Solid-State Electronics 201 (2023) 108580

Contents lists available at ScienceDirect

Solid State Electronics

journal homepage: www.elsevier.com/locate/sse

Graph-based Compact Modeling (GCM) of CMOS transistors for efficient parameter extraction: A machine learning approach[☆]

Amol D. Gaidhane [*], Ziyao Yang, Yu Cao [*]

School of ECEE, Arizona State University, Tempe, AZ, USA

ARTICLE INFO

Keywords:
Graph-based Compact Model (GCM)
FinFETs
Circuit simulation
Machine learning
Graph Neural Networks (GNNs)

ABSTRACT

Parameter extraction of compact transistor models is an expensive process, heavily relying on engineering knowledge and experience. To automate such a process, we propose a novel approach, Graph-based Compact Model (GCM), that integrates physical modeling and data-driven learning. GCM utilizes Graph Neural Networks (GNNs) to establish the model structure, while retaining the physicality in compact models. We implement our GCM in Verilog-A to support circuit simulations. As demonstrated with an academic 7 nm FinFET PDK, the new approach automatically generates a GCM model within a minute, and achieves excellent accuracy and efficiency in SPICE.

1. Introduction

Compact modeling of CMOS transistors is the essential bridge between silicon manufacturing and circuit simulations [1]. To capture the complexity of device physics, an increasing number of model parameters have to be introduced in the compact model, posing enormous challenges in parameter extraction and simulation efficiency. To extract model parameters of CMOS transistors, a machine-learning based method has been proposed [2,3]. Further, the artificial neural networks (ANNs) have been used for the compact modeling of generic transistor behaviors [4,5]. However, without explicit physical meaning, such artificial neural networks (ANNs) impede model scalability and efficiency. Therefore, it is imperative to develop a compact, scalable, and computationally efficient model for CMOS transistors.

Recently, GNNs were proposed to model dynamic systems [6–8]. GNNs describe each physical component as the graph node, and their interactions as edges [9]. Based on GNNs, we introduce our new method, GCM, for compact transistor modeling. For key physical parameters, they are captured by non-linear models and embedded into the graph nodes. For many other fitting parameters, they are replaced by neural networks to connect the nodes together. Parameter extraction, i.e., training of GCM, is data driven through back propagation, with appropriate constraints on physical parameters to improve the robustness. In Section 2, we explain the GCM modeling approach in detail. We implement our GCM model in Verilog-A code to demonstrate the circuit simulation in commercial SPICE simulator. The model validation with an academic 7 nm FinFET PDK is demonstrated in Section 3. Further, we demonstrate the convergence of several model parameters during the training process. Finally, we demonstrate the inverter VTC in SPICE simulator, and compare the simulation time and model parameters with the BSIM model.

2. Model development

The new method of GCM converts a conventional model into a set of nodes for key physical parameters, and connects them into a directional graph. Fig. 1 presents the graph structure to model a transistor, with nodes for selected physical parameters and edges for their relationships as shown in Fig. 1. Similar as other compact transistor models, GCM receives the input features as an vector (e.g., voltages, geometries, etc.), and predicts the output features (e.g., I_{ds} and its derivatives). In this work, we develop a GCM to obtain static characteristics of FinFET at room temperature. To obtain static characteristics, we use V_{ds}, V_{ds} as dynamic inputs and L, T_{OX}, T as static inputs, whereas I_{ds} and its derivatives i.e, G_m and G_{ds} as output nodes.

There are two basic operations in GCM, aggregation and transformation. The aggregation step computes the node value from the input vector or its nearest neighbors, depending the graph structure. The transformation step applies the update function to generate a new value for each node and the output vector. The update function can be physical equations, if the physics is clear, or neural networks for fitting. We keep as many nonlinear relations as possible in GCM to minimize the model size of the neural network. As with GNNs, the

[☆] The review of this paper was arranged by Francisco Gamiz.

[*] Corresponding authors.

E-mail addresses: agaidhan@asu.edu (A.D. Gaidhane), yu.cao@asu.edu (Y. Cao).

https://doi.org/10.1016/j.sse.2022.108580

Available online 3 January 2023

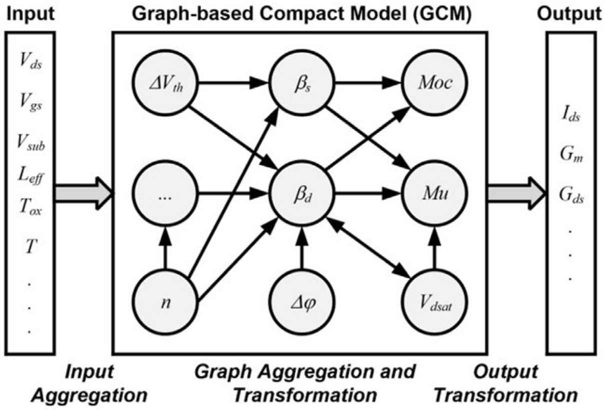


Fig. 1. The structure of graph-based compact model of FinFET. The model receives the input features such as voltages, geometries, etc. as an vector and predicts the drain current (I_{ds}) and its derivatives as output features.

GCM aggregates data based on the graph structure. Additionally, GCM update function is capable of combining physical equations and neural networks. The GCM transforms learned parameters and helps in the prediction of I_{ds}. Finally, we use derivatives of I_{ds} to improve the accuracy of our model.

To demonstrate GCM, we start from a long-channel surface-potential based model of FinFET [10]. The surface potential for the long-channel double-gate FET (DGFET) is given in [10] as

$$\psi(x) = V - \frac{2kT}{q} \ln\left[\frac{t_{si}}{2L_{Di}\beta} \cos\left(\frac{2\beta}{t_{si}}x\right)\right] \quad (1)$$

where V is the electron quasi-fermi potential and it is equal to V_s at the source end and V_d at the drain end. q is the electronic charge, k is the Boltzmann constant, t_{si} is the thickness of the channel, $L_{Di} = \sqrt{2\varepsilon_{si}kT/q^2 n_i}$ is the intrinsic Debye length, ε_{si} is the permittivity of silicon, n_i is the intrinsic carrier density. β is an intermediate parameter which is equal to β_S and β_D at the source and drain side, respectively. As shown in Fig. 1, β_S and β_D are the intermediate nodes in which the equations for are solved after applying boundary conditions to the Poisson's equation using the asymptotic compact modeling approach [11].

In Fig. 1, $\Delta\phi$ is a node for the work-function difference. Moreover, for the short channel effects (SCEs), we add certain nodes, such as n for sub-threshold slope. Similarly, ΔV_{th} is a node for the change in threshold voltage, Moc is for channel length modulation (CLM), Mu is for the effective mobility, and V_{dsat} is for the drain saturation voltage, etc. Overall, the nodes preserve important physics for the FinFET transistor.

Fig. 2 presents an example of GCM feature transformation for the node n, which is a combination of physical equations and a multi-layer perceptron (MLP). In BSIM-CMG [12], the sub-threshold slope factor (n) is calculated as

$$n = \Theta_{SS}\left(\frac{1 + CIT_i + Cdsc}{2C_{si} \parallel C_{ox}}\right) \quad (2)$$

where Θ_{SS} is swing temperature coefficient, C_{si} and C_{ox} are the channel and oxide capacitance, respectively. The $Cdsc$ is given as

$$Cdsc = \frac{0.5}{cosh\left(DVT1SS_i.\frac{L_{eff}}{\lambda}\right) - 1} \times \left(CDSC + CDSCD_a.V_{dsx}\right) \quad (3)$$

where, CIT_i is a parameter for interface trap, $DVT1SS_i$ is sub-threshold swing exponent coefficient parameter, $CDSC$ is a parameter for coupling capacitance between S/D and channel, and $CDSCD_a$ is a parameter for drain-bias sensitivity of $CDSC$. Thus, to model the sub-threshold slope, BSIM-CMG takes four fitting parameters. To model sub-threshold slope node into our GCM, we use the combination of physical equations and MLP as shown in Fig. 2. To model the drain bias dependency as observed in (3), we use MLP to obtain $n_{V_{ds}}$. From (3), $n_{L_{eff}}$ has a specific dependence on the channel length (L_{eff}) [12], we adopt the compact model to keep the physicality and minimize model fitting. Similarly, we use the combination of MLPs and the physical equations for the remaining node. Finally, the drain current and its derivatives as output feature are calculated using the drift-diffusion formulation. In GCM, by replacing fitting parameters and related equations with MLPs, we will leverage model training of neural networks that are data driven and differentiable. Therefore, GCM automatically achieves high fitting accuracy and continuity for circuit simulation. To train our model, we use shallow MLPs with 1 or 2 hidden layers. Each layer contains 4 or 8 neurons, trained with Adam optimizer using PyTorch with a learning rate of 0.003. We use the batch size of 50. Further, we use Gaussian Error Linear Unit (GELU) as an activation function to train our model. In our model, we use 4 MLPs for 5 different physical parameters i.e, *n*, ΔV_{TH}, $\Delta\phi$, V_{dsat}, *Mu* and *Moc*.

Fig. 3 shows a two-phase training procedure for efficient extraction of the GCM model, balancing multiple objectives in the loss function. In Phase 1, we use $I_{ds} - V_{gs}$ data to train the model, with G_m in the

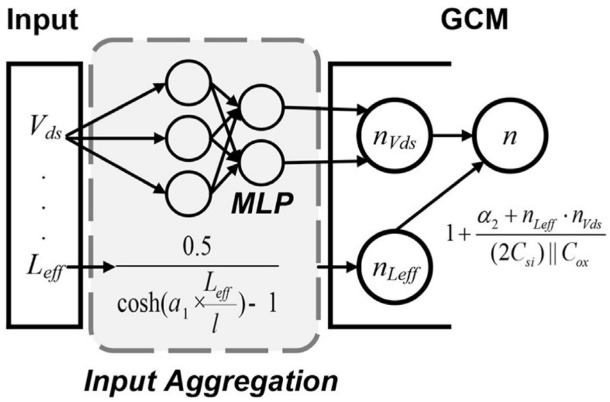

Input
GCM
V_{ds}
.
.
.
L_{eff}
MLP
$\frac{0.5}{\cosh(a_1 \times \frac{L_{eff}}{l}) - 1}$
n_{Vds}
n
n_{Leff}
$1 + \frac{\alpha_2 + n_{Leff} \cdot n_{Vds}}{(2C_{si}) \parallel C_{ox}}$
Input Aggregation

Fig. 2. Combination of a multi-layer perceptron (MLP) and the physical equation in aggregation to obtain the value of sub-threshold factor (n).

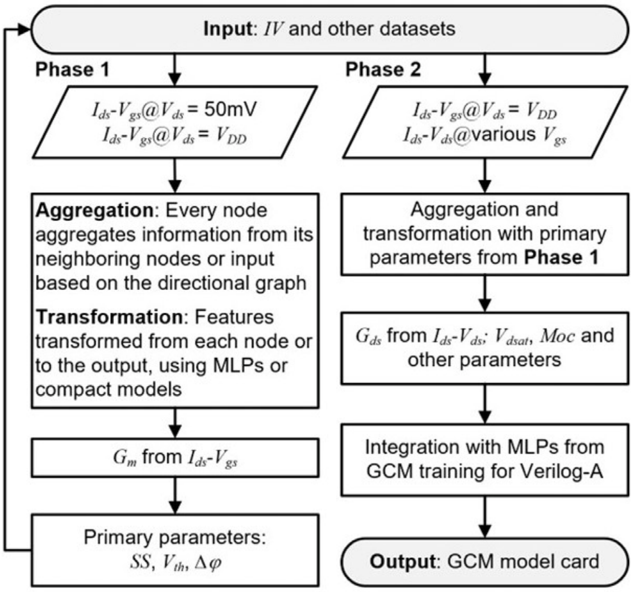
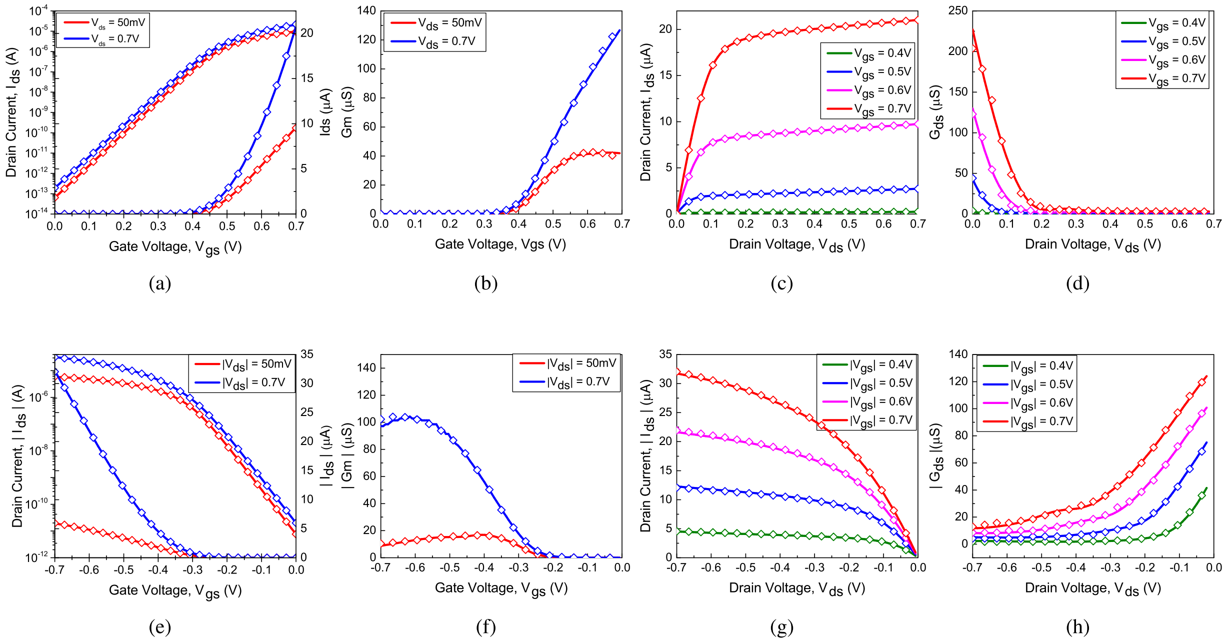

Input: IV and other datasets
Phase 1
Phase 2
I_{ds}-V_{gs}@V_{ds} = 50mV
I_{ds}-V_{gs}@V_{ds} = V_{DD}
I_{ds}-V_{gs}@V_{ds} = V_{DD}
I_{ds}-V_{ds}@various V_{gs}
Aggregation: Every node aggregates information from its neighboring nodes or input based on the directional graph
Transformation: Features transformed from each node or to the output, using MLPs or compact models
G_m from I_{ds}-V_{gs}
Primary parameters: SS, V_{th}, $\Delta\varphi$
Aggregation and transformation with primary parameters from Phase 1
G_{ds} from I_{ds}-V_{ds}; V_{dsat}, Moc and other parameters
Integration with MLPs from GCM training for Verilog-A
Output: GCM model card

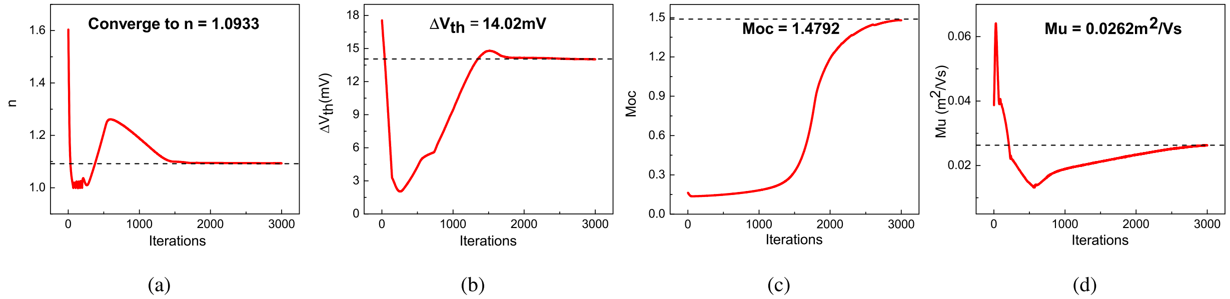
Fig. 3. A two-phase training program for GCM for efficient extraction of GCM model.



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 4. Validation of our GCM model (shown by solid lines) for n-FinFET with 7 nm PDK (shown by symbols) (a) I_{ds}–V_{gs} characteristics at V_{ds} = 50 mV and V_{ds} = 0.7 V, (b) Trans-conductance at V_{ds} = 50 mV and V_{ds} = 0.7 V, (c) I_{ds}–V_{ds} characteristics at multiple V_{gs}, and (d) Output conductance at multiple V_{gs}. Validation of our GCM model for p-FinFET with 7 nm PDK (e) I_{ds}–V_{gs} characteristics at $|V_{ds}|$ = 50 mV and $|V_{ds}|$ = 0.7 V, (f) Trans-conductance at $|V_{ds}|$ = 50 mV and $|V_{ds}|$ = 0.7 V, (g) I_{ds}–V_{ds} characteristics at multiple V_{gs}, and (h) Output conductance at multiple V_{gs}.



(a) (b) (c) (d)

Fig. 5. Training curves of GCM parameters (a) Sub-threshold slope parameter (n),(b) Threshold voltage parameter (ΔV_{th}), (c) Channel length modulation parameter (Moc), and (d) Effective mobility parameter (Mu). It only takes 31.33 s to complete the training of parameters.

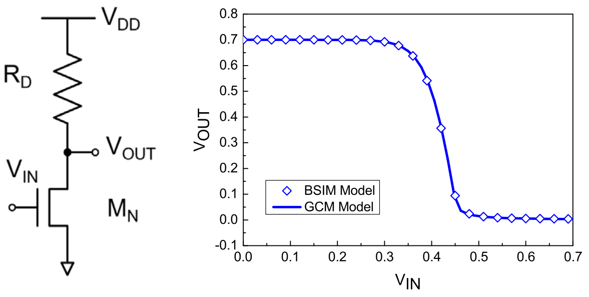
loss function to improve the accuracy. GCM extracts V_{ds} independent parameters in this phase. In Phase 2, we freeze the learned parameters in Phase 1 and extract V_{ds} dependent parameters. We use IV data generated from a BSIM-CMG based FinFET model at 7 nm [13] to train the GCM model.

3. Results and discussion

Fig. 4 validates the main electrical characteristics of the n-FinFET and p-FinFET between the 7 nm FinFET PDK BSIM-CMG model and GCM. The reference data set is generated in HSPICE simulator. The total number of data points required to train our model at different bias conditions are 200. Fig. 4(a) shows the I_{ds}–V_{gs} characteristics at V_{ds} = 50 mV and V_{ds} = 0.7 V and its derivatives shown in Fig. 4(b). Where Fig. 4(c) and (d) show I_{ds}–V_{ds} characteristics at multiple V_{gs} and its derivatives. Similarly, the validation of electrical characteristics for p-FinFET are shown in Fig. 4(e)–(h). The GCM model accurately captures the drain current and its derivatives for both n-FinFET and p-FinFET.

Fig. 5 illustrates the training curves for several GCM model parameters. Fig. 5(a) shows convergence of sub-threshold slope parameter (n) which converge to its final value to 1.0933 very efficiently. Similarly, convergence of threshold voltage parameter (ΔV_{th}), channel length modulation parameter (Moc), and effective mobility parameter (Mu) are shown in Fig. 5(b)–(d).

Fig. 6 demonstrates circuit simulation with GCM through Verilog-A. Fig. 6(b) compares the voltage transfer curve (VTC) NMOS resistive load inverter (as shown in Fig. 6(a)). The table shown in Fig. 6(c) compares the SPICE simulation time and the number of model parameters of GCM with the BSIM model. For the 7 nm FinFET, GCM has 113 parameters and extracts all model parameters in 31.33 s. This is significantly faster than the conventional extraction process, which usually takes hours to days. GCM achieves similar accuracy as BSIM, with shorter simulation time in SPICE.



(a) (b)

	BSIM	GCM
SPICE Simulation Time (ms)	12.745	5.284
Number of Model Parameters	905*	113

*This is for a complete BSIM-CMG model card

(c)

Fig. 6. (a) NMOS resistive load inverter. (b) Comparison of inverter voltage transfer curve (VTC) obtained from our GCM and BSIM model. (c) Comparison of SPICE simulation time and number of model parameters of GCM with the BSIM model.

4. Conclusion

We propose a novel graph-based compact model which is physical and efficient in parameter extraction. The graph structure preserves key physical models, supports automatic learning from data, and is flexible to incorporate more advanced effects (e.g., cryogenic effects).

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yu Cao reports financial support was provided by US Department of Energy. Yu Cao reports financial support was provided by Defense Advanced Research Projects Agency.

Data availability

No data was used for the research described in the article.

Acknowledgment

This work is partially supported by the Defense Advanced Research Projects Agency (DARPA), United States under the program of Low Temperature Logic Technology (LTLT). It is also supported in part by the U.S. Department of Energy, through the Office of Advanced Scientific Computing Research's "Data-Driven Decision Control for Complex Systems (DnC2S)" project.

References

[1] Gildenblat G. Compact modeling: Principles, techniques and applications. Netherlands: Springer; 2010.
[2] Mehta K, Wong H-Y. Prediction of FinFET current-voltage and capacitance-voltage curves using machine learning with autoencoder. IEEE Electron Device Lett 2020;42(2):136–9.
[3] Kao M-Y, Chavez F, Khandelwal S, Hu C. Deep learning-based BSIM-CMG parameter extraction for 10-nm FinFET. IEEE Trans Electron Devices 2022.
[4] Zhang L, Chan M. Artificial neural network design for compact modeling of generic transistors. J Comput Electron 2017;16(3):825–32.
[5] Wang J, Kim Y-H, Ryu J, Jeong C, Choi W, Kim D. Artificial neural network-based compact modeling methodology for advanced transistors. IEEE Trans Electron Devices 2021;68(3):1318–25.
[6] Kazemi SM, Goel R, Jain K, Kobyzev I, Sethi A, Forsyth P, et al. Representation learning for dynamic graphs: A survey. J Mach Learn Res 2020;21(70):1–73.
[7] Xie Y, Li C, Yu B, Zhang C, Tang Z. A survey on dynamic network embedding. 2020, arXiv preprint arXiv:2006.08093.
[8] Barros CD, Mendonça MR, Vieira AB, Ziviani A. A survey on embedding dynamic graphs. ACM Comput Surv 2021;55(1):1–37.
[9] Skarding J, Gabrys B, Musial K. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey. IEEE Access 2021;9:79143–68.
[10] Taur Y, Liang X, Wang W, Lu H. A continuous, analytic drain-current model for DG MOSFETs. IEEE Electron Device Lett 2004;25(2):107–9.
[11] Gaidhane AD, Pahwa G, Verma A, Chauhan YS. Compact modeling of drain current in double gate negative capacitance MFIS transistor. In: 2018 4th IEEE international conference on emerging electronics. ICEE, IEEE; 2018, p. 1–5.
[12] Chauhan YS, Lu D, Venugopalan S, Khandelwal S, Duarte JP, Paydavosi N, et al. FinFET modeling for IC simulation and design: Using the BSIM-CMG standard. Academic Press; 2015.
[13] Clark LT, Vashishtha V, Shifren L, Gujja A, Sinha S, Cline B, et al. ASAP7: A 7-nm FinFET predictive process design kit. Microelectron J 2016;53:105–15.