Low-Cost High-Accuracy Variation Characterization for Nanoscale IC Technologies via Novel Learning-based Techniques

Zhijian Pan[1], Miao Li[2], Jian Yao[2], Hong Lu[2], Zuochang Ye[1], Yanfeng Li[2], and Yan Wang[1]
[1]Institute of Microelectronics, Tsinghua University, Beijing 100084, China
[2]Platform Design Automation, Inc., Cameo Center, Beijing 100102, China
albert_li@platform-da.com, wangy46@tsinghua.edu.cn

***Abstract*—Faster and more accurate variation characterizations of semiconductor devices/circuits are in great demand as process technologies scale down to Fin-FET era. Traditional methods with intensive data testing are extremely costly. In this paper, we propose a novel learning-based high-accuracy data prediction framework inspired by learning methods from computer vision to efficiently characterize variabilities of device/circuit behaviors induced by manufacturing process variations. The key idea is to adaptively learn the underlying data pattern among data with variations from a small set of already obtained data and utilize it to accurately predict the unmeasured data with minimum physical measurement cost. To realize this idea, novel regression modeling techniques based on Gaussian process regression and partial least squares regression with feature extraction and matching are developed. We applied our approach to real-time variation characterization for transistors with multiple geometries from a foundry 28nm CMOS process. The results show that the framework achieves about 14x time speed-up with on average 0.1% error for variation data prediction and under 0.3% error for statistical extraction compared to traditional physical measurements, which demonstrates the efficacy of the framework for accurate and fast variation analysis and statistical modeling.**

I. Introduction

With continuously scaling of the feature size of integrated circuits (ICs), manufacturing process variations caused by fluctuations in device/process parameters become increasingly difficult to be controlled, especially at advanced nanoscale technology node. The severe variations introduce inevitable large-scale variability of circuit performance, thereby leading to great parametric yield loss [1].

A variety of techniques have been developed, such as Monte Carlo analysis and post-silicon tuning, to facilitate statistical IC analysis and optimization [2-3], so that variations can be estimated and minimized to ensure a robust design. The accuracy of underlying characterization for process variations greatly influence the effectiveness of these techniques.

Accurately characterizing and modeling process variation, however, is not a trivial task. A large amount of variation data is required to capture both the systematic and random variations existing in various levels (e.g. lot-to-lot and within-wafer), thus a large number of test structures need to be deployed, by which lots of properties need to be measured. The intensive testing is extremely time-consuming by conventional physical measurement, especially in Fin-FET era.

Recently, several techniques have been proposed to reduce silicon characterization cost. The virtual probe (VP) [4-5] and the work in [6-7] aim to reduce the number of measured dies needed to characterize spatial variation by using numerical algorithms such as discrete cosine transform, sparse regression and hidden Markov tree. Although they successfully reduce the cost of characterizing the spatial systematic device/circuit performance variation, the detailed measurement data for statistical modeling (e.g., a comprehensive set of transistor I-V curves) cannot be obtained by these methods.

For that, the work in [8-9] proposed a novel MOSFET parameter extraction method to extract an entire set of MOSFET model parameters using limited and incomplete I-V measurements based on Bayesian inference. However, the efficacy of the method in reducing the testing costs for statistical extraction relies heavily on the simplicity of the compact model (e.g., in the model used in [8], there are only six key parameters to be extracted). As described in [8], for a standard BSIM model, those few I-V tests are not sufficient to extract all parameters.

In this work, in order to enable the general device modeling or statistical modeling based on the standard models (e.g., BSIM) by using the incomplete measurements, we propose a novel learning-based framework to accurately recover the entire data set from incomplete test data, so as to minimize the testing cost. When the work in [4-7] focuses on reducing the number of measured dies, we try to reduce testing cost per die. The key idea is to adaptively learn the underlying data pattern from already obtained data which may come from either test-site measurement or simulations using mature or early product design kits, and utilize it in the unmeasured data prediction, so that the required amount of physical measurement can be dramatically reduced. In the proposed framework, by borrowing the basic ideas in the computer vision field [10], feature extraction and matching techniques are developed to facilitate high-quality sampling and subsequent data prediction. Novel regression modeling methods are proposed to implicitly learn the underlying data pattern and accurately predict the unmeasured data. While the proposed framework is independent of the type of devices and measurements, we mainly use the MOSFET and I-V measurements to validate the proposed approach.

The rest of this paper is organized as follows. Section II introduces the key idea of the framework. The details of the framework are described in Section III. The efficacy of the proposed framework is demonstrated by the experimental examples in Section IV. The final conclusion is made in Section V.

II. KEY IDEA

The key idea stems from the observation of the special property of variation testing, in which the devices measured on different dies are identical except for some process variations, without considering devices with defects, thus all the measured data can be interpreted as the combination of one specific data pattern describing the properties of the device and some variations. In other words, there may be redundant measurements for

978-3-9819263-0-9/DATE18/©2018 EDAA

repeatedly gathering the data describing basic device properties. For instance, Fig. 1(1) shows a set of industrial measured data of the identical 40nm transistors on 40 different dies. As shown, all the curves are highly overlapped after linear transformation (e.g., translation and scaling) on each of them, which demonstrates there are high degree of similarity among these curves.

Therefore, we proposed our framework, aiming to reduce the redundant testing cost by adaptively learning and utilizing underlying data patterns. Fig. 1(2) shows the basic process. Firstly, several dies are physically measured to obtain the detailed data (e.g., I-V curve set) containing the needed data pattern. For the new dies under test, only a very small set of sample data is measured to capture the variation. Finally, by combining the obtained data, all the unknown data is predicted.

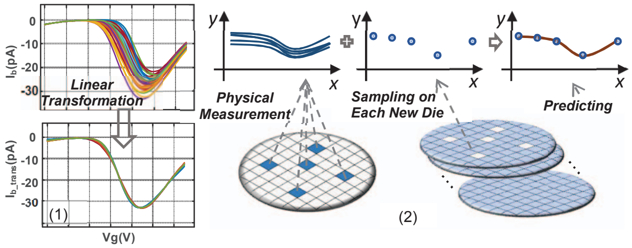

Fig. 1. (1) I-V testing data (2) Basic process of the proposed framework.

Note that for the devices with defects, as will be discussed in Section III-D, they can be detected by strict verification process and remeasured by traditional physical measurements.

III. Proposed Framework

An overview of the proposed framework is shown in Fig. 2. It mainly consists of two stages: data preparation and training/prediction. In the former stage, the core task is to generate high-quality sample data by analyzing the already measured data. In the latter stage, the key task is to build accurate regression models. For that, feature extraction and matching techniques and novel regression modeling techniques are developed.

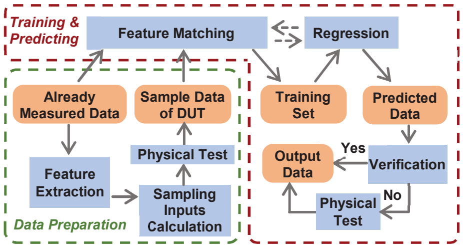

Fig. 2. The flow of the proposed framework.

Since most property testing is conducted by sweeping the inputs (e.g., voltages) and the data can be transferred to the form of curves, the data in this paper is represented as curve data.

A. Feature Extraction and Sampling

The sample points are of great importance for data prediction. The ideal samples are the points that carry all the critical information by which the entire data set can be predicted when the data pattern is known. By analogy with problems in computer vision field, such points can be treated as the features. By extracting and analyzing the feature points of already measured curves, the best feature locations (measurement inputs) of unknown curves are estimated.

1) Feature Extraction

Similar to the usual features (e.g., corners) in the images, the special points (e.g. maxima) are good choices of curve features. Therefore, the initial step of the feature extraction is to adaptively find the predefined special points on a curve.

Suppose $y = g(x)$ is the curve describing the mapping relationship between the measurement output y and the input x, and the inputs swept from a to b constitute X:

$$X = \{x_i | x_i \epsilon [a, b], i = 1, 2, \dots, n\} \quad (1)$$

where n is the number of points tested on the curve. For a measured curve, the following points are selected as the special points and extracted as features:

$$S1 = \{(a, g(a)), (b, g(b))\}$$

$$S2 = \{(x^*, g(x^*)) \mid g(x^*)\ is\ the\ extreme\ value\}$$

$$S3 = \{(x^*, g(x^*)) \mid g'(x^*)\ is\ the\ extreme\ value\}$$

$$S4 = \{(x^*, g(x^*)) \mid c(x^*)\ is\ the\ maximum\ value\}$$

$$c(x) = \ |g''(x)|/(1 + g'(x)^2)^{1.5} \quad (2)$$

where $x^* \in X$. Among these point sets, $S1$ and $S2$ are sets of endpoints and extreme points, which are obviously special on a curve. $S3$ contains the points where y changes faster or slower in the neighborhood. In $S4$, $c(x)$ is the curvature of a curve, which measures how fast a curve is changing direction, thus $S4$ contains points where the curve turns faster in the neighborhood. The union of these point sets constructs the initial feature set.

In most cases, the first and second derivatives used above exist and can be estimated by the numerical differentiation, since nearly all measured curves have physical meanings. In rare cases where the derivatives are not applicable, the union of $S1$ and $S2$ can be extracted as the initial feature set.

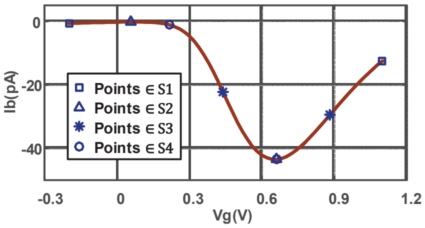

Fig. 3. Examples of feature points.

After the initial feature set is obtained, it is further improved by several modifications, including removing the redundant feature points and supplementing additional ones. Suppose the curve length between two adjacent feature points is l_p and the entire curve length is l_c, the ratio $r_p = l_p/l_c$ controls the remove and addition by comparison with the lower and upper threshold parameters r_L, r_U: 1) for any $r_p < r_L$, the corresponding two feature points are merged to one (the middle point); 2) for any $r_p > r_U$, an extra data point in the middle of them is added into the feature set. The r_L and r_U influence the number of final feature points. The optimal values can be determined by the parameter setting method described in Section III-D.

The feature points extracted by the feature extraction method for one physically measured 40nm transistor I-V curve are plotted as an example in Fig. 3.

2) Sampling on the Device Under Test (DUT)

As discussed in Section II, high similarity exists among different curves during variation testing, thus it is reasonable to assume that the feature points on unmeasured curves are close to those on the measured ones. Therefore, by calculating the means of corresponding test inputs of the extracted feature points on already measured curves, the test inputs for sampling on the unknown curves can be estimated as:

$$X_s = \left\{x_{s_i} \middle| x_{s_i} = arg\min_{x_k} \left| x_k - \frac{1}{N}\sum_{j=1}^{N} x_{s_i}^{(j)} \right|, i = 1,2, \dots m;\ x_k \in X\right\} \quad (3)$$

where x_{s_i} is the i-th element in X_s, $x_{s_i}^{(j)}$ is the test input of the i-th feature point on the j-th measured curve, and N and m are the number of measured curves and extracted feature points.

For the unmeasured devices, suppose the mapping relationship between the measurement output t and the input x is $t = h(x)$. After testing the device according to the sampling inputs in X_s, the sampling output set T_s is then obtained as

$$T_s = \{t_{s_i} | t_{s_i} = h(x_{s_i}), x_{s_i} \epsilon X_s\}. \quad (4)$$

B. Regression Modeling

One key task in the proposed framework is to accurately predict the unknown data based on the obtained sample points and it can be formulated as a regression problem to minimize $\sum_{i=1}^{n}(\hat{t}_i - t_i)^2$, where $\hat{t}_i$ and t_i are the predicted and real output at the input x_i. Traditional regression methods solve this problem by building a hypothesis function $\hat{t}_i = \hat{h}(x_i)$ using the sample points (x_{s_i}, t_{s_i}) to model $t_i = h(x_i)$. However, by these methods, the similarity between curves discussed in Section II cannot be utilized and lots of sample points are required to ensure high-accuracy prediction.

For that, novel regression modeling techniques are proposed, which is a key contribution of this work. In contrast to modeling the relationship between x_i and t_i, the proposed methods predict the output t_i directly through the sample outputs T_s in (4) and the outputs of already measured curves denoted as

$$Y = \left\{y_i^{(j)} \middle| y_i^{(j)} = g^{(j)}(x_i), i = 1,2, \dots n;\ j = 1,2, \dots N\right\} \quad (5)$$

where $g^{(j)}(x)$ is the j-th measured curve and n is the number of points tested on the curves. Two different strategies for modeling are proposed and will be described as follows.

1) Vertical Regression Model (VRM)

Intuitively, since the measured N curves and the unmeasured one have the same underlying pattern, the outputs $y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(N)}$ and t_i at the same input x_i will be very similar. To utilize this property, the first modeling form is proposed as

$$\hat{t}_i = f_{VR}\left(y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(N)}\right), i = 1,2, \dots n. \quad (6)$$

To build this model, from the point view of machine learning, at the training stage, the design matrix Y_{dv} (model inputs) and the response vector T_{rv} (model outputs) are constructed as

$$Y_{dv} = \begin{bmatrix} y_{s_1}^{(1)} & y_{s_1}^{(2)} & \cdots & y_{s_1}^{(N)} \\ y_{s_2}^{(1)} & y_{s_2}^{(2)} & \cdots & y_{s_2}^{(N)} \\ \vdots & \vdots & \vdots & \vdots \\ y_{s_m}^{(1)} & y_{s_m}^{(2)} & \cdots & y_{s_m}^{(N)} \end{bmatrix}, T_{rv} = \begin{bmatrix} t_{s_1} \\ t_{s_2} \\ \vdots \\ t_{s_m} \end{bmatrix} \quad (7)$$

where the outputs of N measured curves at one sampling input x_{s_i} in (3) constitute one row of Y_{dv}, and the corresponding sampling output of the curve to be predicted constitutes one row of T_{rv}. When the model is obtained, the unknown curve data can be predicted by the corresponding data on the measured curves.

For better understanding, Fig. 4(1) shows a simple example of curve prediction problem, in which the data assignment for training and prediction is indicated. As shown, the unknown data on $h(x)$ is predicted by the vertical corresponding points.

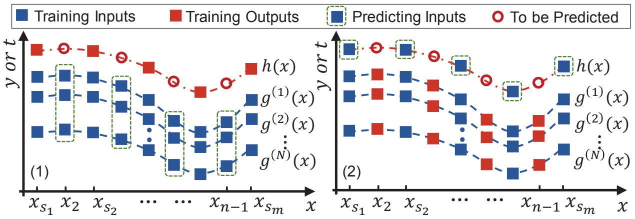

Fig. 4. Data assignment for training and predicting in (1) VRM and (2) HRM.

A variant of Gaussian process regression (GPR) method, the blind GPR [12], is adopted to train this regression model. In contrast to the standard GPR, where the mean of the GP is assumed as zero, in blind GPR, the mean is modeled by linear combination of the given basis functions and the best combination would be efficiently determined by Bayesian method. By this technique, the mean can capture the general trend of the data and the standard GPR can predict the residuals, so as to improve the prediction accuracy. This method is very suitable for our problem, since the variations can be approximately interpreted as the combination of the systematic variation (general trend) and random variation (the residual). Besides, the assumption of Gaussian process is reasonable for variations.

2) Horizontal Regression Model (HRM)

From another perspective on the sample points and curves, intuitively, if the curve pattern has been learned from the already measured curves and the critical points (sample points) on the unknown curve are given, it is not hard to predict the entire curve. Based on this idea, the second modeling form is proposed as

$$\hat{t}_{l_1}, \hat{t}_{l_2}, \dots \hat{t}_{l_{n-m}} = f_{HR}\left(t_{s_1}, t_{s_2}, \dots, t_{s_m}\right) \quad (8)$$

where $\hat{t}_{l_i}$ is the estimation of the test output at the input that does not belong to the sampling input set, namely $\hat{t}_{l_i}$ is the estimation of $t_{l_i} = h(x_{l_i})$, where $x_{l_i} \in X$ and $x_{l_i} \notin X_s, i = 1,2, \dots n - m$.

As shown in (8), this is a multivariate regression modeling problem. At the training stage, the design matrix Y_{dh} and the corresponding response vector T_{rh} are constructed as:

$$Y_{dh} = \begin{bmatrix} y_{s_1}^{(1)} & y_{s_2}^{(1)} & \cdots & y_{s_m}^{(1)} \\ y_{s_1}^{(2)} & y_{s_2}^{(2)} & \cdots & y_{s_m}^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ y_{s_1}^{(N)} & y_{s_2}^{(N)} & \cdots & y_{s_m}^{(N)} \end{bmatrix}, T_{rh} = \begin{bmatrix} y_{l_1}^{(1)} & y_{l_2}^{(1)} & \cdots & y_{l_{n-m}}^{(1)} \\ y_{l_1}^{(2)} & y_{l_2}^{(2)} & \cdots & y_{l_{n-m}}^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ y_{l_1}^{(N)} & y_{l_2}^{(N)} & \cdots & y_{l_{n-m}}^{(N)} \end{bmatrix} \quad (9)$$

where $y_{l_i}^{(j)} = g^{(j)}(x_{l_i}), j = 1,2, \dots N, x_{l_i} \in X$ and $x_{l_i} \notin X_s, i = 1,2, \dots n - m$. After this model is built, by using the sample points on an unknown curve, the entire curve can be predicted. The data assignment for training and predicting is depicted in Fig. 4(2) for the same curve prediction problem shown in Fig. 4(1). In contrast to the vertical modeling, the unknown data on $h(x)$ is predicted by the horizontal sample points.

The columns of both Y_{dh} and T_{rh} are different points on the same curve, thus they are probably correlated with each other. The partial least squares regression (PLSR) method [13] is used to solve this kind of multivariate regression problems. PLSR is a technique that combines the ordinary least squares regression with the idea of principal components analysis and canonical correlation analysis. It is designed to confront the situation that there are a set of dependent variables to be predicted from many probably correlated independent variables. Therefore, PLSR is very suitable for solving our problem.

3) Summary

In summary, both VRM and HRM are capable of learning the underlying data pattern to predict the unknown data. They are in different ways and complementary to each other.

As shown in (7), in VRM training, the number of measured curves determines the dimension of the regression problem, and the number of samples determines the size of the training data set. Commonly, to solve a machine learning problem, lower dimension and larger training data set are more desirable. Therefore, VRM is more suitable for the situation that the complexity of the curve is high so that more sample points are needed, whereas the variations are relatively low so that a smaller number of measured curves are required. For HRM, as shown in (9), the Y_{dh} is the transpose of the counterpart Y_{dv} in VRM, thus the situation is just the opposite. Therefore, HRM is preferred when the curves are relatively simple and the variations are relatively high.

Since each method owns its advantages and can better deal with the problems in different situations, in our framework, they are combined so as to cover most possible situations. For an unknown curve prediction task, both models are built and two different candidate results are generated. The better one is then adaptively selected by the verification process (Section III-D).

C. Feature Matching

In the regression methods discussed above, the data points with the same measurement inputs are regarded as the corresponding points for model training and data prediction. However, because of variations, these points may not be perfectly matched. It is much better when they are matched with each other. For instance, intuitively, the better choices of data used to predict the minimum of an unknown curve are those that are also minimums of the known curves. From the perspective of our regression modeling methods, better matching will lead to lower problem dimension, and hence lower training data size and higher generalization ability [14].

Motivated by locating the best matched data points on different curves so as to assist the regression modeling, we borrow the idea of cross-covariance to develop the feature matching method. Cross-covariance is used to measure the similarity between one sequence and shifted copies of another sequence [15]. In our problems, the outputs on the curves can be regarded as sequences and similarity can be used to evaluate the degree of matching. At the beginning, however, for the unknown curves, only the sampling feature points are known, thus a new sequence with the same length of the curve is created. In the sequence the feature points are at their corresponding positions and other positions are set to zeros to eliminate their influences.

Suppose the elements from the *j*-th measured curve in (5) constitute the sequence $P=[y_1^{(j)},y_2^{(j)},\dots,y_n^{(j)}]$, and the elements in (4) constitutes the sequence $Q=[q_1,q_2,\dots q_n]$, where $q_i=t_{s_i}$ when $i=s_i$ and $q_i=0$ when $i\neq s_i$. Then the cross-covariance is calculated as:

$$c_{hq}(d)=\begin{cases}\sum_{k=1}^{n-d}\left(P(k+d)-\frac{1}{n}\sum_{i=1}^{n}P(i)\right)\left(Q(k)-\frac{1}{n}\sum_{i=1}^{n}Q(i)\right), & d\geq 0\\ c_{hq}(-d), & d<0\end{cases}\qquad(10)$$

where d is the number of shift elements. After c_{hq} is calculated, the best shifted value for matching can be determined as:

$$\Delta d=\arg\max_{|d|<n} c_{hq}(d)\qquad(11)$$

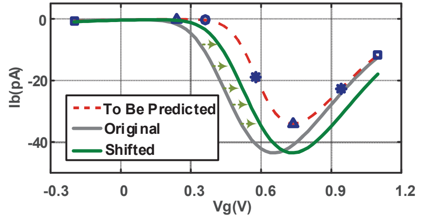

Fig. 5. Example of feature matching.

The new shifted sequence is then obtained as $P_{st}(i)=P(i+\Delta d),\ 0\leq i+\Delta d\leq n$, which will be used to construct the matrices for regression and prediction. As $i+\Delta d$ might overflow when $\Delta d\neq 0$, the length of P_{st} would be shorter than n. Since the shifts are usually not large, this problem can be easily solved by the traditional extrapolation methods [16]. An example of feature matching of real measured I-V curves from 40nm transistors is plotted in Fig. 5. As shown, the shifted curve exhibit much better matching to the curve to be predicted.

D. Verification and Framework Parameter Setting

The verification process aims to check the quality of the data, for both the measured and predicted data, and it is used for the following purposes: 1) to detect and screen the data measured from devices with defects; 2) to select the best predicted data generated by multiple models and algorithms; 3) to judge whether to use the predicted data or to remeasure all of the data.

The verification methods can be classified into two categories: self-check and cross-check. The former one is to check whether the curve traits (e.g., the range of the data value and the continuity of both the original curves and the first derivatives) are normal. The cross-check is to check the similarity between the curve and the already known curves. The similarity is evaluated by Pearson correlation coefficient [11] and the criterion of similarity is adaptively set according to the initial measured data. By these strict checks, the reliability and accuracy of the outputs can be greatly guaranteed.

To set the framework parameter such as the lower and upper thresholds in the feature extraction, the leave-one-out cross-validation [14] is adopted. Suppose there are N measured curves. We assume one of them is unknown and to be predicted by the other $N-1$ ones. The predicted result is evaluated by the relative errors between the measured curve and the predicted one.

E. Summary

Algorithm 1 summarizes the major steps of the framework:

1. Start from N already measured curves.
2. Extract the feature points and calculate the sampling input set by the methods described in Section III-A.
3. For an unmeasured device to be tested, measure it according to X_s in (3) to obtain the sampling outputs T_s in (4).
4. By the feature matching methods (10) (11), find the data on each measured curve that best matches the unmeasured data.
5. Formulate the modeling problem and construct the matrices based on both forms of VRM (6) (7) and HRM (8) (9).
6. Build the VR and HR models by GPR and PLSR methods.
7. Predict the entire curve by the built models.
8. Evaluate the predicted curve data and select the best one as the output based on the verification in Section III-D. If no predicted result is selected, the curve is physically measured.
9. Stop when all devices are tested, otherwise, go to Step 3.

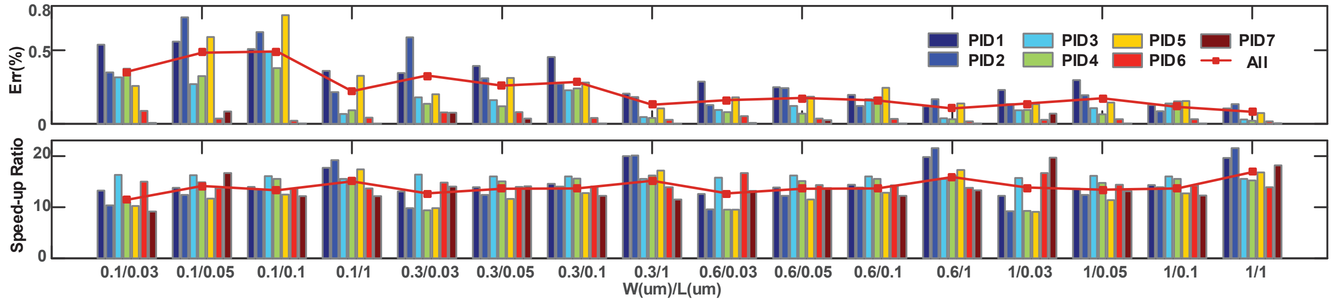

Fig. 6. Details of the experiment results: (1) the relative error, (2) the speed-up ratio.

IV. Experimental Examples

In this section, we demonstrate the efficacy of the proposed framework by applying it to the 28nm CMOS transistor variation testing. To evaluate the generalization ability, all significant I-V properties of transistors with 16 different sizes on 300 dies will be tested. All the transistors are firstly tested by the traditional physical measurements and then they are re-tested based on our proposed framework.

A. Configurations

For a 28nm transistor, the standard test requirements are summarized in Table 1, where each row corresponds to one property for which several curves will be tested.

Table 1. Test Requirements

PID	V_s(V)	V_g(V)	V_d(V)	V_b(V)	Test
1	0	-0.2:0.01:0.9	0.05	0:-0.18:-0.9	I_d
2	0	-0.2:0.01:0.9	0.9	0:-0.18:-0.9	I_d
3	0	0.18:0.18:0.9	0:0.01:0.9	0	I_d
4	0	0.18:0.18:0.9	0:0.01:0.9	-1.05	I_d
5	0	-0.2:0.01:0.9	0.3:0.3:0.9	0	I_d
6	0	-0.2:0.01:0.9	0:0.45:0.9	0	I_g
7	0	-0.2:0.01:0.9	0:0.45:0.9	0	I_b

In the traditional physical measurements, all the required data points will be tested. To the best of our knowledge, in the middle of physical testing, there are no methods to adaptively adjust the measurement steps. Therefore, by the traditional physical testing, there will be totally 3241 data points on 31 I-V curves to be tested for one transistor on one die.

The numerical experiments are performed on a computer with 2.4GHz CPU and 8 GB memory.

B. Experimental Results of Variation Data Prediction

After the tests are completed, both the accuracy and speed-up performance of the propose framework are evaluated. To assess the accuracy, the relative error of a curve is given by:

$$Error = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{t}_i - t_i}{t_i}\right)^2} \quad (12)$$

where t_i and $\hat{t}_i$ are the measured value by the physical testing and the predicted value by our framework at the same measurement input x_i and n is the number of points on the curve. Commonly, when the measurement value (current) is close to zero, only their orders of magnitude are concerned, thus when t_i<1e-13, the original values of t_i and $\hat{t}_i$are substituted by their logarithm values to avoid generating unreasonably high errors.

Based on equation (12), the relative errors of all I-V curves are calculated. There are 148.8k curves in total and the average relative error is 0.11%. All the errors are summarized in the histogram shown in Fig. 7. As shown, most curves exhibit relative errors less than 0.5%.

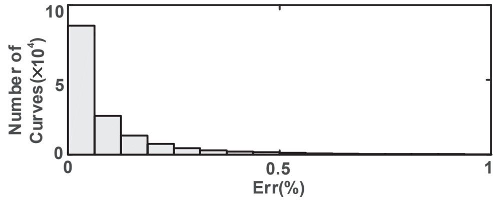

Fig. 7. Histogram of relative error of all measured curves.

To analyze the speed-up performance, the average number of tested samples and average testing time for each property of a single transistor are calculated and summarized in Table 2. As shown, benefited from the reduced number of sampling (real tested points), our framework achieves about 14x time speed-up compared to the traditional physical measurements. In the proposed framework, since the sampling numbers for different curves are adaptively adjusted, the average value is not an integer. The computational time is also counted in calculating the overall testing time, but compared to the entire spent time, it only takes a very small portion (about 3% on average).

Table 2. Testing Time Comparison

PID	Proposed Framework		Traditional Phy. Meas.		Ratio
	Samp.#	Time(ms)	Samp.#	Time(ms)	
1	68.4	460.9	666	6900.3	14.97
2	62.1	450.3	666	6262.6	13.91
3	26.9	142.7	455	2261	15.84
4	28.6	234.7	455	3339.6	14.23
5	34.3	229.7	333	2924.9	12.73
6	22.5	463.1	333	6660	14.38
7	24.9	505.2	333	6660	13.18
Tot.	267.7	2486.5	3241	35008.5	14.08

To evaluate the generalization ability, the relative error and the speed-up performance of the data from each property test of transistors with different sizes are calculated and summarized in Fig. 6. As shown, the performances vary with the PID and transistor size, since the data patterns of different properties are different and the variations usually have different influences on transistors with different sizes.

However, as shown, the maximum error is below 0.8% and at least 8x speed-up can be achieved. Therefore, the proposed framework can deal with variation testing with various data patterns and different degrees of variations.

C. Experimental Results of Statistical Extraction

To illustrate the efficacy of the data obtained by the proposed framework for variation analysis and statistical modeling, several key parameters describing both DC and small-signal properties of a transistor are extracted and compared to those extracted from the physically measured data. The relative error defined by the equation similar to (12) is calculated for all transistors on all dies and the results are summarized in Table 3.

As shown, the DC parameters V_{th}, I_{dlin} and I_{dsat} show quite small errors, and the errors of small-signal parameters g_m and g_{ds} are higher, but they are still very small (below 0.3%).

Table 3. The Extracted Key Transistor Parameters

Para.	V_{th}	I_{dlin}(V_d=0.05V, V_g=0.9V)	I_{dsat}(V_d=0.9V, V_g=0.9V)	g_m(V_d=0.9V, V_g=0.54V)	r_{ds}(V_d=0.9V, V_g=0.54V)
Error	0.02%	3e-3%	7e-4%	0.17%	0.26%

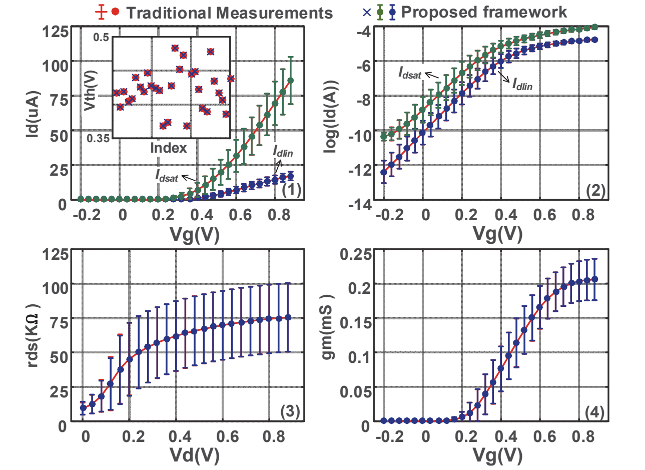

Fig. 8. Mean and ±2σ(standard deviation) of (1) I_{dlin} and I_{dsat}, (2) log(I_{dlin}) and log(I_{dsat}), (3) r_{ds}, and (4) g_m at various voltage biases obtained from both the proposed framework and traditional measurements. The inset in (1) shows V_{th} of several dies extracted from I_{dlin}.

To exhibit more details of the key parameters, the results of the smallest transistor in the experiment are plotted in Fig. 8. As shown, the key parameters obtained from the proposed framework agree well with those from the traditional measurements.

D. Comparison and Discussion

To compare the proposed framework with traditional data prediction methods, we apply the traditional methods, including both widely used interpolation methods (e.g. spline interpolation [16]) and regression methods (e.g. LASSO and GPR) to the same variation testing experiments in Section IV-B. For these methods, the sample points on the unknown curve are the same with those used in the proposed framework. The relative errors versus devices with different sizes are depicted in Fig. 9(1).

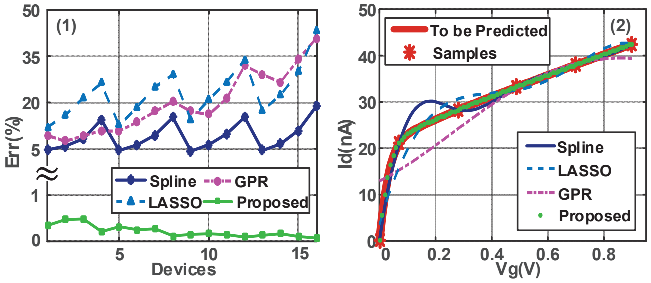

Fig. 9. (1) Relative error and (2) predicted curves of the traditional methods and the proposed framework.

As shown, the proposed framework achieves much higher prediction accuracy than the traditional methods. The main reason is that the already measured curves cannot be utilized by the traditional methods as described in Section III-B. For better explanation, the results predicted by different methods for one specific curve (one I_d-V_g curves from PID3) in the experiment are plotted in Fig. 9(2). This curve is representative of one significant type of I-V curves in transistor testing. As shown, there are six sample points on the curve and it is not trivial to accurately predict other points with such small set of samples. The spline interpolation method failed due to the Runge's phenomenon. Both LASSO and GP regression methods suffer from under-fitting problem [14]. Therefore, six sample points are apparently not enough for these traditional methods.

For one kind of curves, if possible, the prediction accuracy of the traditional methods can be improved by careful tuning. However, a tuned model would have generalization problem. Besides, it usually takes a long time to tune a model. On the contrary, since the proposed framework can learn from similar curves and adaptively adjust the sample points, it possesses high accuracy and generalization capability as shown in Fig. 6.

V. Conclusion

In this work, a novel learning-based high-accuracy data prediction framework is proposed to efficiently reduce the cost of process variation characterization in modern nanoscale IC technologies. The framework aims to adaptively learn the underlying pattern existing among general variation characterization data to accurately predict the unknown data with minimum physical measurement cost. Novel regression modeling techniques with feature extraction and matching are proposed. The approach is applied to the variation testing for multiple 28nm transistors to demonstrate its efficacy. The results exhibit that compared to the physical measurement and other traditional data prediction methods, the proposed framework achieves about 14x time speed-up with nearly no accuracy loss (0.1% average error) for variation data prediction and quite small relative errors (under 0.3%) for statistical extraction.

References

[1] Semiconductor Industry Associate, International Technology Roadmap for Semiconductors, 2013.

[2] Q. Liu et al, "Synthesizing a representative critical path for post-silicon delay prediction," *ISPD* May 2009.

[3] K. Heloue and F. Najm, "Statistical timing analysis with two-sided constraints," in *IEEE ICCAD*, pp. 829–836, 2005.

[4] W. Zhang et al, "Bayesian virtual probe: Minimizing variation characterization cost for nanoscale IC technologies via Bayesian inference," *IEEE DAC*, pp. 262–267, Jun. 2010.

[5] H. Goncalves et al., "A fast spatial variation modeling algorithm for efficient test cost reduction of analog/RF circuits," *IEEE DATE*, pp. 1042–1047, Mar. 2015

[6] N. Kupp, et al, "Spatial correlation modeling for probe test cost reduction in RF devices," *ICCAD*, pp. 23–29, 2012.

[7] K. Huang, et al, "On combining alternate test with spatial correlation modeling in RF ICs," *ETS*., pp. 64–69, 2013.

[8] L. Yu et al., "Remembrance of transistors past: Compact model parameter extraction using Bayesian inference and incomplete new measurements," *DAC*, pp. 1-6, 2014.

[9] L. Yu et al., "Compact model parameter extraction using Bayesian inference, incomplete new measurements, and optimal bias selection," *IEEE TCAD*, vol. 35, no. 7, pp. 1138–1150, Jul. 2016.

[10] Lowe, D. G. "Distinctive image features from scale-invariant key-points." *International Journal of Computer Vision*, pp. 91-110, 2004.

[11] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge Press, 2007.

[12] I. Couckuyt et al, "Blind kriging: Implementation and performance analysis," *AES*., vol. 49, pp. 1–13, Jul. 2012.

[13] Rosipal, R. et al. "Overview and Recent Advances in Partial Least Squares." Berlin, Germany: Springer-Verlag, 2006.

[14] C. Bishop, Pattern Recognition & Machine Learning, Prentice Hall, 2007.

[15] Larsen. "Correlation Functions and Power Spectra." November, 2009.

[16] W. Press, et al, Numerical Recipes: *The Art of Scientific Computing*, Cambridge University Press, 2007