},x_{data}) = (x_{net} - x_{data})/(x_{data} + \delta)$$

where δ is a user-defined infinitesimal parameter that prevents numerical overflow. δ = 1E-10 is used in this study.

The central difference method is used to estimate the derivatives. We first define a perturbation voltage *dv*. During training, we feed (V_{GS},V_{GD}) together with $(V_{GS}+dv,V_{GD}+dv)$, $(V_{GS}-dv,V_{GD}-dv)$, $(V_{GS},V_{GD}-dv)$, $(V_{GS},V_{GD}+dv)$ into the neural network to compute $I_{D,net}(V_{GS},V_{GD})$, $I_{D,net}(V_{GS}+)$, $I_{D,net}(V_{GS}-)$, $I_{D,net}(V_{DS}+)$, $I_{D,net}(V_{DS}-)$ respectively. We then approximate the partial derivatives by:

$$gm_{net} \approx \left[I_{D,net}(V_{GS}+) - I_{D,net}(V_{GS}-)\right]/(2dv)$$
$$gds_{net} \approx \left[I_{D,net}(V_{DS}+) - I_{D,net}(V_{DS}-)\right]/(2dv)$$

With the network design established, we will discuss data preparation in the next section.

IV. Dataset

Ideally, we would use the measured device data from the silicon foundry as the dataset. In this work, it suffices to use the finite element TCAD tool, SENTARUS™, to generate the training *I-V* dataset ($I_{D,data}$). Fig. 2 shows the GAAFET structure and the simulated *IV* characteristics calibrated to the IRDS "1.5nm node" technology [12]. Physics models such as the Philips unified mobility model [13], thin layer mobility, Fermi Dirac Distribution, and correction due to Fermi statistics and quantization effect are included in the simulation.

The same device geometry, including the channel length, width, and thickness, together with the equivalent oxide thickness and gate work function, all of which would be known to the silicon foundry, are also used in the core BSIM-CMG model. Since default parameters are used for all other model parameters without calibration, it results in a large mean and standard deviation of $I_{D,data}/I_{D,BSIM}$ 6.2 and 5.1, respectively.

The neural network is trained with a dataset of ($V_{GS,data}$, $V_{DS,data}$, $I_{D,data}$) with $V_{GS,data}$ and $V_{DS,data}$ range from 0V to 0.65V with 50mV increment. Then, we test the model using a dataset with the same voltage ranges but a smaller voltage increment of 5mV. Training and testing dataset size are therefore 14×14=196 and 131×131=17161, respectively, resulting in a training-to-test-set ratio of ~1/100. Partial derivatives, gds_{data} and gm_{data}, are computed using the central difference formula with dv = 10mV.

TensorFlow with adaptive moment estimation ("Adam") optimization with a learning rate of 1E-4 is used for training.

Adam optimization is a gradient-descent-based optimization algorithm in which the adaptive learning rate is estimated based on the first and second moments of the gradients [14]. Given the small training size of < 200, a single batch is fed into the neural network.

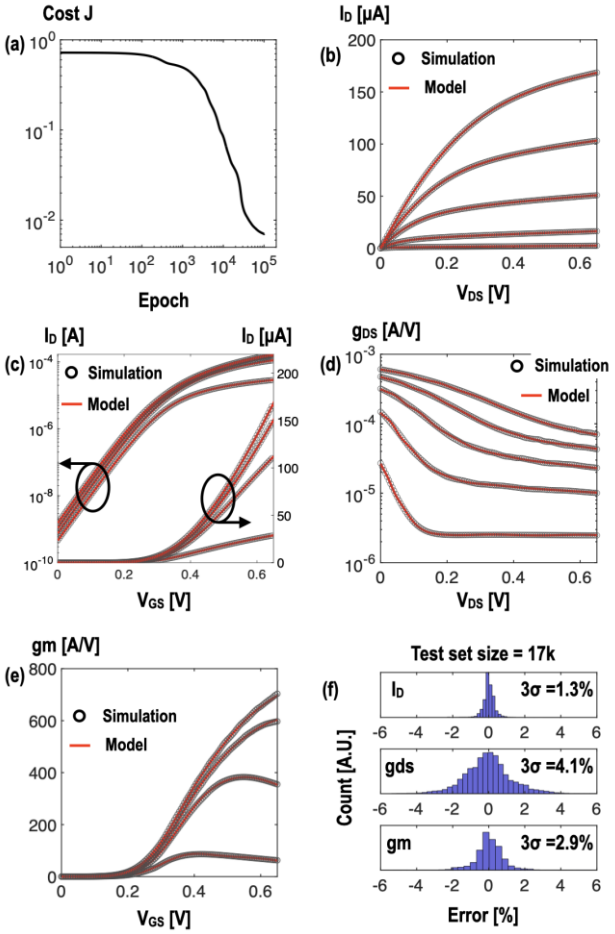


Fig 3. (a) Training error vs epoch. Model well-matches to the test dataset in (b) output and (c) transfer characteristic (d) output conductance, and (e) transconductance. (f) Percentage error distribution in I_D, *gds* and *gm*.

V. Results and Discussions

Fig. 3 shows the cost *vs.* epoch and the testing results for the 3-layer neural network. The trained model well predicts the test *IV*, *gds*, and *gm* to 3σ error of 1.3%, 4.1%, and 2.9%, respectively. Fig. 4 highlights the importance of the relative *gds* and *gm* error terms in the cost function. 3σ errors in *gds* and *gm* increase to 14% and 3.9%, respectively, without them. As shown in Fig. 5, the trained model satisfies the Gummel symmetry [15]. This is unsurprising because ε is symmetric by design, as earlier discussed.

Fig 6 shows the model implementation for circuit simulation. By rewriting (1) as $I_D = I_{D,BSIM} + I_{D,BSIM} \times (\varepsilon - 1)$, one can model the neural network as an add-on current source. Since the core BSIM-CMG (~100 lines of codes) is 10× shorter than the full version counterparts (~ 2000 lines), the deep-learning approach discussed herein reduces the BSIM model calculation time. The sum of that and the added neural network operations, which include matrix multiplications and hyperbolic tangent functions, is expected to be substantially faster than calculating the full BSIM-CMG *I-V* model if the neural network operations can be parallelized using specialized hardware such as the graphical processing units (GPU). Our proposed approach also accelerates process design kit (PDK) development as the parameter extraction process is intrinsic to the network training. All these make our approach very attractive for *I-V* modelling.

VI. Conclusions

In this work, hybrid analytical and deep-learning-assisted MOSFET *IV* modeling is proposed. The reasons for using a physics-based core are discussed. Using a 12nm gate length GAAFET technology as an example, our model utilizing a 3-layer neural network with 18 neurons can predict the simulated *IV* to 1.3% in 3-sigma error, and it satisfies Gummel symmetry. To conclude, deep neural network coupled with physics-based model is a very intriguing prospect for modeling next generation MOSFETs. Work in this vein is dawning and the possibilities are endless.

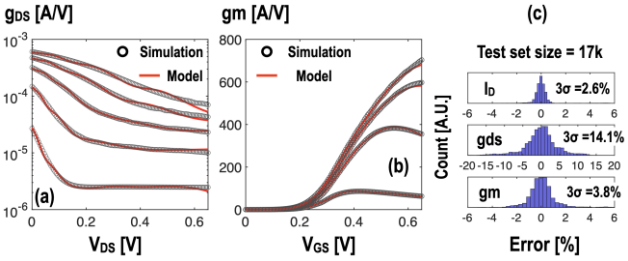


Fig. 4. Without the relative *gds* and *gm* errors terms in the cost function, Model shows larger prediction error in (a) *gds* and (b) *gm*. (c) Percent error distribution in I_D, *gds* and *gm*.

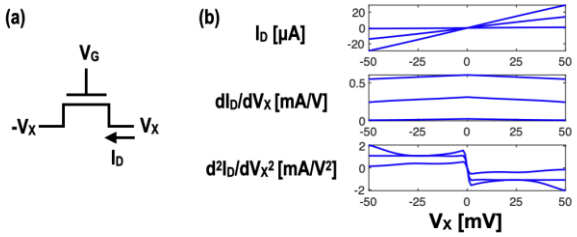


Fig. 5. (a) Gummel symmetry test setup. (b) I_D, first and second derivative of I_D with respect to V_X.

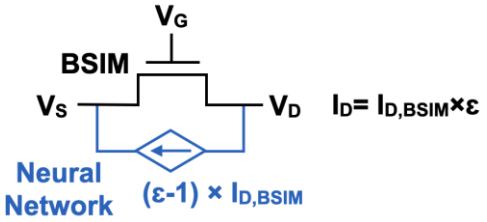


Fig. 6. (a) Model implementation for circuit simulations

References

[1] J. P. Duarte et al., "BSIM-CMG: Standard FinFET compact model for advanced circuit design," *ESSCIRC Conference 2015 - 41st European Solid-State Circuits Conference (ESSCIRC)*, 2015, pp. 196-201, doi: 10.1109/ESSCIRC.2015.7313862.

[2] C. C. Enz and E. A. Vittoz, "Charge-Based MOS Transistor Modeling: The EKV Model for Low-Power and RF IC Design." *John Willey, New York* (2006).

[3] G. Gildenblat et al. "PSP Model." *Department of Electrical Engineering, The Pennsylvania State University and Philips Research,* Aug. 2005

[4] P. G. A. Jespers and B. Murmann, Systematic Design of Analog CMOS Circuits, *Cambridge University Press*, 2017.
[5] M. Li, O. İrsoy, C. Cardie and H. G. Xing, "Physics-Inspired Neural Networks for Efficient Device Compact Modeling," in *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, vol. 2, pp. 44-49, Dec. 2016, doi: 10.1109/JXCDC.2016.2636161.
[6] L. Zhang and M. Chan. "Artificial neural network design for compact modeling of generic transistors," in *Journal of Computational Electronics*, vol. 16, no. 3, pp. 825-832, Sep. 2017, doi: 10.1007/s10825-017-0984-9
[7] K. Mehta and H. -Y. Wong, "Prediction of FinFET Current-Voltage and Capacitance-Voltage Curves Using Machine Learning With Autoencoder," in *IEEE Electron Device Letters*, vol. 42, no. 2, pp. 136-139, Feb. 2021, doi: 10.1109/LED.2020.3045064.
[8] J. Xu and D. E. Root, "Advances in artificial neural network models of active devices," *2015 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, 2015, pp. 1-3, doi: 10.1109/NEMO.2015.7415102.
[9] H. B. Hammouda, M. Mhiri, Z. Gafsi, and K. Besbes, ‘‘Neural-based models of semiconductor devices for SPICE simulator,’’ in *American Journal of Applied Sciences*, vol. 5, no. 4, pp. 785–791, Apr. 2008, doi: 10.3844/ajassp.2008.385.391
[10] Fang Wang and Qi-Jun Zhang, "Knowledge-based neural models for microwave design," in *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 12, pp. 2333-2343, Dec. 1997, doi: 10.1109/22.643839.
[11] J. Wang, Y. -H. Kim, J. Ryu, C. Jeong, W. Choi and D. Kim, "Artificial Neural Network-Based Compact Modeling Methodology for Advanced Transistors," in *IEEE Transactions on Electron Devices*, vol. 68, no. 3, pp. 1318-1325, March 2021, doi: 10.1109/TED.2020.3048918.
[12] IRDS 2020 Edition, 2020. [https://irds.ieee.org/editions/2020]
[13] D. B. M. Klaassen, "A unified mobility model for device simulation," *International Technical Digest on Electron Devices*, 1990, pp. 357-360, doi: 10.1109/IEDM.1990.237157.
[14] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).
[15] C. C. Mcandrew, "Validation of MOSFET model Source–Drain Symmetry," in *IEEE Transactions on Electron Devices*, vol. 53, no. 9, pp. 2202-2206, Sept. 2006, doi: 10.1109/TED.2006.881005.