Circuit convergence study using machine learning compact models

Zheng-Kai Yang, Ming-Hsien Hsu, Chung-Yuan Chang, Ya-Wen Ho, Po-Ning Liu, and Albert Lin

Institute of Electronics Engineering, National Yang-Ming Chiao-Tung University, Taiwan

***Abstract*—Machine learning (ML) compact device models (CM) have emerged as an alternative to physics-based CMs. ML CMs can find a mathematical model close to the device characteristics without much prior knowledge, which saves the time of model formation. Additionally, versatile capabilities such as process-awareness, model merging, and fitting new technologies, promote the usage of ML CMs. While ML CMs draw great attention in CAD, their convergence in SPICE has not been carefully studied. Here different activation functions are used to create ML CMs, and then the circuit convergence is tested. We found that inverse square root unit (ISRU) activation has the best convergence. Besides, gate-to-source and gate-to-drain capacitance is founded to benefit the convergence in transient analysis. The circuit convergence rate is 100% for ISRU, sigmoid, and tanh when the capacitor is present. On the other hand, ISRU significantly outperforms other activation functions in DC sweep, achieving 81% convergence. If quasi-static transient analysis is employed to replace DC sweep, 100% convergence is achieved by ISRU. Due to its superior convergence, ISRU is the most promising for future ML CMs in SPICE.**

***Index Terms*—semiconductor device simulation, machine learning, CMOS, device models**

I. INTRODUCTION

With the evolution of the times, many people pay attention to machine learning. In the early days, humans were not interested in machine learning because machine learning models consume a lot of computing power reference to physical models. Nowadays, as more data or more complex phenomena need to be processed, it is more difficult for physical models to process it in a reasonable manner or in a short time. Metal oxide semiconductor field-effect transistor (MOSFET) is the most basic semiconductor device in circuits. Semiconductor device becomes smaller with Moore's Law, and many characteristics are not well described in emerging areas, which leads to a complex physical model. Besides, the physical model is increasing complexity, so the formation of the physical model takes a lot of time and labor costs[1]. This aspect can cause circuit design companies to be unable to use the most advanced devices to design the circuit in time. As the size of MOSFETs shrinks, the BSIM model has become more complex. When the oxide layer is thicker on the MOSFETs, we do not need to consider the gate leakage current and gate-induced drain leakage (GIDL) effect. At this time, the BSIM3 model is relatively simple. When the oxide layer becomes thinner on the MOSFETs, the device has a gate leakage current and GIDL effect. The BSIM model is expanded with the gate leakage current model and GIDL model [2]. The complexity of the BSIM model also increases with the advanced process. To prevent the short channel effect, halo implantation is available in the process, which makes the electric field and parasitic resistance more complicated [3, 4]. At the same time, the halo implantation is also added to the BSIM model[5]. Afterward, since the shallow trench isolation (STI) process affects threshold voltage and mobility, so the stress effect model is introduced after the BSIM4 model [6]. It can be seen that the device model will become complicated in nano-devices, which leads to a lot of time to compose a physical model. Therefore, researchers are looking after using machine learning to construct a device CM.

The machine learning model is a neural network trained by forwarding propagation and backward propagation [7]. The nonlinearity is formed by activation functions. Specifically, multilayer perceptron (MLP) is a mathematical model which uses a matrix to create an equation. We can adjust the number of neurons and hidden layers, input/output parameters, and activation function. The number of neurons, the number of hidden layers, and the activation function can determine the accuracy and complexity of the model. While physical models require repeated modifications, which cost engineers a lot of time to build physical models. In contrast, engineers only need a database, and then machine learning models can be composed. Among the architecture parameters in ML models, the activation functions play an important role in convergence. Thus, in this work, we study the effect of different activations on circuit convergence.

In literature, there have been some works using ML CMs in semiconductor technologies[1]. Chea-Wei Teo et al. uses the parameters of TCAD and the random forest method to predict the location of the defect[8]. Himmel et al. uses machine learning to complete the etching model [9]. Sidi Bel Abbes extracts the critical parameters of MOSFET and then uses MLP to predict the current value [10]. Creech et al. construct a ML CM for microave devices [11] Chen et. al. use ANN to form a compact model of TFTs, and then V_{gs} and V_{ds}, gate width (W), and gate length(L) are used to predict the current value [12]. Gao et. al. also use NN

to form a device model[13]. Huang et. al. proposed dimension-reduced ANN to reduce the problem of overfitting[14]. Kim uses physics-based loss function and then uses backpropagation to train the device model, and this practice can make the model prediction more accurate [15]. Habal uses voltage and gate length to predict the current value and g_m accurately[16]. While ML has been used extensively in semiconductor process and used in several cases in semiconductor device CMs construction, the application of ML device CMs in SPICE has not prevailed in literature, and few people discuss the application of machine learning models to circuits simulation [17-22]. Samsung and Sandia's paper discusses the accuracy of the ML CMs [18, 19]. V_{gs} and V_{ds} are 0V to 1V with adaptive steps in Sandia's paper. It can be known from the result that when the high-density step value is extracted in the subthreshold, this can make the model prediction more accurate. Samsung's paper proposed that the simulation speed of the ML model is slower than the simulation speed of the traditional model. Hammouda also used ANN and Taylor series to compose a device model, and the device model was put into the circuit simulation. Nevertheless, Hammouda did not mention the convergence issue of ML models in circuit simulation [20].

We choose HSPICE to study the circuit convergence using ML CMs because HSPICE is the most popular SPICE used by industry or academia. We use Verilog-A to build device CMs [23]. Because Verilog-A has defined the fundamental laws, it can save a lot of time for model developers and allow model developers to focus on the device models [24]. While many papers use machine learning to form device models , there are only a few papers that really use machine learning models in circuit simulations [17-22]. Here the focus is on the application of ML CMs to circuits and proposing an activation function for high convergence.

II. METHOD

A. Data collection

BSIM-CMG 110.0.0 is used as the baseline to compare the convergence with ML-based CMs[25]. The ML CMs are fitted to BSIM-CMG IV. V_{gs}, $V_{ds,}$ and the number of fins are the parameters in BSIM-CMG. When collecting the parameters of BSIM-CMG 110.0.0, the gate length is 30 nm, and then we can adjust the number of fins. There are 20 kinds of number of fins from 10 to 200. Subsequently, the fitting is carried in python and Tensorflow. To test the convergence of the circuit, we choose different kinds of activation functions. There are three different types of activation functions [26]. There are ReLu and ISRLU for the discontinuous type and unrestricted output value activation functions. Softplus and Swish possess continuous unrestricted output values. Softsign, ISRU, Sigmoid and hyperbolic tangent possess continuous and restricted output values. Finally, the model is incorportaed into HSPICE using Verilog-A syntax.

V_{gs} and V_{ds} of BSIM-CMG 110.0.0 are set from 0V to 1V with 50mV step for n-MOSFET. On the contrary, V_{gs} and V_{ds} of BSIM-CMG 110.0.0 are set from 0V to -1V with 50mV step for p-MOSFET. We only consider the electrical characteristics caused by the gate and drain and source, and the base should be connected to the source in circuits. Each device has 441 data, so there are 8,820 data for n-MOSFET and p-MOSFET, respectively.

B. Machine learning model formation

After collecting the parameters, the machine learning model is formed in python 3.7.6 [27]. The training model uses MLP to train the device model. Function of MLP is referenced from the Tensorflow 2.0.0[28], Numpy 1.21.2[29], Scikit-Learn 0.24.2 [30], Matplotlib 3.3.1[31], Pandas 1.1.1. First, the data can be shuffled. 90% of data is used as the training set, and10% is used as the test set. When forming the BSIM-CMG model, the input parameters are V_{gs}, $V_{ds,}$ and the number of fins. The number of hidden layers is three, and the number of neurons is fifteen in each hidden layer. The activation functions use ReLu, softsign, hyperbolic tangent, sigmoid, softplus, ISRU, ISRLU, Swish, as expressed by (1a)-(1h) [32].

$$f(x) = \max(0, x) \tag{1a}$$

$$f(x) = \frac{x}{1+|x|} \tag{1b}$$

$$f(x) = \mathrm{Tanh}(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{1c}$$

$$f(x) = \frac{1}{1+e^{-x}} \tag{1d}$$

$$f(x) = \ln(1+e^{x}) \tag{1e}$$

$$f(x) = \frac{x}{\sqrt{1+x^{2}}} \tag{1f}$$

$$f(x)=\begin{cases}\dfrac{x}{\sqrt{1+x^2}} & (0>x)\\ x & (0\le x)\end{cases} \quad (1g)$$

$$f(x)=\frac{x}{1+e^{-x}} \quad (1h)$$

$$y=\sum_{i=1}^{n} v_i(\text{Activation}(w_i x+b_i))+c \quad (2)$$

When the device model is trained, the data should be normalized first. The machine repeats gradient descent and weight update until the gradient of the loss function does not decrease, early stopping can be set.

After the model is trained, the training data and testing data are fed to the models, and root mean square error (RMSE) between the estimated and actual values. This can determine the accuracy of the model.

$$\text{RMSE}=\sqrt{\sum_{i=1}^{n}\frac{(y_{i(\text{real})}-y_{i(\text{predict})})^2}{n}} \quad (3)$$

C. Verilog-A expression

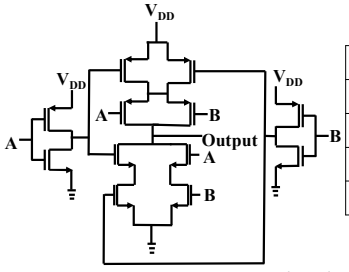
First, the device name, and the device nodes, and the device parameters are defined [23, 33, 34]. Then the weights of the ML model are introduced into the verilog file. MOSFET has three nodes. Device parameters are V_{ds}, V_{gs}, and the number of fins for CMG-MOSFETs, and *L* is uniformly 30nm in this work

Loop expression can form a layer of neuron calculation, and the number cycles of the loop are determined by the number of neurons in a layer. Since the ML model of output value is standardized, it needs to be returned to the original value after the model calculation is completed. Finally, we use the current expression to complete the device model. The output value of the ML model is current in unit of ampere. While we only fit the characteristics of the first quadrant of V_{ds}-V_{gs}, which can cause the circuit simulation to be distorted to the third quadrant. Therefore, we add the third quadrant current in Verilog-A by flipping the current values for negative V_{ds} since the device is symmetric in structures.

D. HSPICE simulation

We use CPU Intel core i7-4790 3.6GHz to test the simulation time of the circuit. The circuit simulator uses the Newton-Raphson method to find the convergence value. If roots of the system of non-linear equations can be found quickly, the function needs to be continuous and smooth. Machine learning has many activation functions with different properties, and each activation function has different smoothness and continuity. As a result, the selection of activation functions has critical effect on the circuit convergence. After constructing Verilog-A models, the device CMs are introduced into HSPICE. We chose a logic gate with a more complicated configuration to test the convergence. As shown in Fig. 1 XOR, instead of common NAND, NOR, or NOT, gate is used. The speed of the ML models are also compared in addition to the convergence.

When we simulate the circuit, the high potential operates at 1V, and the low potential operates at 0V. First, Matlab is used to generate four hundred circuit files with different proportions of device sizes. Then we use batch files to automatically run circuit tests with 400 different ratios of device sizes. Afterward simulation, Matlab R2021a is used to read the waveforms from Cscope. Hspice Toolbox for Matlab and Octave is used so that Matlab can read HSPICE output files[35].



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Fig. 1. XOR circuit and Truth table.

We use transient analysis or DC sweep to test the convergence and simulation speed. In transient simulation, the total transient time is 10ns using sinusoidal stimuli. The amplitude is 0.5V with 0.5V offset. The frequency is set to 2GHz/1GHz for two inputs. We use 400 different n-MOSFET and p-MOSFET ratios to test the convergence of circuits in transient analysis. Then the circuit convergence ratio of each activation function can be displayed together with the simulation time. Afterward, we can choose three activation functions with better convergence, and then using these activation functions to test the subsequent analysis in HSPICE.

The gate to source and gate to drain capacitance, C_{gs} and C_{gd}, is added to the device model, and then the circuit convergence can be tested again.

After transient simulation, the DC sweep is then tested with the scanning range of the DC sweep from 0V to 1V at two input terminals with 0.1V step. Because the convergence in DC sweep is not 100% for any of the activation functions, we use quasi-static transient analysis to replace the DC sweep as illustrated in Fig. 2. The maximum step size of transient analysis is set to 0.016sec. To improve the convergence, the input waveform is carefully ramped. Input A is a triangular wave, and input B will slowly rise from 0V to 1V. When the voltage of input A is from 0V to 1V, the voltage of input B is fixed to a value. When the voltage of input A changes from 1V to 0V, the voltage of Input B is slowly raised by 0.05V. When the voltage of Input B reaches. The calculation is repeated until 400 different W/L ratios are simulated.

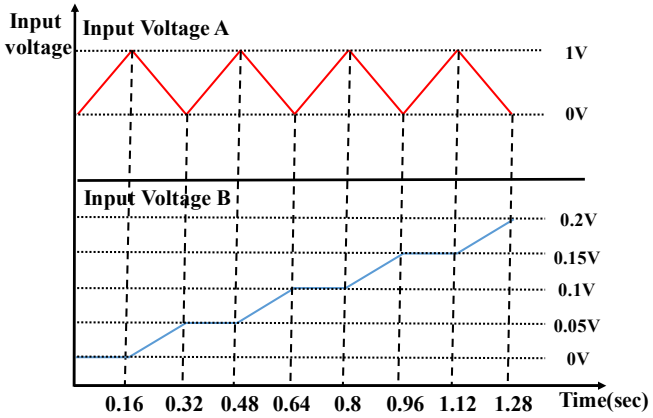


Fig. 2. Quasi-static transient analysis is used to replace the dc sweep.

E. Capacitance model

In our ML CMs, increased capacitance of C_{gs} and C_{gd} in general improves the convergence of the circuit. This is a positive, desired phenomenon, and the problem is that what is the range of C_{gs} and C_{gd} values in realistic devices. C_{gs} and C_{gd} values can be reference from literature [36-43]. The reasonable capacitance values, based on our literature review, fall into the orders of magnitude in 1×10^{-8}F/m - 1×10^{-12}F/m. In fact, the C_{gs} and C_{gd} consists of overlap capacitance (C_{ov}) and channel capacitance (C_{ch}). C_{ov} is a fixed value while C_{ch} is bias dependent. To estimate the worst cases and to see the effect of capacitance orders of magnitude, we ignore C_{ch}, and only retain a fixed-value C_{ov}. Different orders of magnitude of C_{gs} = C_{gd} =C_{ov} are tested, and convergence performance is recorded.

III. RESULTS AND DISCUSSION

A. Fitting ability of ML model

It can be known from Fig. 3 and Fig. 4 that the ML model can accurately predict the current values of the MOSFETs. Different activation functions can also lead to different accuracy in ML model predictions. It can be known from Table I. that the ML model using Swish, the fitting capability is the best for n-MOSFET. On the other hand, ISRLU fits p-MOSFET the best. It can be found from MAPE that the activation functions with larger output ranges has a better fitting ability. On the contrary, the activation functions with limited output ranges can be less effective in fitting. When the input of Swish and ISRLU is >>0, the output value has a linear rise. This can make the output values relatively large, and the fitting ability is better. In addition, it is observed that the prediction accuracy of n-MOSFETs is better than that of p-MOSFETs. To test the predictive ability of the ML model for p-MOSFET, we have tried different loss functions in (4).

$$\sum_{i=1}^{n}\frac{(\log_{10}(I_{Real})-\log_{10}(I_{Predict}))^2}{n} - W\sum_{i=1}^{n}\frac{(I_{Real}-I_{Predict})^2}{n} \tag{4}$$

When W is 0, (11) is equivalently log(I) fitting. W can also be adjusted, from 1 to 10^{10}, to form a balanced fitting between linear and log scales. Based on our trial the results, it can be found that different loss functions are still difficult to improve the MAPE of the ML model for p-MOSFET. Using W from 1 to 10^{10} improves the train set MAPE but does not have much effect on the test set data MAPE. It is seen that the predictive ability of the ML model is relatively poor in the vicinity of 0V. We have also tried to use all the data into training, and then the accuracy of the ML model prediction can be improved. Nevertheless, when all of the data is put into training, overfitting can exist. Thus, we finally use the original loss function (W=0), and 90% of the data is used as training sets.

Table I
MAPE OF ML MODELS

Activation function	n-MOSFET	p-MOSFET
Sigmoid	2.76%	10.69%
ISRU	1.38%	10.52%
ReLu	3.05%	7.04%
ISRLU	1.02%	6.78%
Swish	0.73%	9.47%
Softsign	1.58%	9.66%
Tanh	1.38%	9.33%
Softplus	1.78%	9.39%

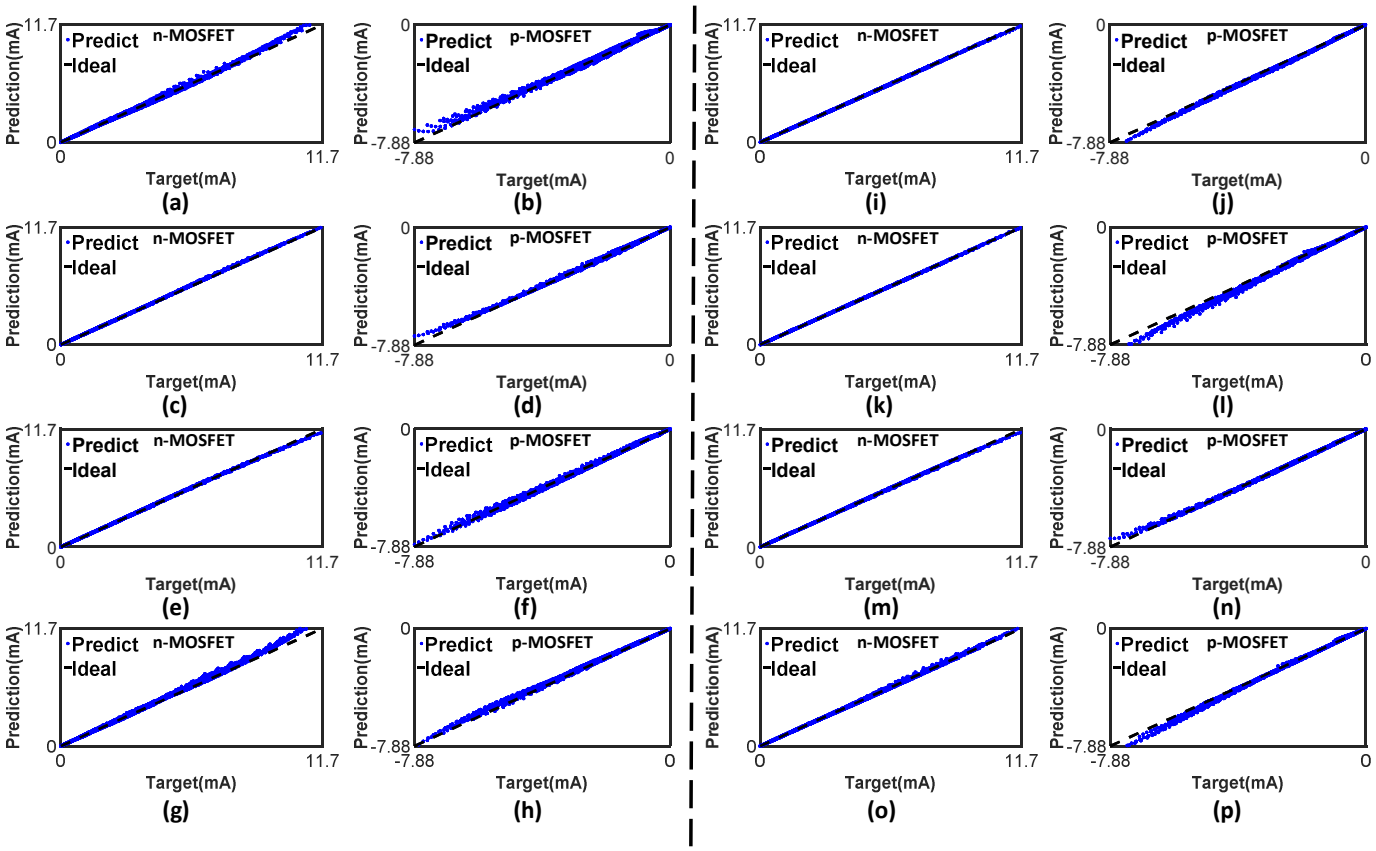


Fig. 3. The scatter plots for different activations (a)-(b) Sigmoid (c)-(d) ISRU (e)-(f) Softsign (g)-(h) ReLu (i)-(j) ISRLU (k)-(l) swish (m)-(n) Tanh (o)-(p) Softplus.

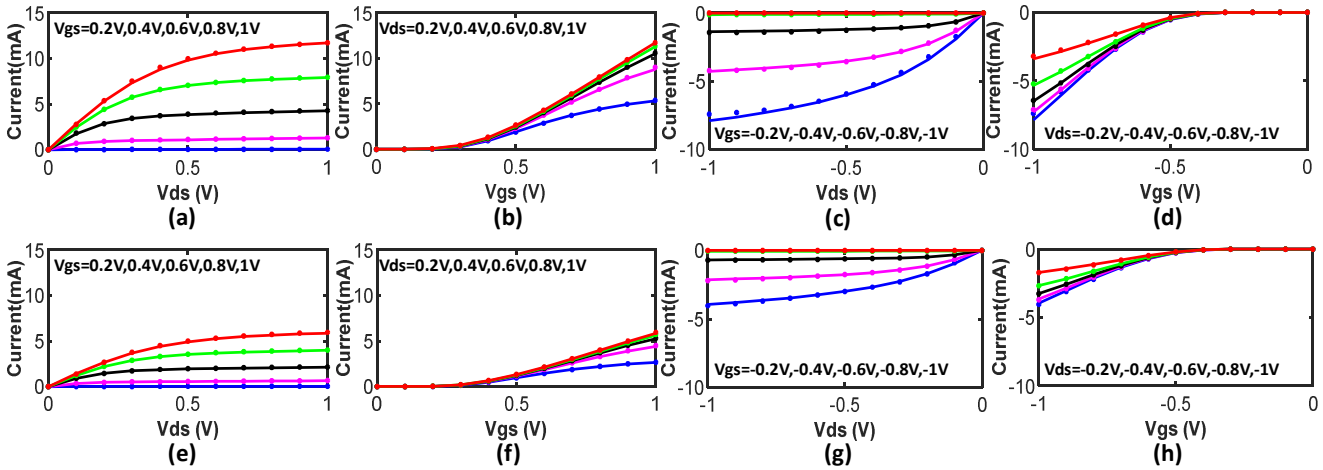


Fig. 4. The ML CMs fitting to BSIM-CMG. (a)-(b) n-MOSFET, L=30nm, NFIN= 200. (c)-(b) p-MOSFET, L=30nm, NFIN= 200. (e)-(f) n-MOSFET, L=30nm, NFIN= 100. (g)-(h)p-MOSFET, L=30nm, NFIN= 100. Activation function is ISRU.

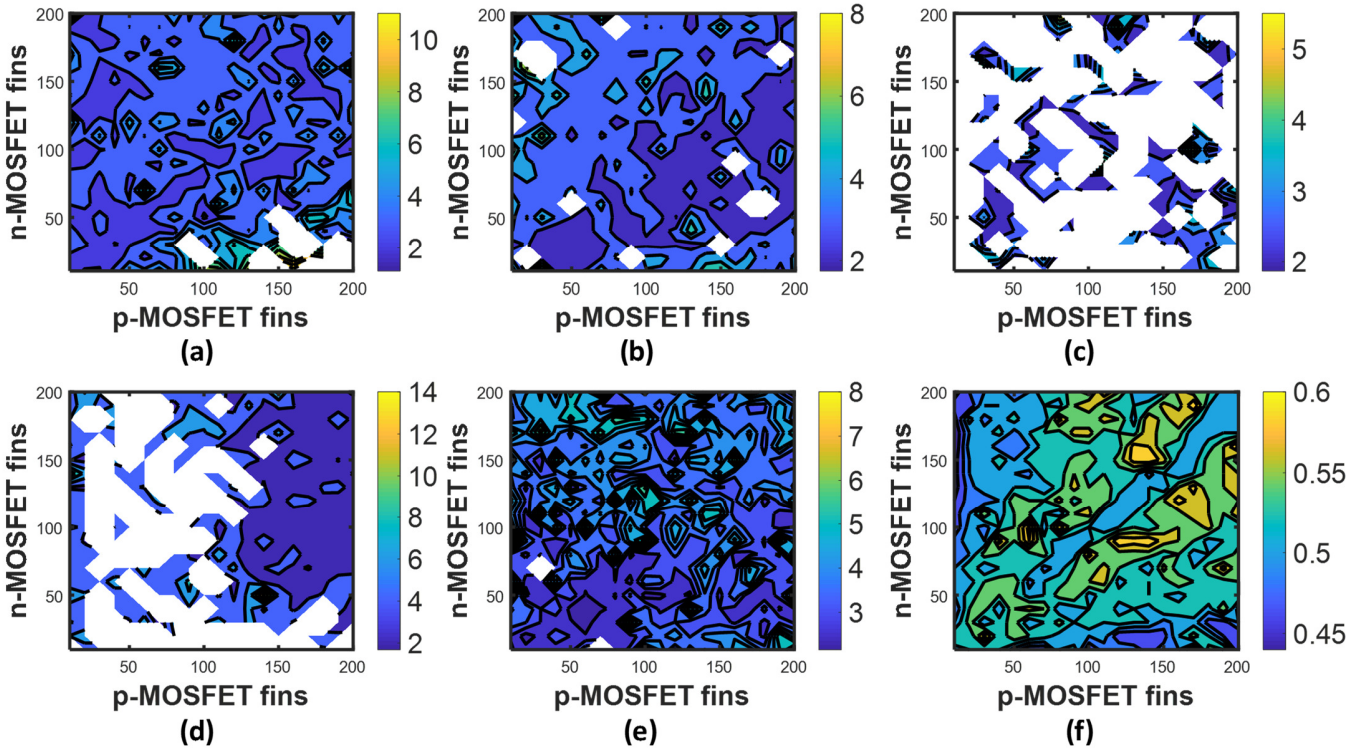


Fig. 5. The contours of simulation time (a) Sigmoid, (b) ISRU, (c) ISRLU, (d) Softsign, (e)Tanh, and (f) BSIM.

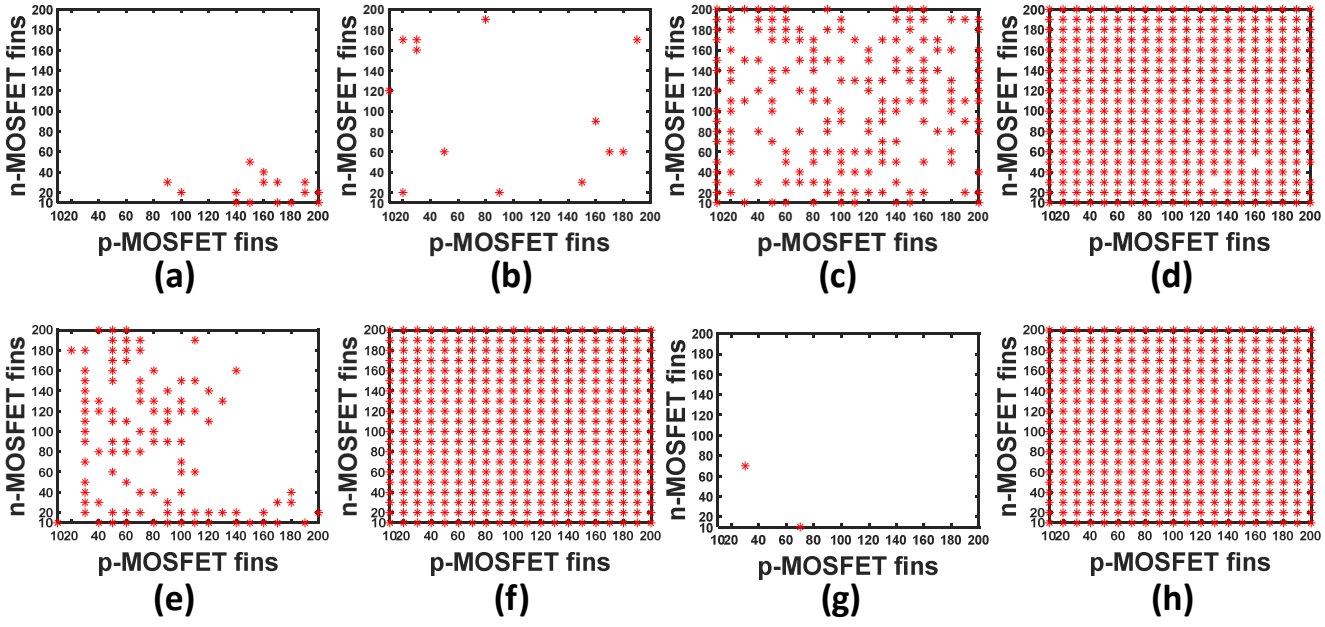


Fig. 6. The scatter plots of the cases w/o convergence, for (a) Sigmoid, (b) ISRU, (c) ISRLU, (d) ReLu, (e) Softsign, (f) Swish, (g) Tanh, and (h) Softplus.

B. Convergence of ML model for transient analysis

Before we test the convergence of the circuit, we check whether the current is minimal to V_{gs} less than 0 and whether the current has an explosive current to V_{gs} greater than 1V. The inspection result shows that the current is smaller than 1×10^{-10}A for V_{gs} less than 0V, and the current is not explosive for V_{gs} greater than 1V. Therefore, in case the node voltage during New-Raphson iterations

is accidentally entering a voltage range outside training, there should not be exploding value problems. As a result, the divergent circuit simulations are more likely due to the ML CM itself.

Before we consider the third quadrant current, we have tested XOR, NAND, NOR, and Inverter. The test result is that the circuit with XOR has the worst convergence, because the XOR circuit is more complicated. The circuit convergence of NAND, NOR, and Inverter is much improved, but it still follows different activation functions, leading to different circuit convergence. In general, activation functions with smooth co-domain, smooth derivatives, and bounded co-domain can lead to better convergence. As far as XOR is concerned, circuit simulation is unable to converge if the third quadrant current in transistor is not specified in ML CMs. Therefore, we added the current in the third quadrant to the ML CMs assuming symmetric device structures, and then the circuit convergence was tested again. The test results show that ISRU, Sigmoid, Softsign, and hyperbolic tangent SPICE convergence are satisfactory in transient analysis, as shown in Table II. The correlation between the average speed in SPICE and convergence, for different activation functions, is actually not very pronounced. The results in Table II show that the average time of circuit simulation of ReLu and ISRLU is relatively fast, but their circuit convergence is rather poor.

When we look for a suitable activation function for HSPICE, we expect to find an activation function that possesses a continuous function values and derivatives. If an activation function has a discontinuity, which can cause the circuit to diverge at the moment of potential transition, such as ReLu. The ReLu function has a discontinuous first derivative at zero, making it impossible to be differentiable. ISRLU function is continuous until the second derivatives, but carefully processed continuity does not improve the SPICE convergence[44]. Another problem of activation functions in SPICE is an unbounded function values, which can cause convergence problems. This is seen in RELU, ISRLU, Swish, and Softplus. Because the value changes dramatically, and errors can accumulate and propagate, which can cause divergence. ISRU, Sigmoid, and hyperbolic tangent are continuous and differentiable functions without piecewise definitions. Besides, the function values are bounded, so the convergence of ISRU, Sigmoid, Softsign, and hyperbolic tangent is better than ReLu, ISRLU, Swish, and Softplus. The derivatives of ISRU, Sigmoid, Softsign, and hyperbolic tangent are in (5a)-(5d), respectively.

Table II

ML MODEL WITHOUT CAPACITANCE (TRANSIENT TIME:10NS, SINUSOID WAVE, THE AMPLITUDE OF 0.5V, THE FREQUENCY OF AC VOLTAGE IS 2GHZ AND 1GHZ).

Different activation function of ML model and BSIM	Convergence ratio	Total convergence /Total test	Simulation time
BSIM	100%	400/400	0.52sec
Sigmoid	96.25%	385/400	3.66sec
ISRU	96.75%	387/400	3.38sec
ReLu	0.75%	3/400	2.78sec
ISRLU	56%	224/400	2.78sec
Swish	0%	0/400	failed
Softsign	74.75%	299/400	4.54sec
Tanh	99.50%	398/400	3.79sec
Softplus	0%	0/400	failed

$$f'(x) = \left(\frac{1}{\sqrt{1+x^2}}\right)^3 \tag{5a}$$

$$f'(x) = \frac{e^{-x}}{(1+e^{-x})^2} \tag{5b}$$

$$f'(x) = \frac{1}{(1+|x|)^2} \tag{5c}$$

$$f'(x) = 1 - \left(\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}\right)^2 \tag{5d}$$

From Table II, it can be found that the ML models of all activation functions are slower than the BSIM model in the circuit simulation [18, 19]. Nevertheless, the BSIM model has many capacitors. The capacitors have differential expressions so that the variation in current and voltage is slowed down, which is beneficial for circuit convergence. Fig. 5 shows the circuit simulation speed of different ratios between n-MOSFET and p-MOSFET. Circuit simulation speed does not have to specific correlation with respect to the n-MOSFET to p-MOSFET *W*/*L* dimension ratio, i.e., n-MOSFET to p-MOSFET dimension ratio has no significant relevance to the circuit convergence. Even if the n-MOSFET to p-MOSFET dimension ratio does not match, the ML models convergence is not deteriorated. On the other hand, the BSIM model does not have any problem in convergence in all n-MOSFET

to p-MOSFET dimension ratios. The scatter plots for divergent cases in terms of the n-MOSFET to p-MOSFET *W/L* dimension ratio is shown in Fig. 6.

C. Convergence for capacitors of ML model

Capacitors are added to the ML CMs in this section to promote the transient analysis convergence. Only practical capacitance values in C_{gs} and C_{gd} can be added to prevent overly large, unrealistic scenarios. From Table III, we can find that the convergence of the circuit has been significantly improved. Since the capacitor can help the voltage to change more slowly, improved circuit convergence is achieved. The capacitance value also deserves some discussion. Certainly, we prefer large capacitors, since small capacitance is regard as an open circuit, which has no effect on the device model and SPICE. The capacitance values, however, needs to be realistic to make the calculation here meaningful. From literature, we found that the reasonable range of C_{gs} and C_{gd} is 1×10^{-8}F/m -1×10^{-12}F/m, while most of the reported MOSFET values falls around 1×10^{-10}F/m. It can be seen from Table III that when the capacitance is 1×10^{-12}F/m, ML models does not converged in all cases. In addition, the results show that the larger the capacitance values are, the more pronounced the improved circuit convergence is. In the capacitance calculations, $I=C\times dV/dt$. dV/dt *does not* directly affect the continuity of the current, while *C* can affect current continuity if the capacitance values have discontinuities. Fortunately, although discontinuous capacitance values can cause current value discontinuities, the effect is not very pronounced for C_{gs} and C_{gd} in the range of 1×10^{-8}F/m -1×10^{-12}F/m. Therefore the continuity of the capacitance values are less critical than the capacitance value in the device model.

In addition to a single XOR gate, we also connect the XOR circuit in series, i.e. two XOR at the first stage connected to one XOR at the second stage, and then we use ISRU ML models to test the convergence of the circuit. The test result is that the 100% convergence is seen when the capacitance value is 1×10^{-10} F/m. Besides, the deterioration of circuit convergence occurs when the device model does not include C_{gs} and C_{gd}. This results in only 87% circuit convergence.

Table III

ML MODEL WITH CAPACITANCE (TRANSIENT TIME:10NS, SINUSOID WAVE, AMPLITUDE OF 0.5V, THE FREQUENCY OF AC VOLTAGE IS 2GHZ AND 1GHZ).

Capacitance	Activation function	Convergence ratio	Total convergence /Total test
1×10^{-12} F/m	Sigmoid	99.75%	399/400
	ISRU	99.75%	399/400
	Tanh	99.75%	399/400
1×10^{-11} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400
1×10^{-10} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400
1×10^{-9} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400
1×10^{-8} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400

D. ML model for DC sweep

When we do dc sweep simulation, we can find that ISRU has the best circuit convergence and Tanh and Sigmoid have worse circuit convergence, as shown in Table IV. The D.C. sweep in Table IV is not 100% even for ISRU, and thus we manage to use transient simulation to replace the D.C. sweep to promote the usage of ML CMs. In general, the dc sweep ramping can be discontinuous if we have nested sweep at two nodes. To circumvent this problem and utilize the previous SPICE solution for the next step, we design the input waveform for quasi-static transient. The results of using quasi-static transient analysis to replace dc sweep are shown in Table V. It can be seen from the results that using transient analysis can significantly improve the convergence. Because the input voltage gradually ramped and the discontinuity in nested sweep is resolved, the convergence of the circuit is significantly improved. Similar to Table III in the previous section, here we have tested different practical capacitor values in C_{gs} and C_{gd} to show the feasibility of the proposed methodology. Since quasi-static transient analysis is operated at low frequencies, capacitance has less effect compared to the results in Table III. Nonetheless, we can still see ISRU has 100% convergence for all capacitance values while sigmoid and tanh fails at a small proportion at some capacitance values.

Table IV

ML MODEL WITHOUT CAPACITANCE (DC SWEEP).

Activation function	Convergence ratio	Total convergence /Total test
Sigmoid	54.25%	217/400
ISRU	81.00%	324/400
Tanh	11.25%	45/400

Table V
ML MODELS WITH CAPACITANCE (USING QUASI-STATIC TRANSIENT FOR DC SWEEP)

Capacitance	Activation function	Convergence ratio	Total convergence /Total test
1×10^{-12} F/m	Sigmoid	97.25%	389/400
	ISRU	100%	400/400
	Tanh	100%	400/400
1×10^{-11} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400
1×10^{-10} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	99.75%	399/400
1×10^{-9} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400
1×10^{-8} F/m	Sigmoid	100%	400/400
	ISRU	100%	400/400
	Tanh	100%	400/400

III. CONCLUSION

The problems in convergence and simulation speed need to be overcome when machine learning is applied to the model of semiconductor devices. Specifically, the convergence of circuit simulations using ML CMs still cannot be compared with the convergence in conventional physical models. Even with the best activation function, i.e. ISRU, the circuit still cannot fully converge. Here we found that to improve the SPICE convergence, the activation functions should be continuous and possess smooth derivatives with bounded function values. Sigmoid, ISRU, and hyperbolic tangent have continuity, smooth derivatives and a bounded co-domain, and thus the circuit converges easier. The circuit convergence ratio of hyperbolic tangent is 99.50% for transient analysis without C_{gs} and C_{gd}. The circuit convergence of sigmoid and ISRU is 95%. Adding C_{gs} and C_{gd}. in transient simulation further improves the convergence. When a realistic capacitance values for C_{gs} and C_{gd} is added to the MOSFET device model, the circuit convergence of tanh, ISRU, and sigmoid is 100% if C_{gs} and C_{gd} is $>10^{-11}$ F/m. Additionally, it is found that the circuit convergence of dc sweep is worse than the circuit convergence of transient analysis. In DC sweep, ISRU significantly outperforms other activation functions. If a quasi-static transient analysis is used to replace DC sweep, and realistic capacitance values are specified in ML MOSFET CMs, ISRU can provide convergent results for all purposes in circuit simulation, including DC and transient analysis. Therefore, in terms of our numerical study, ISRU has the best overall convergence and can potentially be more suitable for future machine learning based CMs in circuit simulation.

REFERENCES

[1] M. Li, O. İrsoy, C. Cardie, and H. G. Xing, "Physics-Inspired Neural Networks for Efficient Device Compact Modeling," *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits,* vol. 2, pp. 44-49, 2016.

[2] C. K. Dabhi, A. S. Roy, and Y. S. Chauhan, "Compact Modeling of Temperature-Dependent Gate-Induced Drain Leakage Including Low-Field Effects," *IEEE Transactions on Electron Devices,* vol. 66, pp. 2892-2897, 2019.

[3] M. Xu, H. Zhu, L. Zhao, H. Yin, J. Zhong, J. Li, *et al.*, "Improved Short Channel Effect Control in Bulk FinFETs With Vertical Implantation to Form Self-Aligned Halo and Punch-Through Stop Pocket," *IEEE Electron Device Letters,* vol. 36, pp. 648-650, 2015.

[4] H. Zhu, H. Zhong, T. Kawamura, Q. Liang, E. Leobandung, and S. Huang, "On the Control of Short-Channel Effect for MOSFETs With Reverse Halo Implantation," *IEEE Electron Device Letters,* vol. 28, pp. 168-170, 2007.

[5] C. Gupta, R. Goel, H. Agarwal, C. Hu, and Y. S. Chauhan, "BSIM-BULK: Accurate Compact Model for Analog and RF Circuit Design," in *2019 IEEE Custom Integrated Circuits Conference (CICC)*, 2019, pp. 1-8.

[6] J. Xue, Z. Ye, Y. Deng, H. Wang, L. Yang, and Z. Yu, "Layout-dependent STI stress analysis and stress-aware RF/analog circuit design optimization," in *2009 IEEE/ACM International Conference on Computer-Aided Design - Digest of Technical Papers*, 2009, pp. 521-528.

[7] L. J. Buturovic and L. T. Citkusev, "Back propagation and forward propagation," in *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, 1992, pp. 486-491 vol.4.

[8] C. Teo, K. L. Low, V. Narang, and A. V. Thean, "TCAD-Enabled Machine Learning Defect Prediction to Accelerate Advanced Semiconductor Device Failure Analysis," in *2019 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, 2019, pp. 1-4.

[9] C. D. Himmel and G. S. May, "Advantages of plasma etch modeling using neural networks over statistical techniques," *IEEE Transactions on Semiconductor Manufacturing,* vol. 6, pp. 103-111, 1993.

[10] A. Mostefai, s. Berrah, and A. Hamza, "Modeling and simulation of transistor mosfet (high-k) using neural network," *Journal of Electrical Engineering : Volume 17 / 2017 - Edition : 1,* vol. Volume 17, p. 74, 01/01 2017.

[11] G. L. Creech, B. J. Paul, C. D. Lesniak, T. J. Jenkins, and M. C. Calcatera, "Artificial neural networks for fast and accurate EM-CAD of microwave circuits," *IEEE Transactions on Microwave Theory and Techniques,* vol. 45, pp. 794-802, 1997.

[12] Q. Chen and G. Chen, "Artificial neural network compact model for TFTs," in *2016 7th International Conference on Computer Aided Design for Thin-Film Transistor Technologies (CAD-TFT)*, 2016, pp. 1-1.

[13] X. Gao, A. Huang, N. Trask, and S. Reza, "Physics-Informed Graph Neural Network for Circuit Compact Model Development," in *2020 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, 2020, pp. 359-362.

[14] A. Huang, Z. Zhong, Y. Guo, and W. Wu, "A dimension-reduced artificial neural network for the compact modeling of semiconductor devices," in *2018 IEEE MTT-S International Wireless Symposium (IWS)*, 2018, pp. 1-4.

[15] Y. Kim, S. Myung, J. Ryu, C. Jeong, and D. Kim, "Physics-augmented Neural Compact Model for Emerging Device Technologies," presented at the 2020 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD), 2020.

[16] H. Habal, D. Tsonev, and M. Schweikardt, "Compact Models for Initial MOSFET Sizing Based on Higher-order Artificial Neural Networks," in *2020 ACM/IEEE 2nd Workshop on Machine Learning for CAD (MLCAD)*, 2020, pp. 111-116.

[17] Y. Lei, X. Huo, and B. Yan, "Deep Neural Network for Device Modeling," in *2018 IEEE 2nd Electron Devices Technology and Manufacturing Conference (EDTM)*, 2018, pp. 154-156.

[18] L. Zhang and M. Chan, "Artificial neural network design for compact modeling of generic transistors," *Journal of Computational Electronics,* vol. 16, pp. 825-832, 2017/09/01 2017.

[19] 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, pp. 1318-1325, 2021.

[20] H. B. Hammouda, M. Mhiri, Z. Gafsi, and K. Besbes, "Neural-Based Models of Semiconductor Devices for SPICE Simulator," *American Journal of Applied Sciences,* vol. 5, pp. 385-391, 2008.

[21] P. B. L. Meijer, "Neural Network Applications in Device and Subcircuit Modelling for Circuit Simulation," PhD thesis, Department of Chemical Engineering and Chemistry, Technische Universiteit Eindhoven, 1996.

[22] K. Aadithya, P. Kuberry, B. Paskaleva, P. Bochev, K. Leeson, A. Mar*, et al.*, "Development, Demonstration, and Validation of Data-driven Compact Diode Models for Circuit Simulation and Analysis," Sandia National Lab. (SNL-NM), Albuquerque, NM (United States), United States SAND-2019-15303R 682080, 2019-12-01 2019.

[23] L. Lemaitre, G. Coram, C. McAndrew, and K. Kundert, "Extensions to Verilog-A to support compact device modeling," in *Proceedings of the 2003 IEEE International Workshop on Behavioral Modeling and Simulation*, 2003, pp. 134-138.

[24] I. Miller and T. Cassagnes, "Verilog-A and Verilog-AMS provides a new dimension in modeling and simulation," presented at the Proceedings of the 2000 Third IEEE International Caracas Conference on Devices, Circuits and Systems, 2000.

[25] S. Khandelwal, J. P. Duarte, A. S. Medury, S. Venugopalan, N. Paydavosi, D. D. Lu*, et al.*, "BSIM-CMG 110.0.0: Multi-gate MOSFET compact model: technical manual," 2014.

[26] B. Ding, H. Qian, and J. Zhou, "Activation functions and their characteristics in deep neural networks," in *2018 Chinese Control And Decision Conference (CCDC)*, 2018, pp. 1836-1841.

[27] G. V. Rossum, and F. L. Drake, "Python 3 Reference Manual.," 2009.

[28] A. A. M. Abadi, P. Barham, E. Brevdo, Z. Chen, C. Citro, et al, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015.

[29] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau*, et al.*, "Array programming with NumPy," *Nature,* vol. 585, pp. 357-362, 2020/09/01 2020.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel*, et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.,* vol. 12, pp. 2825–2830, 2011.

[31] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," *Computing in Science & Engineering,* vol. 9, pp. 90-95, 2007.

[32] E. Alcaide. (2018, January 01, 2018). E-swish: Adjusting Activations to Different Network Depths. arXiv:1801.07145. Available: https://ui.adsabs.harvard.edu/abs/2018arXiv180107145A

[33] C. C. McAndrew, G. J. Coram, K. K. Gullapalli, J. R. Jones, L. W. Nagel, A. S. Roy*, et al.*, "Best Practices for Compact Modeling in Verilog-A," *IEEE Journal of the Electron Devices Society,* vol. 3, pp. 383-396, 2015.

[34] G. J. Coram, "How to (and not to) write a compact model in Verilog-A," presented at the Proceedings of the 2004 IEEE International Behavioral Modeling and Simulation Conference, 2004.

[35] M. H. Perrott. (2011). *HSPICE Toolbox for Matlab and Octave*.

[36] A. Dasgupta and C. hu, "BSIM-CMG Compact Model for IC CAD: from FinFET to Gate-All-Around FET Technology," *Journal of Microelectronic Manufacturing,* vol. 3, p. 20030402, 12/01 2020.

[37] W. Fikry, "Accurate Analysis of the Number of Fins in the Triple-Gate FinFET Capacitance Compact Model," *International Journal of Engineering & Technology Sciences,* 01/01 2015.

[38] S. Rai, J. Sahu, W. Dattatray, R. A. Mishra, and S. Tiwari, "Modelling, Design, and Performance Comparison of Triple Gate Cylindrical and Partially Cylindrical FinFETs for Low-Power Applications," *ISRN Electronics,* vol. 2012, p. 827452, 2012/12/11 2012.

[39] V. Kol'dyaev, A. Clerix, R. M. Arteaga, and L. Deferm, "Characterisation of the Overlap Capacitance of Submicron LDD MOSFETs," in *ESSDERC '95: Proceedings of the 25th European Solid State Device Research Conference*, 1995, pp. 757-760.

[40] R. Singh, K. Aditya, A. Veloso, B. Parvais, and A. Dixit, "Experimental Evaluation of Self-Heating and Analog/RF FOM in GAA-Nanowire FETs," *IEEE Transactions on Electron Devices,* vol. 66, pp. 3279-3285, 2019.

[41] N. P. Maity, R. Maity, S. Maity, and S. Baishya, "Comparative analysis of the quantum FinFET and trigate FinFET based on modeling and simulation," *Journal of Computational Electronics,* vol. 18, pp. 492-499, 2019/06/01 2019.

[42] S. S. Rodriguez, J. C. Tinoco, A. G. Martinez-Lopez, J. Alvarado, and J. Raskin, "Parasitic Gate Capacitance Model for Triple-Gate FinFETs," *IEEE Transactions on Electron Devices,* vol. 60, pp. 3710-3717, 2013.

[43] D. Kim, Y. Kang, M. Ryu, and Y. Kim, "Simple and accurate capacitance modeling of 32nm multi-fin FinFET," in *2013 International SoC Design Conference (ISOCC)*, 2013, pp. 392-393.

[44] B. Carlile, G. Delamarter, P. Kinney, A. Marti, and B. Whitney, "Improving Deep Learning by Inverse Square Root Linear Units (ISRLUs)," presented at the ICLR 2018, 2017.