Received 3 July 2023; revised 6 September 2023 and 23 October 2023; accepted 24 October 2023. Date of publication 30 October 2023; date of current version 20 November 2023. The review of this paper was arranged by Associate Editor K. Endo.

Digital Object Identifier 10.1109/OJNANO.2023.3328425

A Comprehensive Technique Based on Machine Learning for Device and Circuit Modeling of Gate-All-Around Nanosheet Transistors

RAJAT BUTOLA [1], YIMING LI [1,2] (Senior Member, IEEE), AND SEKHAR REDDY KOLA [1] (Graduate Student Member, IEEE)

[1] Parallel and Scientific Computing Laboratory, Electrical Engineering and Computer Science International Graduate Program, National Yang Ming Chiao Tung University, Hsinchu 300093, Taiwan
[2] Institute of Communications Engineering, the Institute of Biomedical Engineering, the Department of Electronics and Electrical Engineering, the Institute of Pioneer Semiconductor Innovation, and the Institute of Artificial Intelligence Innovation, National Yang Ming Chiao Tung University, Hsinchu 300093, Taiwan

CORRESPONDING AUTHOR: YIMING LI. (e-mail: ymli@nycu.edu.tw)

This work was supported in part by the National Science and Technology Council, Taiwan under Grant NSTC 112-2221-E-A49-171 and in part by the 2022 Qualcomm Taiwan Research Program (NYCU) under Grant NAT-487835 SOW.

ABSTRACT Machine learning (ML) is poised to play an important part in advancing the predicting capability in semiconductor device compact modeling domain. One major advantage of ML-based compact modeling is its ability to capture complex relationships and patterns in large datasets. Therefore, in this paper a novel design scheme based on dynamically adaptive neural network (DANN) is proposed to develop fast and accurate compact model (CM). This framework constitutes a powerful yet computationally efficient methodology and exhibits emergent dynamic behaviors. This paper demonstrates that the compact model based on ML can be designed to replicate the performance of conventional compact model for nanodevices. For this work, gate-all-around (GAA) nanosheet (NS) device characteristics are comprehensively analyzed for process variability sources using the proposed model. The device geometry parameters such as channel length, nanosheet width and nanosheet thickness are fed as input features to the DANN model. The adaptive neural network learns dynamically by updating weights of the model in accordance with the input features and achieves accurate neural weight convergence. The proposed model predicted the electrical characteristics of NS devices with less than 1% error rate. The model is also implemented and validated for the simulations of digital circuit designs such as inverter, and logic gates.

INDEX TERMS Adaptive neural network, circuit simulation, dynamic weights, gate-all-around, MOSFET, nanodevice, nanosheet, process variability.

I. INTRODUCTION

Over the past few years a number of fields have benefited from developments in machine learning (ML) beyond computer science [1], [2]. In recent times, ML techniques have emerged as valuable tools for enhancing semiconductor device modeling for electronic design automation [3], [4], [5], [6]. Previous studies have demonstrated that ML has the capacity to effectively address a variety of semiconductor challenges [7], [8], [9], [10], [11], particularly in the context of device modeling tasks [12], [13], [14], [15]. This emphasizes the potential of ML to offer innovative solutions within the semiconductor domain [16], [17]. As we know, scaling is a fundamental principle in the semiconductor industry [18]. It has been a driving force behind the rapid increase in computing power, energy efficiency, and the number of transistors incorporated into an integrated circuit (IC) over the past few years. Gate-all-around (GAA) Silicon (Si) nanosheet (NS) is such a nanoscale device that is used to produce chips under 5 nm technology node [19], [20], [21]. Thus, NS is one of the most advanced transistors for device logic applications for future technology nodes [22], [23]. However, due to scaling the devices have become more prone to process variation (PV) and therefore they need to be considered separately in IC designing [24], [25], [26], [27], [28], [29], [30], [31]. Recently many ML-based solutions are utilized successfully to analyze the effects of PV in semiconductor devices [32], [33]. Artificial Neural Network (ANN), a subset of ML, is also applied to address PV in semiconductor devices by modeling and predicting the impact of variations on device characteristics. The process of applying ANN to address PV in semiconductor devices starts from collecting the data and preparing it for ANN model. The quality and relevance of the data is very crucial for the accuracy of the ANN model. The dataset must consist of a range of PV to ensure that it is representative and includes a variety of input patterns. The model is trained with collected dataset and learns the complex relationships between input parameters and output metrices. ANNs have the ability to capture the complex nonlinear relationships between input process parameters and device performance. They offer several advantages for process variation modeling. The ANN provides rapid predictions that allow for quick decision-making during manufacturing. It can also reduce the requirement for expensive manufacturing testing by predicting potential issues caused by process variations. In recent past many attempts have been made to predict the effects of PVs on the characteristics of emerging nanodevices. In [34], an ML approach based on ANN is proposed for ultra-scaled GAA devices that assess the degree of process variability accurately. In another work, authors proposed an ANN technique to demonstrate its high accuracy compared to the tcad simulations for dimension and work function variations [35]. Similarly, the effect of multiple process variability source effects, global variation (GV) and local variation (LV), are also considered and analyzed using ML-assisted device modeling method [36].

Furthermore, due to continuation of device scaling, nanodevices such as NS have more complex underlying physics [37]. As a result, the existing compact models (CM) are not befitting for NS devices. Additionally, developing a new physical CM is a very complicated task that takes a long time and requires special expertise of the device. Compact modeling is basically a set of analytical equations based on the physics of the specific transistor and can reproduce complex nonlinear transistor characteristics. These equations are utilized to represent the electrical behavior of devices. However, as reported in [38], it faces three key challenges 1) foundry secrecy: to keep its technology details confidential, 2) innovation requires new CMs which takes long time to develop, and 3) early evaluation of technology.

Therefore, in this paper, we move away from the conventional CMs and investigate the use of ML, especially, ANNs based CMs for emerging nanodevices [39], [40], [41], [42], [43], [44], [45], [46]. In recent past many attempts have been made to build fast and accurate CMs with short turnaround time (TAT: refers to the amount of time taken to complete a process) by using ANNs. They are critical to bridge the early-stage design optimization with technology for novel emerging semiconductor devices. An effective CM must have expertise to capture complex non-linear relationships in a large multivariate dataset. Therefore, the ANN-based CMs are the best approach as they can efficiently capture both linear as well as non-linear multivariate relationships. The ANN models have already been proposed as a means to compact modeling. These models can be divided into two category: physics-based models and pure ML models. Gao et al. [40] proposed physics-informed graph neural network (pigNN) for rapid and automated CM development. They applied this model to accurately capture the smooth transition between the exponential and quasi-linear response regions of non-ideal PN diode. In another work, [41] a comprehensive physics-based CM using grove-frohman (GF) model and ANN is presented for emerging GAA MOSFETs. The model is implemented accurately on NS and CFET devices without suffering from divergent issues in circuit simulation. Similarly, in [42] authors demonstrated partially-physics-based MOSFET current-voltage CM using deep learning model. As presented in these works, the physics based CMs essentially built upon the fundamental principles of semiconductor physics, which govern the behavior of devices. These CMs directly incorporate physical equations and parameters that describe the device operation. However, these physics-based models are sometimes not able to capture complex device behaviors accurately, especially if the physics is not fully understood.

On the contrary, the pure ANN based ML models are highly flexible and are appropriately able to capture complex nonlinear relationships without requiring in-depth domain knowledge. In [47], the ANN CM is proposed that improved the ANN training TAT and spice simulation TAT by exploring the model capability for model retargeting and variability modeling. The ANN-based CM presented in [48] is accurately used in the initial sizing of analog circuit components without simulation. Similarly, in [49] authors proposed a fast ANN-based CM technique to imitate parameter extraction (MPE) flow to replace the existing complicated CM implementation.

However, these existing research face two main challenges: 1) lack to exhibit emergent dynamic behaviors, and 2) a large amount of data requirement for training. All the current CM are based on static ANN models which lack dynamic behavior and therefore have fixed computational graphs and parameters (such as bias and weights) at the inference stage. Moreover, the parameter space being searched in the space of' static networks is much smaller than for dynamic networks. Therefore, they cannot adapt their structures or parameters to different inputs, leading to notable disadvantages in terms of computational efficiency, accuracy, and adaptiveness. Moreover, fixed parameters may also limit their interpretability, efficiency, and representation power. Adaptive weight updating, and real dynamics are not involved in the present models.

Furthermore, the process of analyzing complex physics and electrical characteristics of emerging semiconductor devices usually requires a large amount of dataset. Generally, tcad is used to construct the training dataset for ANNs. It is accurate but not time efficient due to this, the time and cost of data acquisition is often very high and prohibitive. The other simulator that can be used to construct device data is spice simulator. They are fast but not as accurate as tcad. Therefore, following the previous study, in this paper we present a novel transistor CM based on DANN that determines the weights dynamically from predicting accuracies of the trained network with training dataset. It also used hybrid training dataset constructed from tcad and spice simulator to comprehensively reduce the data generation time and augment the accuracy.

The key contributions of this research are as follows.

1) A ML method based on adaptive weight strategy is proposed where the model's weights are dynamically adjusted in accordance with the input features to improve the training convergence and to reduce the generalization error.
2) Data acquisition time is reduced by generating half of the training samples from spice simulation which is a much faster technique as compared to tcad.
3) Validating the proposed CM on nanosheet based circuits for advanced sub-3-nm technology nodes.

The rest of this article is organized as follows. Section II presents the description of ML techniques for semiconductor devices including proposed model. Section III consists of the trade- off between TCAD and SPICE simulation. Section IV discusses device overview and variability modeling. Section V discusses the results obtained from the experiments. Finally, Section VI concludes this paper.

II. MACHINE LEARNING TECHNIQUES FOR SEMICONDUCTOR DEVICES

A. MACHINE LEARNING FOR SEMICONDUCTOR

Machine learning (ML) is a subfield of artificial intelligence (AI) that focuses on the development of algorithms and models that are capable of learning from data and making intelligent predictions. The concept of ML was proposed in 1959 and since then several ML concepts and algorithms have been developed. Specially, in recent years it has witnessed tremendous growth and achievements that has influenced various other fields apart from computer science. In semiconductor fabrication, ML has gained tremendous attention and revolutionized the semiconductor industry by analyzing the large amount of simulation data, recognize the useful patterns and extract valuable insights. Semiconductor manufacturing using ML is known as smart manufacturing. It has been utilized in semiconductor applications such as failure analysis and yield enhancement, device characterization and modeling, design optimization, etc.

In device modeling, different kinds of simulation and experimental data of various past and emerging MOS transistors have been accumulated, and some relevant characteristics analysis have been assessed using ML. Therefore, as an emerging technological tool, ML can play an important role in the design, manufacturing, and operation of emerging nanodevices. However, it also faces some challenges: 1) Data acquisition and pre-processing; Device data are generally complex and nonlinear, so effective data acquisition and pre-processing are required. The accuracy and quality of the data need to be considered. 2) Scalability and real-time operation; Scalability and real-time operation are important challenges for ML research, and more efficient and scalable algorithms and models need to be discovered for device modeling applications. The analogy between simulation and ML approach is illustrated in Fig. 1.

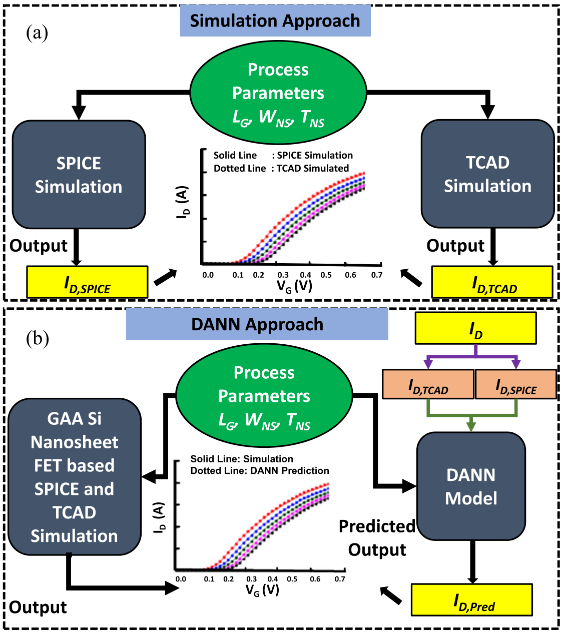

FIGURE 1. Analogy between simulation and machine learning approach. (a) Tcad and spice datasets are generated using different process parameters to produce the I_D-V_G characteristics. (b) Proposed DANN approach is utilized to predict I_D-V_D characteristics using the hspice and tcad datasets.

B. STATIC NETWORKS

The ANN algorithms are extensively used to model nonlinear statistical modeling [50]. They are a kind of information processing system which emulates biological neural network [51]. Biological neurons are simulated in these networks by obtaining information from the outside environment or other artificial neurons, making very simple operations, and producing results to the outside environment [52]. The structure of ANN consists of an input layer, one or more hidden layers (specified when developing the structure of a neural network according to the problem to be solved) and an output layer. These layers consist of neurons and each neuron is connected to other neurons of the next layer by connection called weight. ANN provides various advantages, such as ability to inherently detect complex nonlinear relationships between dependent and independent variables, detection ability of all possible interactions between predictor variables, require less formal statistical training, and the possibility of multiple training algorithms. One of the characteristics of ANN algorithms, especially for semiconductor device modeling, is that it does not require physical descriptions of the phenomena involved in the process. They provide powerful analysis of device characteristics and offer complex processing of large input/output information arrays.

So, these feedforward neural networks have been the most widely used form of ANN in representing the nonlinear dynamic processes of semiconductor devices. However, in these networks the evolved solutions are usually static in nature in the sense that the model parameters remain fixed throughout the inference process. Whereas, in various important control problems, the input features may change gradually or drastically and hence to maintain sufficient performance the model needs to adapt. Adaptation is also important to make the ANN models more general. For relatively simple dataset, ANN is more popular to be explored as a predictive model. The reduction of computational complexity and memory requirement are the primary goals to utilize these static models. Due to reduction of computational cost and memory consumption, the static ANN is particularly useful for applications that can work in limited resources. Since the weights and biases of static ANN are fixed, therefore, it is advantageous to explore it in the recursive dataset that can predict the same behavior multiple times. Besides these advantages, static models lack the ability of adaptability of new data sets and their performance degrades while dealing with unseen information. Therefore, one way to make the ANN models to achieve adaptive solutions is to evolve neural networks with dynamic weights that lead to robust adaptability.

C. PROPOSED DYNAMIC ADAPTIVE NEURAL NETWORKS

The goal of dynamic adaptive neural network (DANN) is different from static networks that simply utilize learning to device modeling applications. DANN learning mechanisms are optimized by dynamically updating the weights to adapt to new or changed problems that can arise in the domain. They are very effective in tasks that depend on an aspect of the input features that varies randomly from one trial to the next. The dynamic weighting approach [53], [54], [55] allows ANN to adapt to novel situations where fixed-weight networks could not perform well. The parameter space that is being searched in adaptive networks space is much larger than for static ANNs. Moreover, the network's weights must be evolved for each random input features. The knowledge about the error is collected that the neural networks make with the training dataset in learning phase and this knowledge is carry forward to take benefit during the prediction of new instances. We aim to assign higher ensemble weight for the more reliable and efficient neural networks. The relationship is created between input features and predicting errors from training dataset, and the weights of adaptive NN are determined dynamically based on the relationship of error and input features.

In this proposed work, a dynamical learning framework is applied for the assessment of process variation effects in emerging nanodevice GAA NSFET which integrates with a hybrid dataset of tcad and spice simulation and takes both accuracy and time factor into account owing to two main reasons: 1) tcad provides reliable device physics, but data generation process is time-consuming, in contrast 2) spice offers rapid simulations but a slight compromise in accuracy. It is a typical approach where the weights of ANN are adjusted based on the input features. The weights are dynamically determined from the prediction accuracies of the trained ANNs with the training dataset. The modeling framework of the proposed model is shown in Fig. 2. The framework consists of three ANNs. The architecture of each ANN used consists of a multi-layer feed forward NN with one hidden layer constituting of different set of neurons.

The first ANN model (ANN_1) incorporates 15 neurons in hidden layer and used hyperbolic tangent (tanh) as activation function in the neurons whereas the activation function used in output layer is linear transfer function. The other two networks ANN^{w1} and ANN^{w2} are similar in structure with having 5 neurons in hidden layer and used relu as activation function in hidden layer and linear transfer function in output layer. These three ANNs perform different tasks. First of all, the spice samples are used to train the first ANN_1 model having one hidden layer and 15 neurons. The predicted response of ANN_1 is labeled as $\text{I}_{\text{D,ANN,}}$ and the weights of the network are also evaluated. These weights have less importance since the accuracy of the spice data is insufficient. Therefore, in the next step, the tcad data is assimilated to make sure overall accuracy. Thus, in the second step, based on three different current datasets (i.e., $\text{I}_{\text{D,SPICE}}$, $\text{I}_{\text{D,TCAD}}$ and predicted $\text{I}_{\text{D,ANN}}$) two sets of weights W_{NN} and W_{TCAD} are generated as below:

$$W_{NN} = \begin{cases} \frac{I_{D,ANN}-I_{D,TCAD}}{I_{D,SPICE}-I_{D,TCAD}} & I_{D,TCAD} > I_{D,SPICE} > I_{D,ANN} \\ & I_{D,TCAD} < I_{D,SPICE} < I_{D,ANN} \\ 0 & I_{D,SPICE} > I_{D,TCAD} > I_{D,ANN} \\ & I_{D,SPICE} < I_{D,TCAD} < I_{D,ANN} \\ 1 & I_{D,SPICE} > I_{D,ANN} > I_{D,TCAD} \\ & I_{D,SPICE} < I_{D,ANN} < I_{D,TCAD} \end{cases} \tag{1}$$

$$W_{TCAD} = \begin{cases} \frac{I_{D,ANN}-I_{D,TCAD}}{I_{D,SPICE}-I_{D,TCAD}} & I_{D,TCAD} > I_{D,SPICE} > I_{D,ANN} \\ & I_{D,TCAD} < I_{D,SPICE} < I_{D,ANN} \\ 1 & I_{D,SPICE} > I_{D,TCAD} > I_{D,ANN} \\ & I_{D,SPICE} < I_{D,TCAD} < I_{D,ANN} \\ 0 & I_{D,SPICE} > I_{D,ANN} > I_{D,TCAD} \\ & I_{D,SPICE} < I_{D,ANN} < I_{D,TCAD} \end{cases} \tag{2}$$

The intention behind defining these weights is to enhance the accuracy of the model by incorporating information from tcad simulations. When the $\text{I}_{\text{D,SPICE}}$ values fall outside the range defined by $\text{I}_{\text{D,TCAD}}$ and $\text{I}_{\text{D,ANN}}$, they are transformed to binary values zero or one for W_{NN} and one or zero for W_{TCAD}, according to the conditions as represented in equations. On the contrary, when $\text{I}_{\text{D,SPICE}}$ falls within the range between $\text{I}_{\text{D,TCAD}}$ and $\text{I}_{\text{D,ANN}}$, then the values for W_{NN} and W_{TCAD} are determined using the ratios as demonstrated in equations. These ratios are calculated based on the $\text{I}_{\text{D,SPICE}}$, $\text{I}_{\text{D,TCAD}}$, and $\text{I}_{\text{D,ANN}}$ currents, ensuring that the model assigns appropriate values to these weights. In this context, this approach optimizes the weights to make the $\text{I}_{\text{D,TCAD}}$ contributes significantly to improve the overall accuracy of the model.

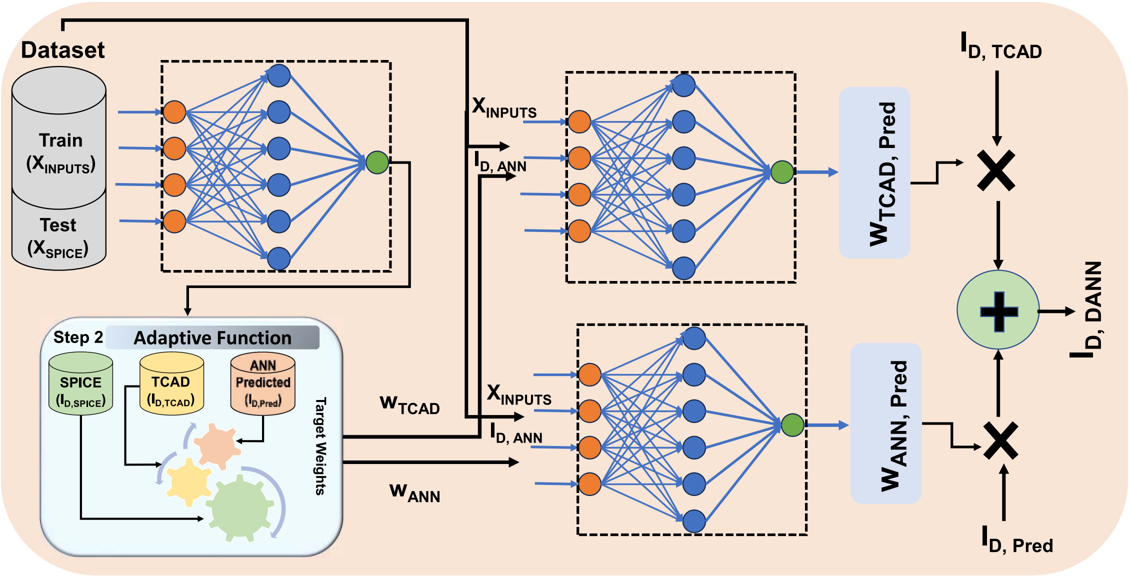

FIGURE 2. Schematic architecture of the proposed dynamic adaptive neural network (DANN). It consists of three artificial neural networks. The first ANN model is fed using spice dataset. The other two ANNs are exploited to predict the weights which are utilized to manipulate the final predicted output, i.e., $I_{DA\text{-}NN}$. The tcad dataset is used to augment the overall accuracy of the model. The adaptive function performs the mapping using spice, tcad and the ANN predicted datasets.

TABLE 1. Model Parameters and Hyperparameters of the DANN With Respect to Process Variability Sources

Hyperparameters	DANN
(I/P, O/P) Dimension	(5, 1)
No. of Hidden Layers	1
No. of Neurons in each layer	$\text{ANN}^{1} = 15$, $\text{ANN}^{w1} = 5$ $\text{ANN}^{w2} = 5$
Activation Function	Tanh, relu
Learning Rate	0.001
Optimizer	Adam
Batch Size	5
Epochs (Fixed)	5000
Early stopping Epoch	683

These generated weights are then input into two other ANNs (i.e., ANN^{w2} and ANN^{w3}) and the models train with data (X_{input}, W_{NN}) and (X_{input}, W_{TCAD}). The final outcome of the DANN is the sum of predicted values through ANN^{W2} and ANN^{W3} multiplied with their corresponding I_D values.

The input space of the network consists of randomly generated geometrical parameters of nanosheet device. During the training process of the proposed framework, the "early stopping" regularization technique is used on the training dataset to improve the generalization of the model and avoid overfitting. The maximum failure epoch is set to 25. The generalization error of every neural network is estimated after each epoch. If the error doesn't decrease further in 25 consecutive epochs, the training immediately stops to avoid over fitting. The list of hyperparameters of our explored ML model is shown in Table 1. The raw simulated data is also preprocessed before directly feeding into the model. Since the output drain current has a very wide range (spans over approximately 10^{-5} to 10^{-11}) therefore to scale the range the logarithmic scaling is applied for high training efficiency. In this context, it is interesting to note that all three ANNs utilized in the DANN framework are shallow networks with a single hidden layer and fewer neurons compared to standard ANNs that utilize two or three hidden layers and have a relatively large number of neurons. This significantly reduces the number of calculations when the model is implemented in Verilog-A for the circuit simulation and minimizes the complexity of the network and speeds up the overall simulation time of circuits.

III. TRADE-OFF BETWEEN TCAD AND SPICE SIMULATION

The tcad tool is the foremost choice to generate training data for ANNs due to its high accuracy, reliable convergence, and design optimization. However, rigorous results require an enormous number of simulations and powerful computers. Thus, the expensive computational efficiency is the bottleneck of tcad simulation. As a result, the overall CM development process has slowed down. Therefore, another tool i.e., the spice simulator, which is fast and creates training datasets rapidly, is used to reduce the data generation time by a significant margin. spice simulation has gained its superiority over tcad simulation due to adequate computational cost. However, the flip side of it is the lack of accuracy as compared to tcad simulations, especially for process variation. Hence, firstly the tcad simulations are performed for GAA NS nFET and pFET to generate the half-training dataset by carefully calibrating against the experimental dataset. Then the calibrated spice simulator is used for the generation of the remaining samples that nearly approximate the tcad characteristics. To do that we performed the local fitting on the nominal device and the spice model parameters were calibrated. As we know, the spice offers local accuracy. Hence, for process variations, the compact model is not able to predict accurate device characteristics with same parameter set. However, computational efficiency is a great advantage of compact model over tcad. Hence, we proposed a model which can utilize the tcad data to augment the overall accuracy in the final output. Therefore, to overcome the inefficiency of both simulation tools, the trade-off is achieved between tcad and spice simulation in our proposed DANN methodology.

IV. OVERVIEW OF DEVICE AND VARIABILITY MODELING

The development of advanced integrated circuits is reaching a critical point. The chip industries are ready to move to new transistor architectures for the high-volume production of chips, even though this brings along new complexities and investments. FinFETs reached their scaling limits when enhancing the drive current by making the fins taller in cell height scaling becomes increasingly challenging. This is when NSFET architecture came into existence. NSFET offers a larger drive current per footprint than fins and provides benefit for further CMOS scaling. Therefore, the transition from FinFETs to NSFET has gradually taken place for sub-3 nm technology nodes.

The NSFET comes from the gate-all-around family. They provide advantages such as better gate controllability, and scalability. They are also superior to FinFET and nanowire because of their low power consumption, high speed, and small size [19], [56]. They offer design flexibility that allows the designer to adjust the effective width of the channel to provide more drive current and speed up the ON-OFF switching of the transistor. A sheet can be made wider to boost the current, whereas a sheet can be made narrower to limit the power consumption. This way, GAA NSFET provides variable designing options to make it possible to match the aggressive scaling of technology nodes.

For this work, 3-D device simulation is performed using the Sentaurus TCAD tool [57] to simulate GAA Si NS devices with 16-nm gate length. To accurately account for quantum confinement effects and electron/hole temperature, a self-consistent solution approach involving hydrodynamic (HD) and density gradient (DG) equations is adopted. Our mobility models are comprehensive and encompass several crucial components [23]. These models include the Philips unified model, high-field saturation model, thin layer model, and Lombardi high-k mobility model, which address phonon and Coulomb scatterings at the channel/insulator interface [19], [58]. To incorporate quasi-ballistic effects, we have integrated the low-field ballistic mobility model into our simulations. Additionally, our analysis considers Hurkx band-to-band tunneling and the Shockley-Read Hall recombination mechanism [20]. The NSFET device structures are simulated after validating with experimental calibration to ensure accuracy. The utilization of these models ensures the accurate behavior of NSFET device in TCAD simulations. Since NSFET suffers from process variations, therefore, to capture device variability is a very challenging and necessary problem. NSFET devices can have different sources of device variability. For this study, we considered the channel length (L), nanosheet thickness (T_{NS}), and nanosheet width (W_{NS}) that have a significant effect on the electrical characteristics of the devices [58]. The device parameters and their mean and standard deviation are L is 16 and ±3, W is 25 and ±3, and T is 5 and ±2. By varying these parameters, various variability nanosheet devices data is generated which is used for training the ML model.

V. RESULTS AND DISCUSSION

In this section, the results of this work are discussed in detail. After the successful training of the proposed model, its performance is assessed by evaluating it for two different tasks. Firstly, the modeling capability of the DANN is assessed by modeling the untrained range or portion of characteristics of the same device curves. Next, the predicting capability of the model is evaluated by predicting the characteristics of different devices that are exclusively used for testing purposes.

A. MODELING CAPABILITY OF THE DANN MODEL

In order to conduct the experiment, first of all, the range of curves (gate voltage, $V_G = 0$ V to 0.7 V) is divided into two parts: 1) training range that is set from $V_G = 0.0$ V to 0.6 V (also called in-model training range) and 2) untrained (or test) range from $V_G = 0.6$ V to 0.7 V (also called out-model training range). This range can be altered according to the model capability, data availability and task in hand. Fig. 3(a) and (b) show the device characteristics with training range and untrained range together for n-type and p-type NSFET, respectively. These figures encompass all the variability sources and display comprehensive results. The predicted training range is represented by using red dots and untrained range is shown with blue dots. The black lines show tcad simulated target data. Generally, ANN based models suffer from limited extrapolation problems which means they might not perform well outside the data range they were trained on. The samples beyond the training range may lead to inaccurate predictions. However, the proposed model shows exceptional performance by modeling the untrained range accurately.

Further, it is important to mention here that, to get in-depth knowledge from the data in training phase, the step size of the training data is set to 4 mV for gate voltage. On the other hand, the drain voltage V_D is varied as 50 mV and 0.3 V to 0.7 V with a step size of = 0.1 V. It is possible to simulate circuits accurately with specific terminal voltages, so drain voltage and gate voltage values are selected for output characteristics and transfer characteristics, respectively. It can be confirmed from Fig. 3(a) and (b) that the model is precisely able to imitate both the trained as well as the untrained range. The corresponding rms error for both n-type and p-type GAA-NSFET devices is reported at approximately 0.1%. The R^2-scores for n-type and p-type are 0.9994 and 0.9989

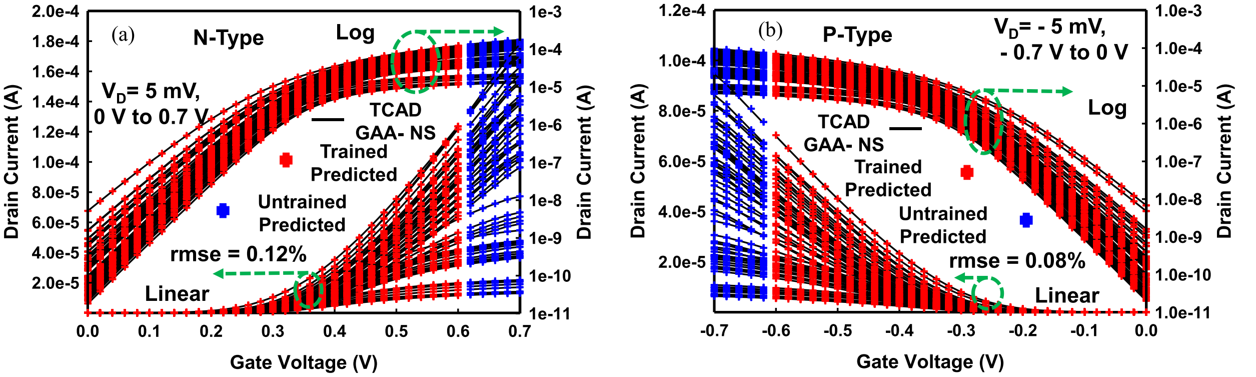

FIGURE 3. Device modeling of (a) n-type and (b) p-type of gate-all-around silicon nanosheet FET using dynamic adaptive neural network. These plots are measured with different V_D values such as 5 mV and 0V to 0.7V. The predicted training range is represented using red dots, predicted untrained range is shown with blue dots and black line shows the TCAD target data in linear and logarithmic scale. The rmse values for n-type and p-type NSFET are 0.12% and 0.08%, respectively.

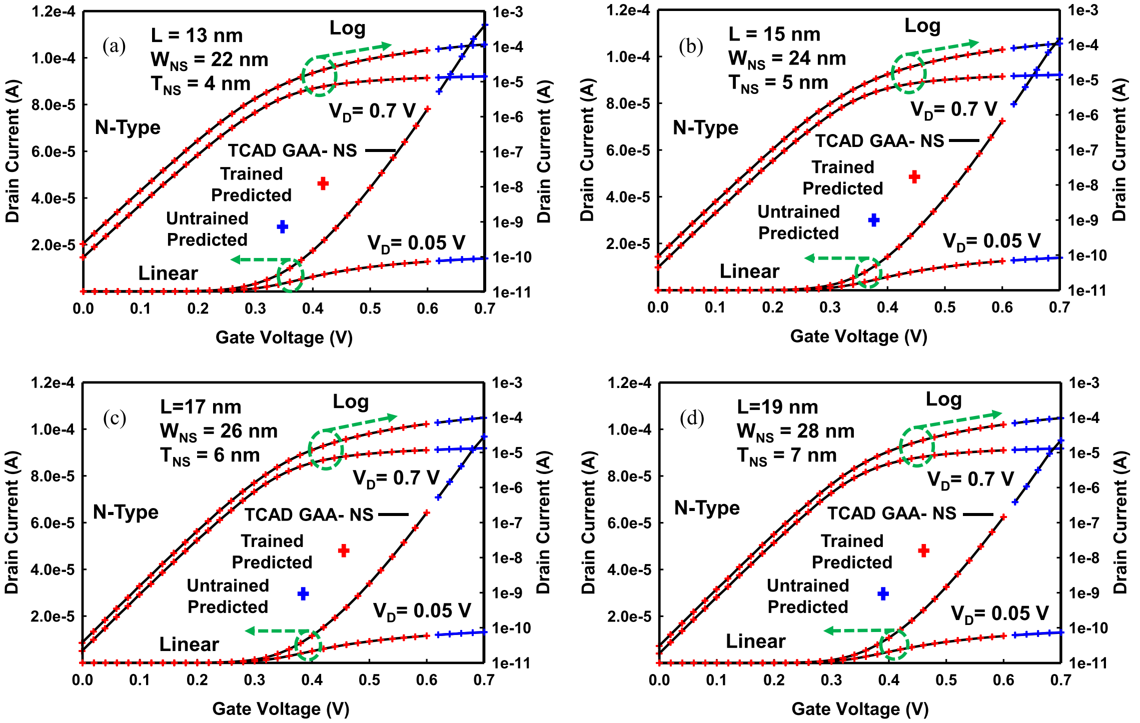

FIGURE 4. Illustration of comparison of simulated (line) vs train predicted (red) and untrained predicted (blue) values in linear and logarithmic scale. The model is trained for V_G ranges from 0 to 0.6V and untrained data ranges from V_{GS} = 0.6 to 0.7 V. Device modeling is depicted for (in nm) (a) L = 13, W_{NS} = 22, T_{NS} = 4 (b) L = 15, W_{NS} = 24, T_{NS} = 5, (c) L = 17, W_{NS} = 26, T_{NS} = 6 and (d) L = 19, W_{NS} = 28, T_{NS} = 7. I_D and V_G are measured with respect to different V_D, i.e., 0.05V and 0.7V.

respectively, which also validate the modeling potential of the proposed model.

To conduct a more detailed analysis, we further investigated multiple n-type devices. These devices had distinct variations in channel lengths, nanosheet widths, and nanosheet thicknesses. The comprehensive findings are visually presented in Fig. 4(a)–(d), both in linear and logarithmic scales. Through these plots, we closely evaluated the model's performance under different bias conditions: linear bias (V_D = 5 mV) and saturation bias (V_D = 0.7 V). The plots illustrate the close alignment between model's predictions and simulated values. The fitting of these curves is notably accurate, showcasing the robust modeling capability of our proposed method. Similarly, this analysis extends to p-type GAA-NSFET devices, depicted in Fig. 5(a)–(b). For each individual device, characterized by specific parameters, the DANN model consistently demonstrates its applicability and accuracy. Furthermore, in Fig. 6(a) and (b) the performance of GAA NSFET is evaluated by modeling the output characteristics (I_D-V_D curves) in similar manner for various V_G

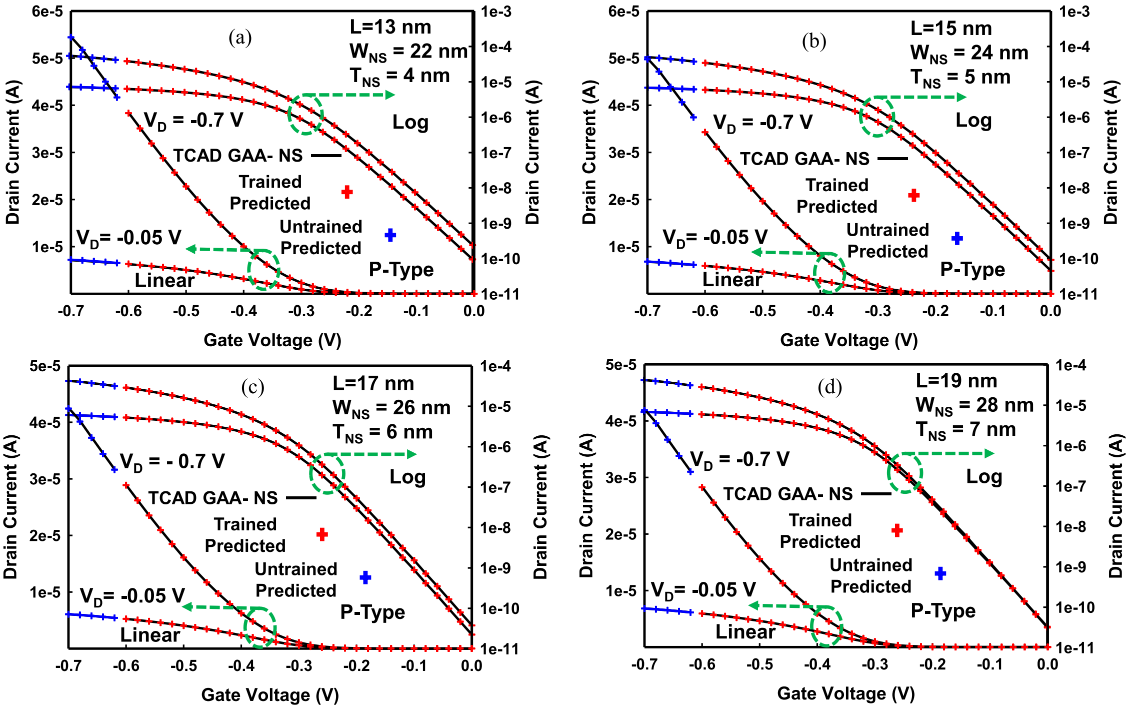

FIGURE 5. Device modeling of p-type gate-all-around silicon nanosheet FET is depicted for (in nm) (a) L = 13, W_{NS} = 22, T_{NS} = 4 (b) L = 15, W_{NS} = 24, T_{NS} = 5, (c) L = 17, W_{NS} = 26, T_{NS} = 6 and (d) L = 19, W_{NS} = 28, T_{NS} = 7 using dynamic adaptive neural network in linear as well as logarithmic scale. In (a)–(d), the drain current and gate voltage are plotted for V_D = 0.05V and 0.7V. These plots show a good agreement between the predicted values and the simulated values.

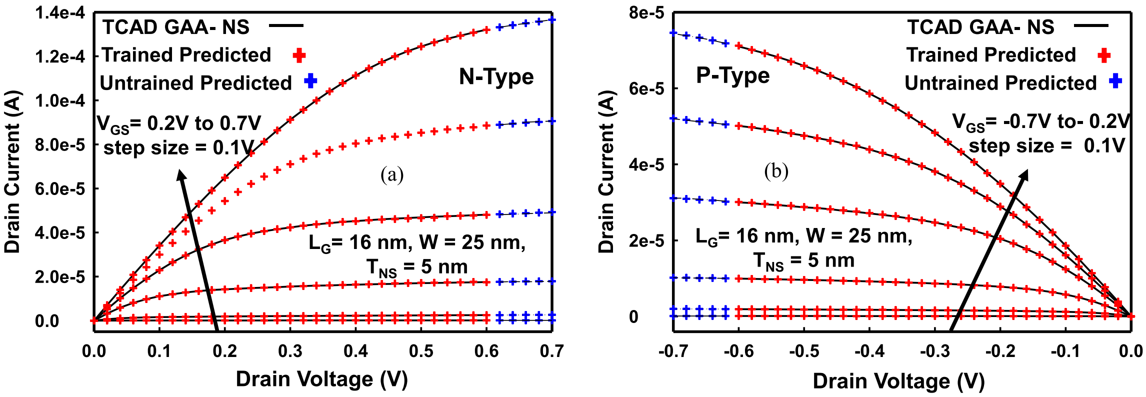

FIGURE 6. Comparison of simulated and predicted output characteristics. (a) and (b) depict the training and testing I_D-V_G characteristics for the simulated and predicted data for n-type and p-type GAA NSFETs, respectively. (-) solid line represents the simulated value of NSFET values and red and blue markers represents the training and untrained range values modeled by the proposed DANN technique. The outputs for the proposed model are nearly equal to true simulated values.

values ranging from 0.2 V to 0.7 V with a step size of 0.1 V for n-type and p-type devices using proposed model. The R^2-score of I_D-V_D curves is 0.9838 and 0.9817 for n-type and p-type GAA NSFET, respectively and the corresponding rms error is reported approximately 0.35%. All these results show that the outputs of the model are in good agreement with the target values. Therefore, as observed from the results, it also indicates that the proposed model learned the trend successfully not only for in-model training range, but it also has a good prediction performance on out-model training range.

B. EXPLORING THE PREDICTIVE EFFICACY OF THE DANN MODEL

In this section we investigate the model's ability to predict new devices. Unlike the previous section, this time we organized the data into different device groups. The training and test dataset has different devices. The voltage range for every

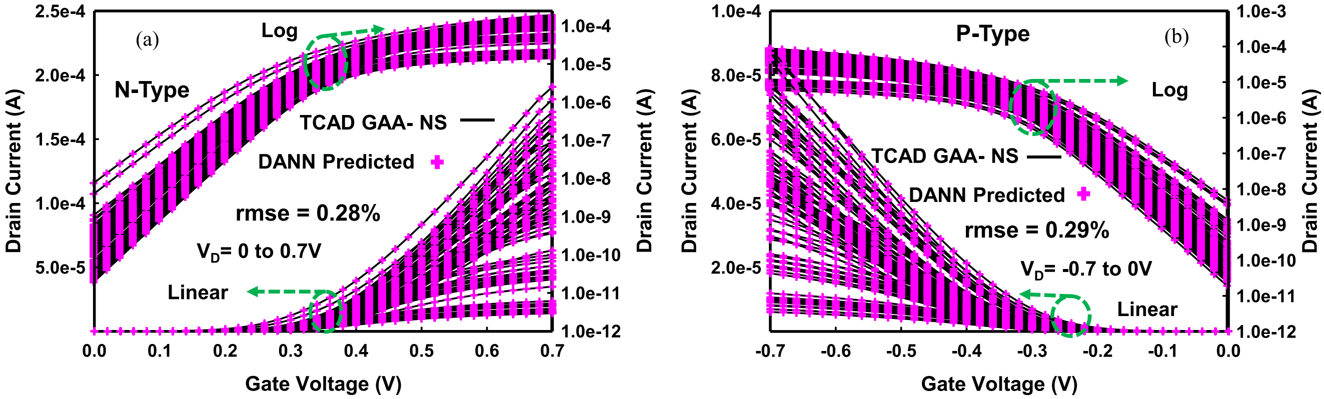

FIGURE 7. Device modeling of (a) n-type and (b) p-type of gate-all-around silicon nanosheet FET using dynamic adaptive neural network. These plots are measured with different V_D values such as 0 V to 0.7 V for n-type and −0.7 V to 0 V for p-type. The plots are shown in linear as well as in logarithmic scale.

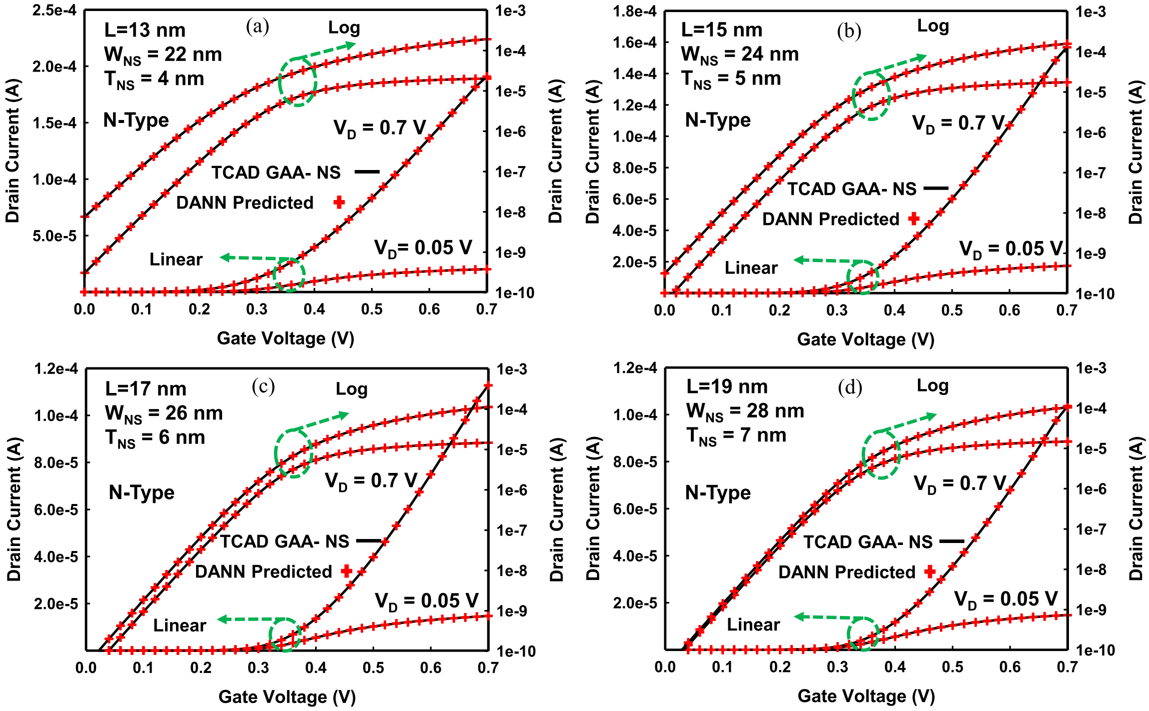

FIGURE 8. Prediction of n-type gate-all-around silicon nanosheet FET is depicted for (in nm) (a) L = 13, W_{NS} = 22, T_{NS} = 4 (b) L = 15, W_{NS} = 24, T_{NS} = 5, (c) L=17, W_{NS} = 26, T_{NS} = 6 and (d) L = 19, W_{NS} = 28, T_{NS} = 7 using dynamic adaptive neural network. In (a)–(d), the drain current and gate voltage are measured with respect to V_D = 0.05V and 0.7V. The model shows excellent fitting of curves.

device either in training set or in test set covers the complete range of gate voltage from 0 V to 0.7 V unlike in the case of modeling. These two sets consist of different device samples with various random dimensions. The proposed model is first trained with the training samples and upon successful completion of this training, the fine-tuned model was then subjected to evaluation using the device samples from the test set.

The outcomes of the DANN predictions for n-type and p-type GAA NSFET device characteristics under process variations are depicted in Fig. 7(a) and (b) respectively. These plots are presented in both linear and logarithmic scales, exhibiting different drain bias conditions. In these figures, red symbols denote the predicted values, while solid lines correspond to the simulated target values. The comparison between these symbols and lines offers a clear insight into the accuracy of our model's predictions in capturing the behavior of both n-type and p-type GAA NSFET devices. The R^2-score for n-type NSFET devices is recorded 0.9991 and for p-type

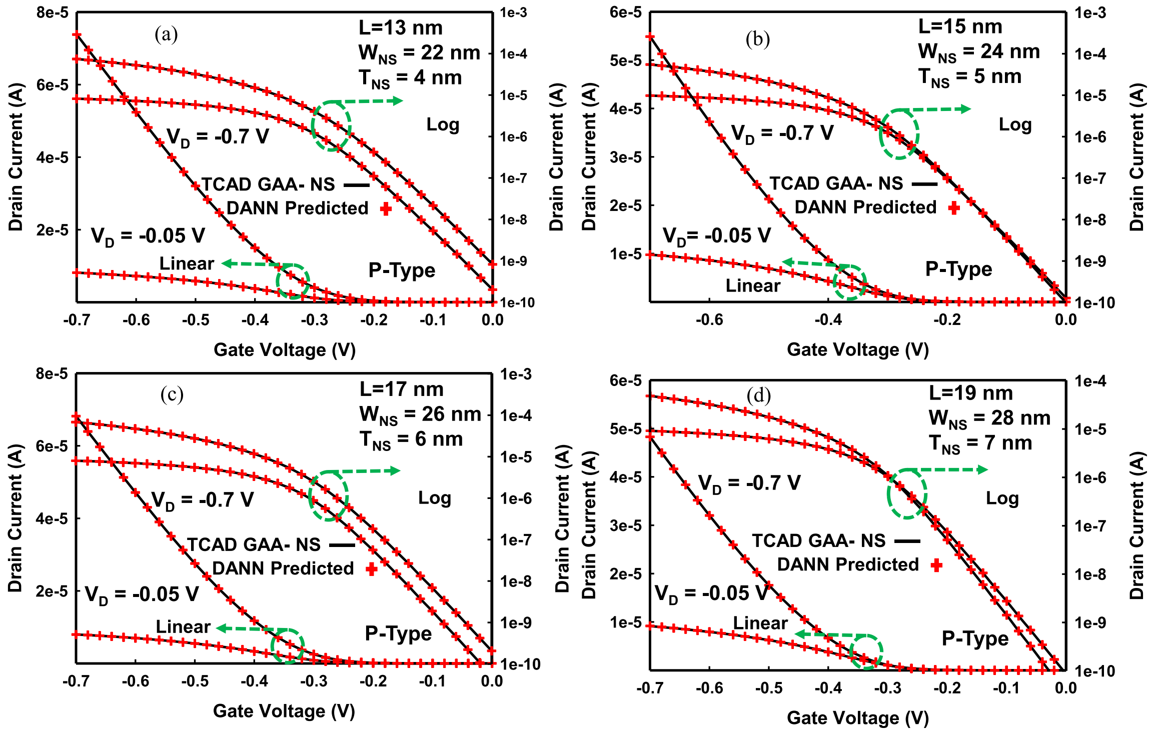

FIGURE 9. Comparison of simulated and predicted output of the proposed model for p-type gate-all-around silicon nanosheet FET is depicted for (in nm) (a) L = 13, W_{NS} = 22, T_{NS} = 4 (b) L = 15, W_{NS} = 24, T_{NS} = 5, (c) L=17, W_{NS} = 26, T_{NS} = 6 and (d) L = 19, W_{NS} = 28, T_{NS} = 7. The solid black lines (-) represent the true simulated values and markers represent the predicted values. The plots show a good agreement between the predicted and the simulated values.

NSFET devices is achieved 0.9993 which are very close to ideal value of 1. Similarly, the error for both the devices is recorded less than 0.3%. We know that the process variation data is complex and hence complex ML models are required to capture the variability.

However, as can be observed from the results, DANN technique, even with simple network structure is able to produce excellent results owing to adaptive weighting strategy. In Fig. 8(a)–(d) four sets of n-type NSFET devices with random process variability are demonstrated for linear and saturation biases for better visualization and understanding. The accuracy can be confirmed from both the linear and the logarithmic scales. Similarly, Fig. 9(a)–(d) show the results for four sets of individual device characteristics for p-type NSFETs. Additionally, Fig. 10(a)–(b) illustrate the output characteristics, I_D-V_D, curves for different gate voltages. In terms of quantitative evaluation, the R^2-score for n-type and p-type GAA NSFET stands 0.9862 and 0.9827, respectively, for the I_D-V_D curves. These scores represent the perfect matching between the model's predictions and the actual observed values. The overall performance of the proposed model is outstanding, with the rms error registering below 0.4%. The presented outcomes effectively demonstrate the ability of the DANN model to accurately capture the expected values for various NSFET devices. The proposed model gives an in-depth analysis of variations that arise due to process variation effects.

C. APPLICATION OF DANN FOR CIRCUITS SIMULATION

To further explore the potential of the DANN model for circuit analysis, it is ported into the spice simulator. To verify the applicability of the ANN model for circuit simulation, the weights and biases of the trained model are extracted and converted to Verilog-A language which can be executed in spice for circuit simulation. Statistical comparisons between the DANN model and simulations are performed for logic circuits based on NSFETs. The performance of DANN-CM is verified by applying it to multiple digital circuits such as inverter and logic gates (OR, NOR, AND, and NAND). Fig. 11(a) shows the circuit structure of inverter that produces logic functions and is the primary component of all integrated circuits (ICs). The inverter contains both n-type and p-type devices, two opposite-polarity, in a complementary manner to speed the switching of capacitive loads. The voltage transfer characteristics are generated from the spice simulator and compared with the predicted outputs of the DANN models as shown in Fig. 11(b). The result indicates perfect matching of characteristics with almost the same results for spice simulation and DANN models.

Similarly, to pave the way for NSFETs and corresponding DANN models for more general digital circuit designs, four basic logic gates (i.e., OR, NOR, AND, and NAND) based on the NSFETs are simulated with DANN, as shown in Fig. 12(a)–(d). The simulated and DANN predicted

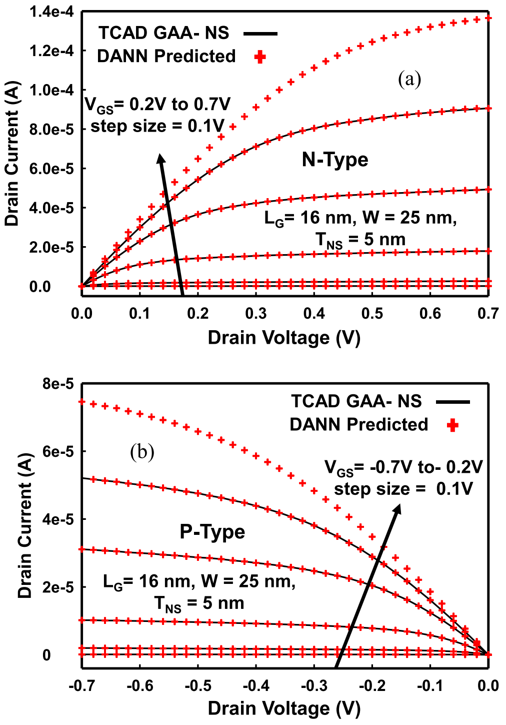

FIGURE 10. Comparison between simulated and predicted output of the proposed DANN model. (a) and (b) shows DANN predicted curves for n-type and p-type, respectively. The gate bias is varied from ±0.2 to ±0.7V.

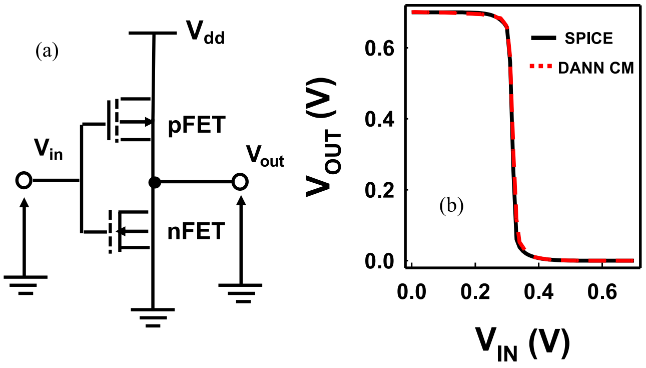

FIGURE 11. Illustration of circuit simulation for NSFET based circuits. The schematic of CMOS inverter is shown in (a) and (b) shows the comparison of voltage transfer characteristics (VTC) between spice and DANN-based inverter. The model shows good agreement between different VTC curves.

waveforms of the OR, NOR, AND, and NAND circuits are shown in Fig. 13(a)–(d). The results for logic gates and DANN model depict overall good agreement with simulation samples which further verifies that the DANN-based CM can be used for future emerging nanodevice circuit simulation and design efforts. These results demonstrate that the proposed DANN model can achieve very good fitting accuracy in the presence of various process parameter variation conditions and shows great potential for circuit simulation.

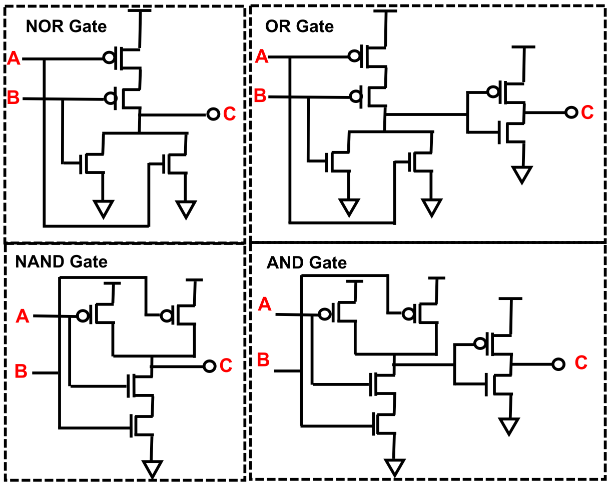

FIGURE 12. It illustrates the schematic of NSFET based logic gates (a) NOR, (b) OR, (c) NAND, and (d) AND gate. They are implemented using gate-all-around nanosheet transistors and then verified by the proposed framework.

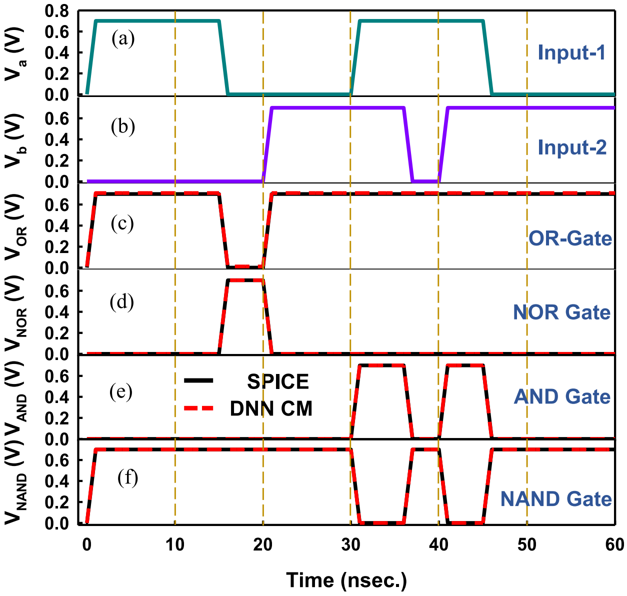

FIGURE 13. It depicts the waveforms of (a) first input (green line), (b) second input (blue line), (c) and (d) are result comparison between spice (black line) and DANN CM (red dash line) for OR gate and NOR gate respectively. Similarly, (e) and (f) are result comparison for AND gate and NAND gate, respectively.

The overall factors that contribute to the high accuracy of the DANN model in the proposed work can be summarized as follows:

- This work adopted hybrid data training approach and because of this approach we leverage the precision of tcad tool to ensure high accuracy.
- The proposed model contributes to its high accuracy by giving attention to identifying and then reducing the critical sources of PV in NS devices.

TABLE 2. Comparison of DANN Methodology With the Past Works That Utilized ANN Models for Circuit Simulation

Examined Features	[45]	[46]	[47]	[48]	Propose work
Explored device	GAA FETs	TFT	MOSFET	GAA Si Nanosheet	GAA Si Nanosheet
Device parameters	L, W, Temp.	L, W	L, W, Temp.	L_G, W_{NS}, T_{NS}, L_{CDD}, Temp.	L_G, W_{NS}, T_{NS}
Training data acquisition	High	High	Very High	High	Low
No. of Hidden Layers/Neurons	2/(20, 15)	3/(10, 10, 10)	3/(32, 32, 32)	2/(10, 10)	(1, 1, 1)/(15, 5 5)
No. of Epochs	5000	-	-	200,000	< 700
Error rate	< 1%	-	< 4%	< 1%	< 0.3%

- The model undergoes a rigorous evaluation process in testing phase, including scalability, and fitting accuracy assessments.
- The innovative technique has adjusted the weights of NN dynamically based on input features. This enables the model to adapt and refine its predictions as it processes data and results in improved accuracy.

Therefore, the combination of hybrid approach, focused process variation analysis, and dynamic weighting enables this model to achieve a high accuracy in predicting and modeling the device behavior.

D. PERFORMANCE COMPARISON AGAINST BASELINE MODEL

To give substance to the performance of the proposed modeling framework, it is compared against baseline model. The conventional static ANN model is constructed as baseline. To build the best baseline model, different hyperparameters combination are tried. We start to train the model with one hidden layer, including the tanh activation function, adam optimizer and 25 neurons. It is examined that the model is not able to estimate the device characteristics due to insufficient model complexity and underfits the data.

Therefore, one more hidden layer is introduced into the network and trained again. We find that the model's performance is improved a lot when trained with two hidden layers. However, as compared to DANN model, this baseline model was still underperforming. When we further increased the hidden layers, the baseline model started overfitting the data due to the large complexity of neural network.

Fig. 14 illustrated the loss function versus number of epochs plots for different baseline ANN models. Similarly, Fig. 15 confirms the prediction performance of DANN over baseline model by assessing high R^2-score and small RMSE values.

E. COMPARISON WITH RELATED WORKS

It is important to discuss the significance of our research by comparing it with other related studies. We have meticulously

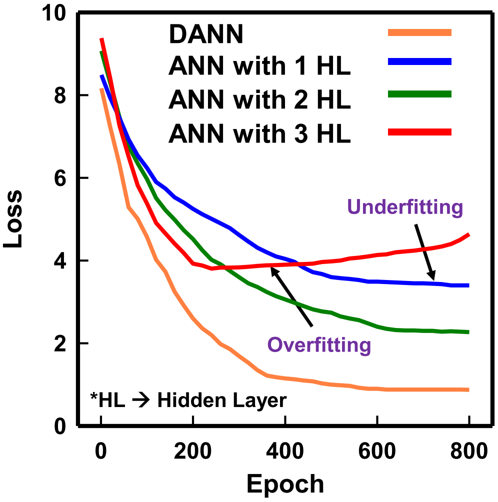

FIGURE 14. Loss function versus number of epochs for the different baseline ANN models. To evaluate the performance of DANN against conventional ANN, various baseline models with different number of hidden layers are trained. The plot shows that ANN with one and three hidden layers suffer from underfitting and overfitting, respectively. The ANN with two-hidden layers performs the best, however its loss is still too high as compared to the DANN model.

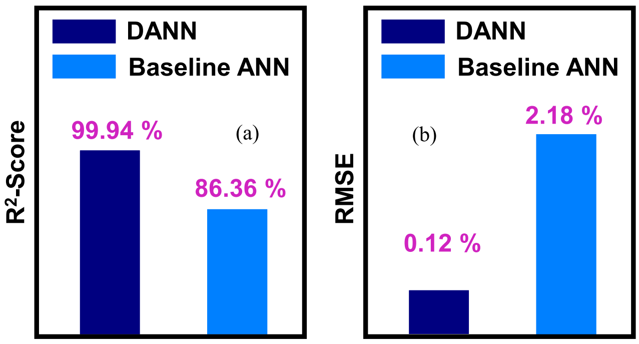

FIGURE 15. Histogram of comparison of (a) R^2-score and (b) RMSE values for DANN and baseline static ANN models. The high R^2-score and the small RMSE values of proposed DANN model shows the well-trained performance as compared to the baseline models.

examined in Table 2, the limitations of previous works [45], [46], [47], and [48], and have introduced a methodology that brings forth the contributions of this research. In this work we innovatively introduced the DANN model, which is trained using precise tcad simulations and optimized for computational efficiency via spice simulations. The proposed DANN model showcases a significant reduction in data acquisition needed for training. It can be employed to implement CM s across multiple logic circuits. Our model has achieved the lowest error rate of less than 0.3% compared to other models.

For this study, tcad simulation takes 30 mins/sample and spice takes 20 ms/sample approximately. To generate the whole data set, 15 days are consumed by tcad. In contrast, the DANN training is accomplished within 45 min. Then, the testing of 200 samples took approximately 5 msec, which is equivalent to 25 microseconds/samples. In other words, the DANN model prediction is approximately a million times faster than tcad simulation if the ML-model is properly illustrated and tested. DANN is more time efficient because all three neural networks used in DANN are shallow networks with only one hidden layer and fewer neurons. The simplest structure of the proposed model also helps to accelerate the time and reduce the computing resources required for the circuit simulation and process engineering.

VI. CONCLUSION

In this work a new methodology DANN is successfully developed for device and circuit simulation of GAA NSFETs. Two different data generation techniques tcad and spice are explored to provide reliable device physics as well as rapid simulations. The model is evaluated for two different tasks. First the modeling capability of the proposed methodology is investigated by predicting the out-model training range with high accuracy for various geometrical variations. Next, the efficacy of the model is explored for the predicting capability by estimating the device characteristics from the unseen dataset. The proposed model validates its fitting accuracy for various geometrical conditions. All the results predicted by the model agree well with the simulated. Finally, the DANN based CM is developed and validated against novel NSFET circuits. The results demonstrate that the DANN can capture the device and circuit characteristics with high precision. In this paper we proposed a fast compact modeling technique for emerging semiconductor nanodevices that provide a solution for the early-stage circuit design and process technology.

REFERENCES

[1] J. N. Tripathi, H. Vaghasiya, D. Junjariya, and A. Chordia, “Machine learning techniques for modeling and performance analysis of interconnects,” *IEEE Open J. Nanotechnol.*, vol. 2, pp. 178–190, 2021, doi: 10.1109/OJNANO.2021.3133325.

[2] S. Soni, G. Verma, H. Nehete, and B. K. Kaushik, “Energy efficient spin-based implementation of neuromorphic functions in CNNs,” *IEEE Open J. Nanotechnol.*, vol. 4, pp. 102–108, 2023, doi: 10.1109/OJNANO.2023.3261959.

[3] S. Kim et al., “Automatic modeling of logic device performance based on machine learning and explainable AI,” in *Proc. Int. Conf. Simul. Semicond. Processes Devices*, 2020, pp. 47–50, doi: 10.23919/SISPAD49475.2020.9241681.

[4] R. Ghoshhajra, K. Biswas, M. Sultana, and A. Sarkar, “Ensemble learning strategy in modeling of future generation nanoscale devices using Machine Learning,” in *Proc. IEEE Devices Integr. Circuit*, 2023, pp. 546–550, doi: 10.1109/DevIC57758.2023.10134917.

[5] H. Zhang, Y. Jing, and P. Zhou, “Machine learning-based device modeling and performance optimization for FinFETs,” *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 70, no. 4, pp. 1585–1589, Apr. 2023, doi: 10.1109/TCSII.2022.3224172.

[6] Y. Guo et al., “Application of machine learning method in the modeling and designing of semiconductor power devices,” in *Proc. IEEE Int. Conf. Solid-State Integr. Circuit Technol.*, 2022, pp. 1–4, doi: 10.1109/ICSICT55466.2022.9963153.

[7] M. Li, O. İrsoy, C. Cardie, and H. G. Xing, “Physics-inspired neural networks for efficient device compact modeling,” *IEEE J. Explor. Solid-State Comput. Devices Circuits*, vol. 2, pp. 44–49, Dec. 2016, doi: 10.1109/JXCDC.2016.2636161.

[8] H.-C. Choi, H. Yun, J.-S. Yoon, and R.-H. Baek, “Neural approach for modeling and optimizing Si-MOSFET manufacturing,” *IEEE Access*, vol. 8, pp. 159351–159370, 2020, doi: 10.1109/ACCESS.2020.3019933.

[9] T. Wu and J. Guo, “Speed up quantum transport device simulation on ferroelectric tunnel junction with machine learning methods,” *IEEE Trans. Electron Devices*, vol. 67, no. 11, pp. 5229–5235, Nov. 2020, doi: 10.1109/TED.2020.3025982.

[10] 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. Simul. Semicond. Processes Devices*, 2019, pp. 1–4, doi: 10.1109/SISPAD.2019.8870440.

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

[12] M.-H. Oh, M.-W. Kwon, K. Park, and B.-G. Park, “Sensitivity analysis based on neural network for optimizing device characteristics,” *IEEE Electron Device Lett.*, vol. 41, no. 10, pp. 1548–1551, Oct. 2020, doi: 10.1109/LED.2020.3016119.

[13] C. Akbar, Y. Li, and W. L. Sung, “Machine learning aided device simulation of work function fluctuation for multichannel gate-all-around silicon nanosheet MOSFETs,” *IEEE Trans. Electron Devices*, vol. 68, no. 11, pp. 5490–5497, Nov. 2021, doi: 10.1109/TED.2021.3084910.

[14] J. Lim, J. Lee, and C. Shin, “Probabilistic artificial neural network for line-edge-roughness-induced random variation in FinFET,” *IEEE Access*, vol. 9, pp. 86581–86589, 2021, doi: 10.1109/ACCESS.2021.3088461.

[15] R. Ghoshhajra, K. Biswas, and A. Sarkar, “A review on machine learning approaches for predicting the effect of device parameters on performance of nanoscale MOSFETs,” in *Proc. Devices Integr. Circuit*, 2021, pp. 489–493, doi: 10.1109/DevIC50843.2021.9455840.

[16] C. Akbar, Y. Li, and W.-L. Sung, “Deep learning algorithms for the work function fluctuation of random nanosized metal grains on gate-all-around silicon nanowire MOSFETs,” *IEEE Access*, vol. 9, pp. 73467–73481, 2021, doi: 10.1109/ACCESS.2021.3079981.

[17] C. Akbar, Y. Li, and W.-L. Sung, “Deep learning approach to inverse grain pattern of nanosized metal gate for multichannel gate-all-around silicon nanosheet MOSFETs,” *IEEE Trans. Semicond. Manuf.*, vol. 34, no. 4, pp. 513–520, Nov. 2021, doi: 10.1109/TSM.2021.3116250.

[18] S. Khandelwal, J. P. Duarte, Y. S. Chauhan, and C. Hu, “Modeling 20-nm germanium FinFET with the industry standard FinFET model,” *IEEE Electron Device Lett.*, vol. 35, no. 7, pp. 711–713, Jul. 2014, doi: 10.1109/LED.2014.2323956.

[19] D. Nagy, G. Espiñeira, G. Indalecio, A. J. García-Loureiro, K. Kalna, and N. Seoane, “Benchmarking of FinFET, nanosheet, and nanowire FET architectures for future technology nodes,” *IEEE Access*, vol. 8, pp. 53196–53202, 2020, doi: 10.1109/ACCESS.2020.2980925.

[20] J.-S. Yoon et al., “DC performance variations by grain boundary in source/drain epitaxy of sub-3-nm nanosheet field-effect transistors,” *IEEE Access*, vol. 10, pp. 22032–22037, 2022, doi: 10.1109/ACCESS.2022.3154049.

[21] N. Loubet et al., “Stacked nanosheet gate-all-around transistor to enable scaling beyond FinFET,” in *Proc. Symp. VLSI Technol.*, 2017, pp. T230–T231, doi: 10.23919/VLSIT.2017.7998183.

[22] Y. Sun et al., “Impact of process fluctuations on RF small-signal parameter of gate-all-around nanosheet transistor beyond 3 nm node,”

IEEE Trans. Electron Devices, vol. 69, no. 1, pp. 31–38, Jan. 2022, doi: 10.1109/TED.2021.3130009.

[23] W.-L. Sung and Y. Li, "Characteristics of stacked gate-all-around Si nanosheet MOSFETs with metal sidewall source/drain and their impacts on CMOS circuit properties," *IEEE Trans. Electron Devices*, vol. 68, no. 6, pp. 3124–3128, Jun. 2021, doi: 10.1109/TED.2021.3074126.

[24] T. Dutta, G. Pahwa, A. Agarwal, and Y. S. Chauhan, "Impact of process variations on negative capacitance FinFET devices and circuits," *IEEE Electron Device Lett.*, vol. 39, no. 1, pp. 147–150, Jan. 2018, doi: 10.1109/LED.2017.2770158.

[25] Z. Zhang, X. Jiang, R. Wang, S. Guo, Y. Wang, and R. Huang, "Extraction of process variation parameters in FinFET technology based on compact modeling and characterization," *IEEE Trans. Electron Devices*, vol. 65, no. 3, pp. 847–854, Mar. 2018, doi: 10.1109/TED.2018.2790083.

[26] P. H. Vardhan, A. Amita, S. Ganguly, and U. Ganguly, "Threshold voltage variability in nanosheet GAA transistors," *IEEE Trans. Electron Devices*, vol. 66, no. 10, pp. 4433–4438, Oct. 2019, doi: 10.1109/TED.2019.2933061.

[27] J.-S. Yoon, S. Lee, J. Lee, J. Jeong, H. Yun, and R.-H. Baek, "Reduction of process variations for sub-5-nm node fin and nanosheet FETs using novel process scheme," *IEEE Trans. Electron Devices*, vol. 67, no. 7, pp. 2732–2737, Jul. 2020, doi: 10.1109/TED.2020.2995340.

[28] A. Gorad and U. Ganguly, "Analytical estimation of LER-like variability in GAA nano-sheet transistors," in *Proc. Int. Symp. VLSI Technol., Syst., Appl.*, 2019, pp. 1–2, doi: 10.1109/VLSI-TSA.2019.8804637.

[29] Y. P. Pundir et al., "Effect of process-induced variations on analog performance of silicon based nanosheet transistor," *Silicon*, vol. 15, no. 1, pp. 4449–4455, Feb. 2023, doi: 10.1007/s12633-023-02365-y.

[30] A. Pouydebasque et al., "High-performance high- K/metal planar self-aligned gate-all-around CMOS devices," *IEEE Trans. Nanotechnol.*, vol. 7, no. 5, pp. 551–557, Sep. 2008, doi: 10.1109/TNANO.2008.2002981.

[31] N. Seoane, J. G. Fernandez, K. Kalna, E. Comesana, and A. García-Loureiro, "Simulations of statistical variability in *n*-type FinFET, nanowire, and nanosheet FETs," *IEEE Electron Device Lett.*, vol. 42, no. 10, pp. 1416–1419, Oct. 2021, doi: 10.1109/LED.2021.3109586.

[32] H. Y. Wong et al., "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.

[33] H. Carrillo-Nunez, 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.

[34] K. Ko, J. K. Lee, M. Kang, J. Jeon, and H. Shin, "Prediction of process variation effect for ultrascaled GAA vertical FET devices using a machine learning approach," *IEEE Trans. Electron Devices*, vol. 66, no. 10, pp. 4474–4477, Oct. 2019, doi: 10.1109/TED.2019.2937786.

[35] Y. Lyu, W. Chen, M. Zheng, B. Yin, J. Li, and L. Cai, "Machine learning assisted nanoscale device modeling for nanosheet FETs with process variations," in *Proc. IEEE Silicon Nanoelectronics Workshop*, 2022, pp. 1–2, doi: 10.1109/SNW56633.2022.9889011.

[36] Y. Lyu, W. Chen, M. Zheng, B. Yin, J. Li, and L. Cai, "Machine learning-assisted device modeling with process variations for advanced technology," *IEEE J. Electron Devices Soc.*, vol. 11, pp. 303–310, 2023, doi: 10.1109/JEDS.2023.3277548.

[37] C.-Y. Huang et al., "3-D self-aligned stacked NMOS-on-PMOS nanoribbon transistors for continued Moore's law scaling," in *Proc. IEEE Int. Electron Devices Meeting*, 2020, pp. 20.6.1–20.6.4, doi: 10.1109/IEDM13553.2020.9372066.

[38] F. Klemme, J. Prinz, V. M. van Santen, J. Henkel, and H. Amrouch, "Modeling emerging technologies using machine learning: Challenges and opportunities," in *Proc. IEEE/ACM Int. Conf. Comput. Aided Des.*, 2020, pp. 1–9, doi: 10.1145/3400302.3415770.

[39] G. Qi et al., "The device compact model based on multi-gradient neural network and its application on MoS2 field effect transistors," in *Proc. IEEE Electron Devices Technol. Manuf. Conf.*, 2022, pp. 88–90, doi: 10.1109/EDTM53872.2022.9798306.

[40] X. Gao, A. Huang, N. Trask, and S. Reza, "Physics-informed graph neural network for circuit compact model development," in *Proc. Int. Conf. Simul. Semicond. Processes Devices*, 2020, pp. 359–362, doi: 10.23919/SISPAD49475.2020.9241634.

[41] Y.-S. Yang, Y. Li, and S. R. Kola, "A physical-based artificial neural networks compact modeling framework for emerging FETs," *IEEE Trans. Electron Device*, early access, doi: 10.1109/TED.2023.3269410.

[42] M.-Y. Kao, H. Kam, and C. Hu, "Deep-learning-assisted physics-driven MOSFET current-voltage modeling," *IEEE Electron Device Lett.*, vol. 43, no. 6, pp. 974–977, Jun. 2022, doi: 10.1109/LED.2022.3168243.

[43] C.-T. Tung, M.-Y. Kao, and C. Hu, "Neural network-based and modeling with high accuracy and potential model speed," *IEEE Trans. Electron Devices*, vol. 69, no. 11, pp. 6476–6479, Nov. 2022, doi: 10.1109/TED.2022.3208514.

[44] Q. Yang et al., "Transistor compact model based on multigradient neural network and its application in SPICE circuit simulations for gate-all-around Si cold source FETs," *IEEE Trans. Electron Devices*, vol. 68, no. 9, pp. 4181–4188, Sep. 2021, doi: 10.1109/TED.2021.3093376.

[45] J. Wei, H. Wang, T. Zhao, Y.-L. Jiang, and J. Wan, "A new compact MOSFET model based on artificial neural network with unique data pre-processing and sampling techniques," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 42, no. 4, pp. 1250–1254, Apr. 2023, doi: 10.1109/TCAD.2022.3193330.

[46] Q. Chen and G. Chen, "Artificial neural network compact model for TFTs," in *Proc. IEEE 7th Int. Conf. Comput. Aided Des. Thin-Film Transistor Technol.*, 2016, p. 11, doi: 10.1109/CAD-TFT.2016.7785057.

[47] 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 Trans. Electron Devices*, vol. 68, no. 3, pp. 1318–1325, Mar. 2021, doi: 10.1109/TED.2020.3048918.

[48] H. Jeong et al., "Fast and expandable ANN-based compact model and parameter extraction for emerging transistors," *IEEE J. Electron Devices Soc.*, vol. 11, pp. 153–160, 2023, doi: 10.1109/JEDS.2023.3246477.

[49] H. Habal, D. Tsonev, and M. Schweikardt, "Compact models for initial MOSFET sizing based on higher-order artificial neural networks," in *Proc. Workshop Mach. Learn. CAD*, 2020, pp. 111–116, doi: 10.1145/3380446.3430632.

[50] J. Arous, D. Ben Ayed, and N. Ellouze, "Cooperative static and dynamic neural networks for phoneme recognition," in *Proc. IEEE 6th Int. Conf. Sci. Electron., Technol. Inf. Telecommun.*, 2012, pp. 847–851, doi: 10.1109/SETIT.2012.6482026.

[51] H. Y. Chan and S. H. Zak, "On neural networks that design neural associative memories," *IEEE Trans. Neural Netw.*, vol. 8, no. 2, pp. 360–372, Mar. 1997, doi: 10.1109/72.557674.

[52] F. Lin, "Supervised learning in neural networks: Feedback-network-free implementation and biological plausibility," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7888–7898, Dec. 2022, doi: 10.1109/TNNLS.2021.3089134.

[53] Y. Han, G. Huang, S. Song, L. Yang, H. Wang, and Y. Wang, "Dynamic neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7436–7456, Nov. 2022, doi: 10.1109/TPAMI.2021.3117837.

[54] Z.-Q. Shen and F.-S. Kong, "Dynamically weighted ensemble neural networks for regression problems," in *Proc. Int. Conf. Mach. Learn. Cybern.*, 2004, vol. 6, pp. 3492–3496, doi: 10.1109/ICMLC.2004.138039.

[55] V. M. Becerra, F. R. Garces, S. J. Nasuto, and W. Holderbaum, "An efficient parameterization of dynamic neural networks for nonlinear system identification," *IEEE Trans. Neural Netw.*, vol. 16, no. 4, pp. 983–988, Jul. 2005, doi: 10.1109/TNN.2005.849844.

[56] W. Lu, P. Xie, and C. M. Lieber, "Nanowire transistor performance limits and applications," *IEEE Trans. Electron Devices*, vol. 55, no. 11, pp. 2859–2876, Nov. 2008, doi: 10.1109/TED.2008.2005158.

[57] Sentaurus Device User Guide, Version O-2018.06, Synopsys, Mountain View, CA, USA, 2018.

[58] S. R. Kola, Y. Li, and M.-H. Chuang, "Intrinsic parameter fluctuation and process variation effect of vertically stacked silicon nanosheet complementary field-effect transistors," in *Proc. Int. Symp. Qual. Electron. Des.*, 2023, pp. 1–8, doi: 10.1109/ISQED57927.2023.10129391.