Real Time TCAD Calibration via Transfer Learning

Yunji Choi, Sanghoon Myung*, Kyeyeop Kim, Gijae Kang,
Beomwon Jung, Yongwoo Jeon, Kyungmi Yeom, Songyi Han, Jaehoon Jeong, and Dae Sin Kim
Computational Science and Engineering Team, Innovation Center, Samsung Electronics
Email: *shoon.myung@samsung.com

2024 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD) | 979-8-3315-1635-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/SISPAD62626.2024.10733328

***Abstract*—This study introduces a novel calibration framework of deep learning model, called Real Time TCAD (RTT) calibration. The proposed method comprises two steps: the pre-training step and the transfer learning step. In the pre-training step, a deep learning model is trained with a large amount of data generated from the uncalibrated TCAD simulation to learn the underlying semiconductor physics. During the transfer learning step, the pre-trained model is fine-tuned with actual measurements to minimize the discrepancy between the model predictions and the actual measurements. To prevent overfitting to the experimental data, we introduce a novel loss function that enables the model to learn the modified physics while preserving the underlying TCAD knowledge. The proposed method significantly reduces both calibration and data generation time while maintaining high model accuracy. Furthermore, our method tackles the critical problem of missing values in the data, which is commonly encountered in practice. Experimental results on a 28-nm logic process demonstrate the effectiveness of the proposed method.**

***Index Terms*—Real Time TCAD, Neural Network, Calibration, Transfer Learning**

I. Introduction

The miniaturization of transistors has led to a significant increase in development costs. To mitigate these expenses, many companies have employed Technology Computer-Aided Design (TCAD) simulations to predict the electrical characteristics of transistors. However, with the emergence of various multi-dimensional transistors, the utilization of TCAD occasionally incurs significant time costs, ranging from hours to days. Therefore, Artificial Intelligence (AI) models that offer considerable speed advantages over TCAD have garnered great attention, especially when provided with sufficient data [1]–[7]. Several studies have attempted to replace the TCAD simulators themselves [1]–[4]. This study introduces deep learning to provide an initial solution that assists TCAD in achieving fast convergence [5]. Additionally, some studies have also aimed to substitute the compact model for SPICE simulation [8]–[10].

Nevertheless, these data-driven AI models are vulnerable to distributional shifts caused by process changes or equipment variations [11]. In such cases, it is necessary to modify the existing TCAD models, commonly referred to as TCAD calibration (see Fig. 1 (a)). Afterward, new training data is generated from calibrated TCAD. Due to the considerable time consumption required for these tasks (Fig. 1 (a)), many companies have not yet aggressively introduced AI. To solve this problem, this study proposes a novel calibration framework leveraging a deep learning model, referred to as Real Time TCAD (RTT) [1]–[4] calibration, as shown in Fig. 1 (b). It utilizes transfer learning to reduce the gap between measurements and the RTT model within a shorter calibration time. The rest of this paper is organized as follows. Section II details our method for enabling real time calibration. Section III demonstrates the effectiveness of our calibration method in terms of both accuracy and time efficiency. Section IV provides a summary of this paper along with a brief discussion of future directions.

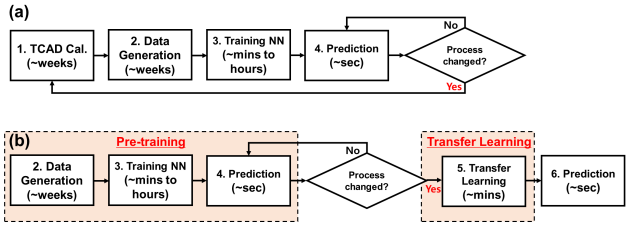


Fig. 1. Illustration of the process flow from the data generation to the deployment of an AI model. (a) The conventional method must restart the flow from the beginning when the process changes. (b) The proposed flow can alleviate the ineffective of iterative calibration and data generation procedures via transfer learning that we propose.

II. Methodologies

We start by reviewing the RTT models [1] that we employed to estimate the electrical characteristics. There are two primary types of the experimental data: major electrical characteristics such as threshold voltage (V_T) and current-*versus*-voltage (I-V) curve. The former represents the simplest form of data that is consistently obtainable. In contrast, the latter refers to the I-V curves, which can provide a wealth of information but are often unavailable due to the time-consuming nature of measurement compared to the former. To handle these distinct data types, we employed two separate model architectures.

Model architecture. The first model is an artificial neural network consisting of a few fully-connected layers designed to predict major electrical characteristics (Fig. 2 (a)). The second model employs a combination of fully-connected layers to map the input space to a high-dimensional space and Long Short-Term Memory (LSTM) [12] to capture sequential information, as illustrated in Fig. 2 (b). This approach takes into account not only the current voltage step but also the preceding ones. Therefore, this model enables more efficient training with

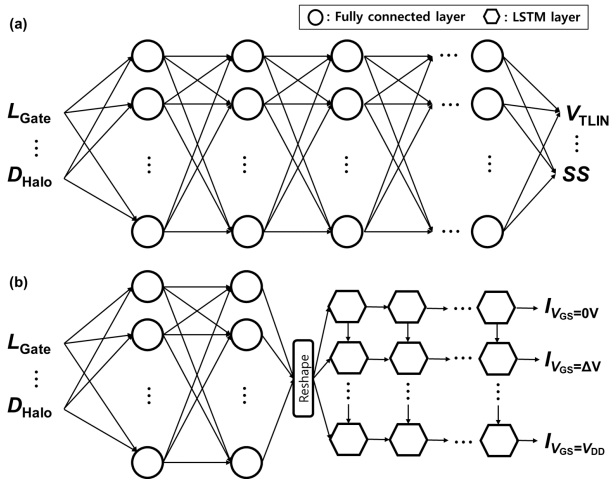


Fig. 2. Illustration of model architectures for the prediction of (a) electrical characteristics and (b) I-V curves. Model (a) consists of several fully-connected layers that composes of identical units and uses GELU activation function [13]. Model (b) combines two distinct layers. The first layer is the fully-connected layers to elevate the input vector to a high-dimensional vector. The other is LSTM layers to consider the latent vector as sequential I-V curve after reshaping the output vector.

sequential voltage-current data by considering the outcomes of previous voltage steps [1].

With these models, we describe how to apply the transfer learning in detail. We propose a novel calibration framework consisting of two steps, as illustrated in Fig. 1 (b): i) pre-training and ii) transfer learning.

Pre-training. The goal of this step is to enable the deep learning model to learn the underlying semiconductor physics from the data generated by the TCAD simulator, with minimal effort. To achieve this, we utilize an uncalibrated TCAD simulator to generate a substantial dataset for training purposes. Subsequently, we train the aforementioned two models with these uncalibrated TCAD data. We denote these pre-trained models as $(f(x; \mathbf{w}_0))$. It is important to note that the TCAD simulator and its data are NO longer required after this step, thereby avoiding unnecessary procedures in the conventional flow, as shown in Fig. 1 (a).

Transfer learning. The objective of this step is to minimize the discrepancy between the the pre-trained models $(f(x; \mathbf{w}_0))$ and actual measurements. This step corresponds to TCAD calibration, and we hence dub this step as "RTT calibration". Different from the previous step (pre-training), we initialize the weights ($\mathbf{w}$) with those of the pre-trained models ($\mathbf{w}_0$), and then train the models (i.e., the calibrated models, $f(x; \mathbf{w}_*)$) with the measurements as follows:

$$\mathbf{w}(i+1) = \mathbf{w}(i) - \gamma \nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}), \quad \mathbf{w}(0) = \mathbf{w}_0,$$

where γ denotes a learning rate and $\mathcal{L}$ represents a loss function. In practice, we often consider $\mathcal{L}$ as *mean-squared-errors*.

In this step, two significant challenges exist. The first challenge is that some of the output characteristics are not measured. This implies that we may need to train the model with insufficient measurement data, or in some cases, we may lack any measurement data at all for model training. The second challenge is the number of measurement data is typically scarce. As a result, the model may over-fit to the sparse dataset.

Loss function. If any of the output features that the model learned are not measured, it is not feasible to train the model using a conventional loss function such as *mean-squared-error*. However, assuming that the model can learn the relationships between outputs [1], it can infer the missing outputs from the related ones. To this end, we propose a loss function that utilizes all existing outputs without discarding samples containing missing values, formulated as follows:

$$\mathcal{L}_{mask} = (y \odot \mathbb{1}(y,m) - f(x;\mathbf{w}) \odot \mathbb{1}(y,m))^2,$$
$$s.t. \quad \mathbb{1}(a,b) = \begin{cases} 1, & if \quad a \neq b \\ 0, & otherwise, \end{cases}$$

where y denotes the measurements, $\odot$ represents element-wise multiplication, and m indicates the masking value used to assign a predetermined value that represents the missing data. In this loss function, $\mathbb{1}$ is used to determine whether a missing value exists or not. If there is no masking value, the loss function aims to minimize the difference between the predicted value and the true one. On the other hand, if the masking value exists, the predicted output is excluded from the loss calculation. This approach allows the model to infer missing values though training by leveraging other training instances or related outputs.

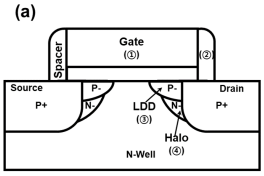
Additionally, it is imperative to prevent over-fitting of the model resulting from the limited data availability [14]. Specifically, an over-fitted model often yields inaccurate results that deviate from the fundamental physics. Therefore, we incorporate regularization into the loss function to constrain the updated model weights ($\mathbf{w}$) from deviating significantly [15] from the pre-trained weights ($\mathbf{w}_0$, this means underlying physics that the model learned), as follows:

$$\mathcal{L} = \mathcal{L}_{mask} + \lambda(\mathbf{w} - \mathbf{w}_0)^2,$$

where λ represents the regularization strength. As λ is a hyper-parameter for training, we set it to 0.01 in this study. A higher λ encourages the model to learn from the measurements while strictly adhering to the underlying physics. Conversely, a lower λ allows the model to prioritize learning from the measurements with less strict adherence to the underlying physics.

III. Experimental Results

In this section, we commence the description of the 28-nm Logic device. Fig. 3 (a) depicts the input variables for the RTT models that we control to optimize the performance of the transistors. Fig. 3 (b) indicates two different output variables: i) major electrical characteristics (V_T, I_{ON}, SS, and so on) and ii) I-V curves. We applied our method, the RTT calibration, to two RTT models for these two types of output data.



(b)

	Category	Variables
Input	① Gate	Gate length
	② Spacer	Spacer thickness
	③ LDD Implantation	Energy, dose, tilt
	④ Halo Implantation	Dose
Output_1	Electrical Characteristics	V_{TLIN}, V_{TSAT}, I_{DLIN}, I_{DSAT}, I_{SOFF}, I_{BOFF}, SS
Output_2	Current-Voltage(*I-V*) Curve	I_{DS}-V_{GS}

Abbreviation

V_{TLIN}: Gate voltage when $I_{DS}=I_{TH}$, $V_{DS}=0.05V$
I_{DLIN}: Drain current when $V_{DS}=0.05V$, $V_{GS}=V_{DD}$
I_{SOFF}: Source current when $V_{DS}=V_{DD}$, $V_{GS}=0V$
SS: Inverse subthreshold slope, $(\frac{d\log(I_{DS})}{dV_{GS}})^{-1}$ when $V_{DS}=V_{DD}$, $V_{GS}=0$
V_{TSAT}: Gate voltage when $I_{DS}=I_{TH}$, $V_{DS}=V_{DD}$
I_{DSAT}: Drain current when V_{DS}=VDD, $V_{GS}=V_{DD}$
I_{BOFF}: Body current when $V_{DS}=V_{DD}$, $V_{GS}=0V$

Fig. 3. (a) Illustration and (b) table on input and output variables of 28-nm p-type MOSFET. Inputs are related to process recipes and initial structures. Outputs are electrical characteristics of the device.

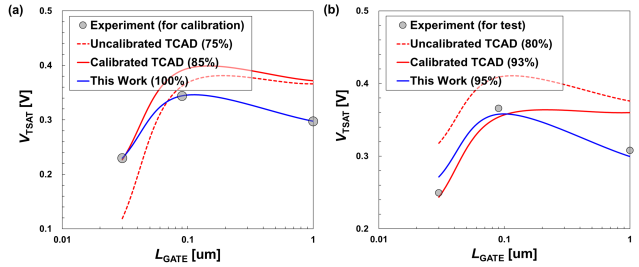


Fig. 4. V_T roll-off curve on (a) calibration wafers and (b) test wafers. Numbers in parentheses indicate the accuracy.

A. RTT Calibration for Electrical Characteristics

In this calibration, we used twenty input variables and seven output variables, respectively. A total of 1,000 data points were generated from the uncalibrated TCAD simulator and used to train the RTT model, as shown in Fig. 2 (a). As a consequence of the first step, the accuracy of the pre-trained RTT model surpassed 99%. Then, we conducted experiments with various input conditions, including implantation doses across different gate lengths on the wafers.

Calibration results. As indicated in Fig. 4, the results of the pre-trained RTT model (uncalibrated TCAD results) are inconsistent with experiments. In this scenario, we always perform TCAD calibration tasks. Hence, we proceeded with the calibration of two models: i) TCAD simulator (manual calibration by a TCAD expert) and ii) RTT model (conducting transfer learning, as described in Sec. II). In the case of TCAD calibration (Fig. 4, solid red line), the TCAD expert calibrated the TCAD simulator solely at the design length (minimal length) due to time constraints. As a result, the TCAD calibration results aligned well with the measurement for the design length, whereas those for other lengths exhibited lower accuracy as anticipated, as depicted in Fig. 4 (a). In contrast to TCAD calibration, the results of transfer learning (Fig. 4, solid blue line) are consistent with the measurements across the whole lengths because these data were used in training. We can use all the data since our framework exhibits superior time efficiency. Fig. 5 demonstrates that our approach can reduce calibration and data generation time by 99.9% in cases where the process has been changed.

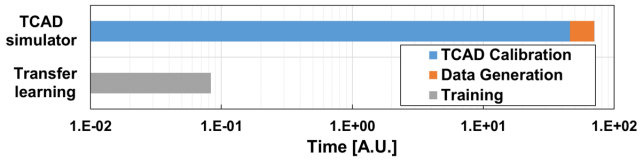


Fig. 5. Comparison of time consumption between TCAD calibration and transfer learning.

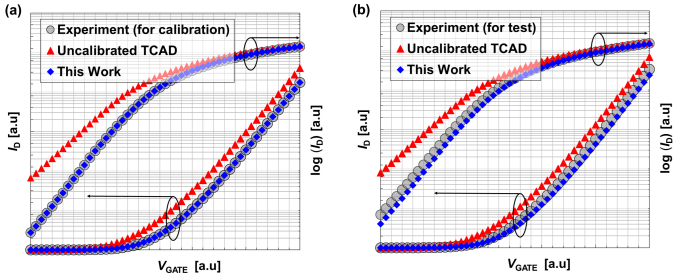


Fig. 6. I_{DSAT}-V_{GS} data of two different wafers for (a) calibration and (b) test. For comparison, we employed 50 regular intervals of voltage (ΔV_{GS}) on the x-axis. The left y-axis represents the linear scale of I_{DSAT}, while the right y-axis represents the logarithmic scale.

Test results. Next, we conducted experiments on the next wafers to evaluate two calibrated models: the TCAD and RTT models. As expected, both models exhibited good agreement with the experiments at the design length (Fig. 4 (b)). However, across the overall lengths, the results from the calibrated RTT model exhibited a closer alignment with the experimental data compared to those from the calibrated TCAD model.

B. RTT Calibration for Current-Voltage Curve

The input variables are identical with electrical characteristics, but the output variables are the drain current, composed of fifty voltage steps. In this experiment, we trained the RTT model, shown in Fig. 2 (b), with 1,000 uncalibrated TCAD data points. The accuracy of the pre-trained RTT model exceeded 99% when compared to the TCAD simulator. The experimental data for calibration and testing are same input condition of previous experiments. The predictions of the pre-trained RTT model deviated from the experimental data (Fig. 6 (a), red triangles).

Calibration results. To reduce this gap, we calibrated the pre-trained RTT model (Fig. 2 (b)) with experimental data through transfer learning. Consequently, the predictions of the calibrated RTT model closely align with the experimental data, as shown in Fig. 6 (a). In this scenario, we did not calibrate the TCAD simulator to the measurement due to time constraints.

Test results. We conducted experiments on subsequent wafers, identical to the assessment of the RTT model for electrical characteristics. As depicted in Fig. 6 (b), our result nearly overlaps with the experimental I-V curve on both linear and logarithmic scales. This indeed suggests that our calibration methodology using the RTT model effectively functions within a limited time frame.

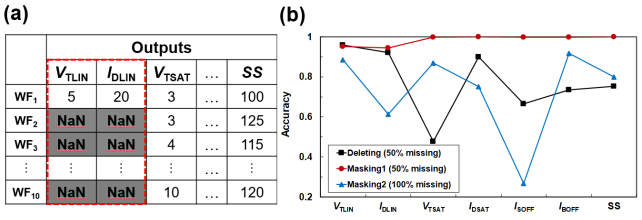


Fig. 7. (a) Description when some of output features are skipped. (b) An accuracy plot of all outputs according to the ration of missing values. $\mathcal{L}_{mask}$ enables the RTT model to fully utilize the data although 50% of data are missing values.

C. Missing Value Problems

It is frequently observed that some of the output characteristics are not measured. For example, we often omit measuring the electrical characteristics at low drain voltage to reduce time expenses, as they are considered less important compared to the high voltage characteristics. In such cases, the conventional approach is to remove samples that contain missing values. However, this approach leads to a substantial decrease in the amount of training data, which often deteriorates the performance of the model. As a result, the model may predict inaccurate outcomes that deviate from the underlying physics. In contrast to this conventional approach, our method utilizes a novel loss function ($\mathcal{L}_{mask}$) that enables the model to be trained using all samples, including those that contain missing values. To assess our method, we assume two scenarios: i) half and ii) all of electrical characteristics in case that drain voltage is low, are not available (Fig. 7 (a)). Fig. 7 (b) shows that our method performs well even when half of the data is missing. Compared to the conventional method (sample removal), our method improves accuracy by 52% points. Moreover, even when all data contain missing values, our method still achieves high accuracy, whereas the conventional method fails to do so.

IV. Conclusion

This article proposes a simple yet effective method to reduce expenses for calibration and data generation when process changed. By eliminating inefficient procedures, it can reduce the time expenses by almost 99.9%, while maintaining high model accuracy. In particular, our method can be applicable to tasks where some features of the measurements are unavailable. This feature can improve data efficiency by up to 52%. Although this work yielded promising results, we could not conduct extensive experiments due to the limited number of wafers. As a result, it cannot be guaranteed that the calibrated RTT model fully covers all areas of interest. Therefore, future work will focus on assessing the coverage of the model and exploring methods to expand its applicability.

References

[1] S. Myung, J. Kim, Y. Jeon, W. Jang, I. Huh, J. Kim, S. Han, K.-h. Baek, J. Ryu, Y.-S. Kim, J. Doh, J.-h. Kim, C. Jeong, and D. S. Kim, "Real-time tcad: A new paradigm for tcad in the artificial intelligence era," in *2020 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*. IEEE, 2020, pp. 347–350, doi: https://doi.org/10.23919/SISPAD49475.2020.9241622.

[2] S. Myung, W. Jang, S. Jin, J. M. Choe, C. Jeong, and D. S. Kim, "Restructuring tcad system: Teaching traditional tcad new tricks," in *2021 IEEE International Electron Devices Meeting (IEDM)*, 2021, pp. 18.2.1–18.2.4, doi: https://doi.org/10.1109/IEDM19574.2021.9720616.

[3] S. Myung, B. Choi, W. Jang, J. Kim, I. Huh, J. M. Choe, Y.-G. Kim, and D. S. Kim, "Comprehensive studies on deep learning applicable to tcad," *Japanese Journal of Applied Physics*, vol. 62, no. SC, p. SC0808, 2023, doi: https://doi.org/10.35848/1347-4065/acbaa6.

[4] W. Jang, S. Myung, J. M. Choe, Y.-G. Kim, and D. S. Kim, "Tcad device simulation with graph neural network," *IEEE Electron Device Letters*, vol. 44, no. 8, pp. 1368–1371, 2023, doi: https://doi.org/10.1109/LED.2023.3290930.

[5] S.-C. Han, J. Choi, and S.-M. Hong, "Acceleration of semiconductor device simulation with approximate solutions predicted by trained neural networks," *IEEE Transactions on Electron Devices*, vol. 68, no. 11, pp. 5483–5489, 2021, doi: https://doi.org/10.1109/TED.2021.3075192.

[6] C. Jeong, S. Myung, I. Huh, B. Choi, J. Kim, H. Jang, H. Lee, D. Park, K. Lee, W. Jang *et al.*, "Bridging tcad and ai: Its application to semiconductor design," *IEEE Transactions on Electron Devices*, vol. 68, no. 11, pp. 5364–5371, 2021, doi: https://doi.org/10.1109/TED.2021.3093844.

[7] H. Dhillon, K. Mehta, M. Xiao, B. Wang, Y. Zhang, and H. Y. Wong, "Tcad-augmented machine learning with and without domain expertise," *IEEE Transactions on Electron Devices*, vol. 68, no. 11, pp. 5498–5503, 2021, doi: https://doi.org/10.1109/TED.2021.3073378.

[8] J. Wang, Y.-H. Kim, J. Ryu, C. Jeong, W. Choi, and D. Kim, "Artificial neural network-based compact modeling methodology for advanced transistors," *IEEE Transactions on Electron Devices*, vol. 68, no. 3, pp. 1318–1325, 2021, doi: https://doi.org/10.1109/TED.2020.3048918.

[9] Y. Kim, S. Myung, J. Ryu, C. Jeong, and D. S. Kim, "Physics-augmented neural compact model for emerging device technologies," in *2020 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*. IEEE, 2020, pp. 257–260, doi: https://doi.org/10.23919/SISPAD49475.2020.9241638.

[10] S. Myung, D. Shin, K. Kim, Y. Choi, G. Kang, S. Han, J. Jeong, and D. S. Kim, "A new industry standard compact model integrating tcad into spice," in *2024 Symposium on VLSI Technology and Circuits*. IEEE, 2024.

[11] S. Myung, H. Jang, B. Choi, J. Ryu, H. Kim, S. W. Park, C. Jeong, and D. S. Kim, "A novel approach for semiconductor etching process with inductive biases," *arXiv preprint*, 2021, doi: https://doi.org/10.48550/arXiv.2104.02468.

[12] A. Graves and A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012, doi: https://doi.org/10.1162/neco.1997.9.8.1735.

[13] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016, doi: https://doi.org/10.48550/arXiv.1606.08415.

[14] S. Myung, I. Huh, W. Jang, J. M. Choe, J. Ryu, D. Kim, K.-E. Kim, and C. Jeong, "Pac-net: A model pruning approach to inductive transfer learning," in *International Conference on Machine Learning*. PMLR, 2022, pp. 16 240–16 252. [Online]. Available: https://proceedings.mlr.press/v162/myung22a.html

[15] L. Xuhong, Y. Grandvalet, and F. Davoine, "Explicit inductive bias for transfer learning with convolutional networks," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2825–2834. [Online]. Available: https://proceedings.mlr.press/v80/li18a.html