Resistive Open Defect Classification of Embedded Cells under Variations

Najafi-Haghi, Zahra Paria; Wunderlich, Hans-Joachim

Proceedings of the IEEE Latin-American Test Symposium (LATS'21), Virtual, 27 - 29 October 2021, pp. 1-6

doi: https://doi.org/10.1109/LATS53581.2021.9651857

Abstract: Small Delay Faults (SDFs) due to defects and marginalities have to be distinguished from extra delays due to process variations, since they may form a reliability threat even if the resulting timing is in the specification. In this paper, it is shown that these faults can still be identified, even if the corresponding defect cell is deeply embedded into a combinational circuit and its behavioral features are affected by several masking impacts of the rest of the circuit. The results of a few delay tests at different voltages and frequencies serve as the input to machine learning procedures which can classify a circuit as marginal due to defects or just slow due to variations. Several machine learning techniques are investigated and compared with respect to accuracy, precision, and recall for different circuit sizes and defect scales. The classification strategies are powerful enough to sort out defect devices without a major impact on yield.

Preprint

Resistive Open Defect Classification of Embedded Cells under Variations

Zahra Paria Najafi-Haghi, Hans-Joachim Wunderlich
Institute of Computer Architecture and Computer Engineering, University of Stuttgart, Germany
Email: {najafi-haghi, wu}@informatik.uni-stuttgart.de

***Abstract*—Small Delay Faults (SDFs) due to defects and marginalities have to be distinguished from extra delays due to process variations, since they may form a reliability threat even if the resulting timing is in the specification. In this paper, it is shown that these faults can still be identified, even if the corresponding defect cell is deeply embedded into a combinational circuit and its behavioral features are affected by several masking impacts of the rest of the circuit.**

The results of a few delay tests at different voltages and frequencies serve as the input to machine learning procedures which can classify a circuit as marginal due to defects or just slow due to variations. Several machine learning techniques are investigated and compared with respect to accuracy, precision, and recall for different circuit sizes and defect scales. The classification strategies are powerful enough to sort out defect devices without a major impact on yield.

***Index Terms*—Small delay faults, variations, reliability, defect modeling, statistical learning.**

I. Introduction

Resistive opens and resistive bridges result most often in Small Delay Faults (SDFs) [1] [2] [3] which are hard to detect during production testing. Since they may change rather early in the circuit's lifetime and turn into a catastrophic fault, they have to be covered during the test of high-quality systems [4] [5] [6]. It is well known that testing at varying voltages and especially low voltage testing increase the fault coverage significantly [7] [8] [9] [10], and modern systems with Adaptive Voltage Frequency Scaling (AVFS) have all the means to support this test strategy [11]. However, technology scaling comes with the additional difficulty that circuits are subject to process variations which similarly affect the timing as SDFs do. In both cases, the circuit behavior may be slowed down but stays still within the specification. Yet, a slow circuit due to variations may be safe whereas a slow circuit due to resistive defects may form a reliability threat. Distinguishing cells slow due to variations, from defect cells has been the subject of ongoing research [12], [13], [10], and [14]. A severe limitation of the previous works comes with the fact, that the defect cells are subject to variations too. In some cases, a defect fast cell may still be faster than a defect-free one.

In a recent paper, it was shown that cells from the Open Cell Library (OCL) [15] can be classified with high accuracy by statistical learning techniques, even if both defect and defect-free cells are subject to variations [16]. This knowledge was derived from extensive Monte Carlo Spice simulation [17] which created data sets for defect and defect-free cells to be used as input to supervised learning. While this classification is encouraging, it is not immediately applicable to the production testing of a combinational circuit, in which cells are embedded and not directly controllable and observable. In addition, single defect classification does not cover the challenges due to the masking impact of all the other cells in the sensitized path. The paper at hand investigates the possibility to classify a cell even if it is deeply embedded into a combinational circuit as in Fig. 1, with up to 16 levels of cells in the sensitized path from the embedded cell to the primary output, which exceeds the range of usual propagation paths.

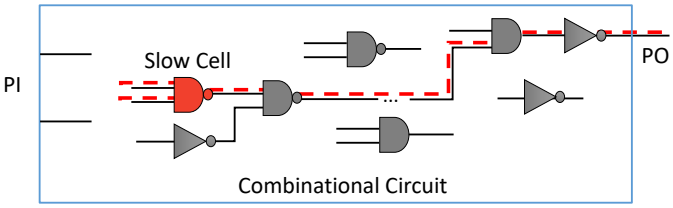

Fig. 1. An embedded slow cell in a combinational circuit

Based on extensive Monte Carlo Spice simulation, it is shown, that a slow cell embedded into a combinational circuit under variations changes the behavior of the primary outputs sufficiently to classify the root cause of the delay into variations or resistive defects. The performance of several statistical learning techniques is investigated, which learn from the generated data set to classify defect and defect-free circuits under variations. A detailed electronic analysis for different behaviors of defect and defect-free models under variations is performed, which is as far as we know the first time that defect classification problem is analyzed besides using statistical learning techniques. The evaluation takes into account how deeply the gate is embedded, the fault sizes, and different classification metrics such as accuracy and F1-score. The evaluation results are extremely encouraging, for the feasibility of exploiting this statistical learning-based technique in production test of multi-gate bigger circuits or in CAD-tools in presence of masking impact, with more efficient multi-level simulators as already presented in [18] [19].

The rest of the paper is organized as follows. The next section gives some orientation of the research background for defect detection under variations as well as specific challenges in combinational circuit behavior analysis. Supporting electrical fundamentals of the proposed strategy are extracted and

978-1-6654-2057-0/21/$31.00

presented in Section III. Section IV describes in detail the data generation process by Monte Carlo Spice simulation. In Section V, the statistical learning techniques for classification are introduced and Section VI evaluates the performance of the classification with respect to the size of the circuit as well as the size of the marginal defects.

II. Research Background

A. State of the Art

Many works have studied the challenges to test a circuit under varying operating conditions, e.g., [20] [21] [22]. Already in the 90s, [7] [8] have intentionally used the varying voltage by applying Very-Low-Voltage test to detect defects causing possible Early Life Failure (ELF). [23] also presented a strategy to make use of dynamic voltage scaling for overall fault coverage improvement.

Resistive opens, resistive bridges, and gate-oxide pinholes are considered as defect mechanisms which may appear as SDFs [16]. The behavior induced by resistive defects has been analyzed in [24], and [14] investigates the detection of the resistive defects by using delay tests.

Defect detection under variations has been taken into consideration as a *variation-aware test* [25]. Publications [10] and [14] propose methods to distinguish process variations from marginal defects by using the different delay behavior under varying voltages. However, their defect model adds a fixed amount of delay and does not consider process variation simultaneously. [16] used statistical learning schemes for differentiating process variations and marginal defects, by some experiments applied to an isolated single cell.

B. Combinational circuit behavior under variations and defects

If a defect cell is embedded into a combinational circuit (Fig. 1), its timing behavior has to be propagated to a primary output for classification. As seen above, rather small changes of the timing of the defect cell have to be propagated, and the mechanisms already known from glitch analysis or soft error analysis have to be considered. This section summarises the main challenges of data collection from the output of a combinational circuit, which impact the defect classification of an embedded cell.

- **Logical masking**: To detect a delay, a propagation path has to be sensitized. The discussion below assumes that only appropriate test patterns are used. ATPG is not the subject of this paper.

- **Electrical masking**: CMOS is a self-restoring technology that filters short pulses and reshapes the slopes of transitions. Therefore, not only the cell under investigation but also the entire propagation path has to be subject to analog simulation. It will be shown that electrical masking is still sufficiently transparent to let timing information through for classification.

- **Timing masking**: All the cells on the propagation path suffer from variations, and in general the propagation time of a path has to be modeled by a skewed multi-variable distribution [26] [27]. Since this distribution does not affect the output shape with respect to different voltages, it is assumed below that the path propagation delay follows a Gaussian distribution.

III. Electronic Analysis of Delay Faults under Variations

Repeating from textbooks, e.g., [28], the simulating delay τ of a CMOS transistor can be roughly expressed by

$$\tau = \frac{C_L V_{dd} L T_{ox}}{\mu \epsilon_{ox} W (V_{dd} - V_t)^{\alpha}} \quad (1)$$

Here τ denotes the delay, C_L is the capacitance at the gate output, V_{dd} is the supply voltage, V_t is the transistor threshold voltage, α is the Sakurai's index which can be taken equal to 1 in scaled technologies, L is the transistor channel length, W is the transistor channel width, μ is the carrier's mobility, t_{ox} is the gate oxide thickness, and ϵ_{ox} is the gate oxide permittivity. The simulated delay versus V_{dd} for the original defect-free NAND with nominal process parameters can be seen as the black curve in the plot of Fig. 2.

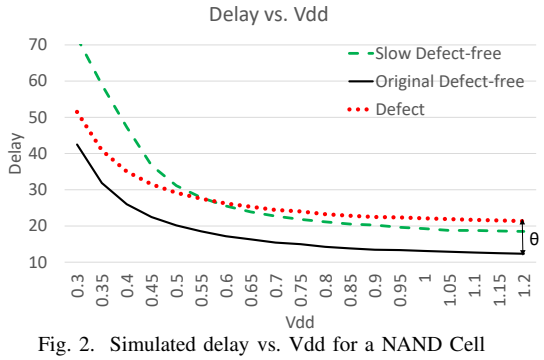

Fig. 2. Simulated delay vs. Vdd for a NAND Cell

Various process parameters can contribute to process variations. Threshold voltage fluctuation is considered as the major source of variations in timing [29] which is the focus of this work on modeling process variations. Assume, V_t is now increased by δ due to process variations. By replacing V_t with $V_t - \delta$ and some mathematical reformulations, Eq. 1 is transformed into:

$$\tau = \underbrace{\frac{C_L (V_{dd} - \delta) L T_{ox}}{\mu \epsilon_{ox} W (V_{dd} - \delta - V_t)^{\alpha}}}_{\text{I}} + \underbrace{\frac{C_L \delta L T_{ox}}{\mu \epsilon_{ox} W (V_{dd} - \delta - V_t)}}_{\text{II}} \quad (2)$$

Part I of Eq. 2 shifts the black curve to the right by δ, part II shifts it upwards with an increased slope. Combining this results in the dashed green curve in Fig. 2, which is in fact observed in simulation.

On the other hand, for constant V_t, the delay of a primitive gate can be expressed by the usual RC model, $\tau = C_L * R_{eff}$, and a resistive open will just increase R_{eff}. The new delay is $\tau' = C_L * (R_{eff} + \delta)$, which means a resistive defect of size δ will shift the black curve of Fig. 2 upwards by $C_L * \delta$ resulting in the dotted red curve. A similar analysis can be performed for resistive bridges and gate oxide pinholes, but is beyond the scope of this paper.

Fig. 2 indicates that it may be impossible to distinguish a slow defect-free and a fast defect cell, based just on the delay measurement for a single voltage. However, the shapes

of the two functions are sufficiently different, such that delay measurements at only 13 different voltages allow a highly accurate classification by statistical learning methods as seen in Section VI. The brief analysis presented here explains also why the experiments in [16] were successful, even if the defect cell is fast due to variations: reduced V_t moves the red curve to the left and reduces the slope, then the resulting curve is significantly different from any defect-free behavior.

A simplified combinational circuit including an inverter chain is modeled to investigate the masking impacts on the circuit delay behavior. The inverter-cell is selected since it has the minimum number of transistors and often the highest impact of the variations on the delay behavior of the embedded cell. The parasitic elements and buffers are added to the chain to consider the fanout impact as well. Fig. 3 depicts a NAND cell with a resistive defect which is equal to $3\sigma_\tau$ of the cell and compares it with the embedded NAND cell in front of a propagation chain of 16 inverters where all the transistors are subject to $\sigma = 0.2\mu$ variation in their length (L) and width (W) individually. From this, μ_τ and σ_τ as the nominal values

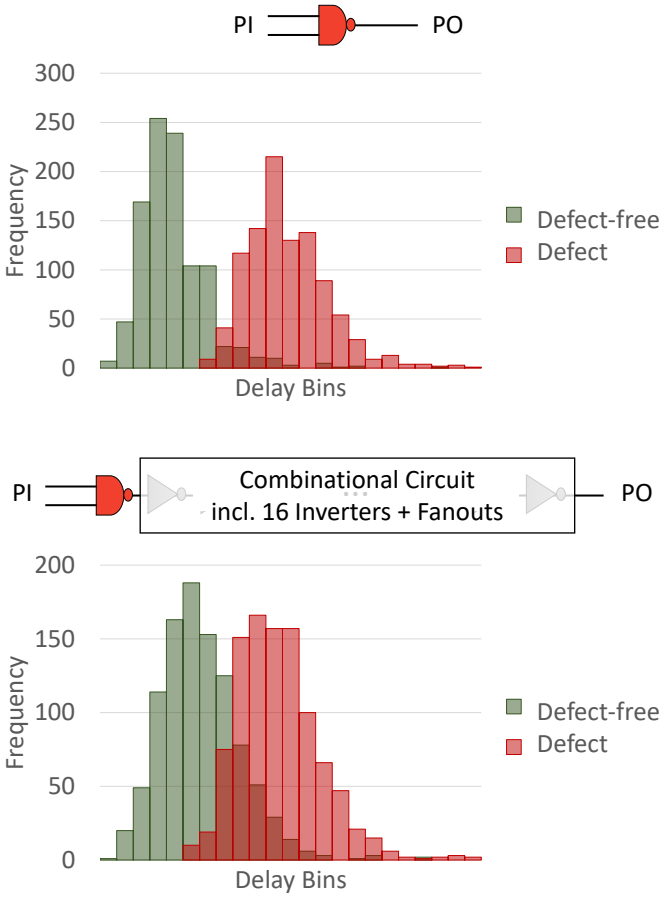

Fig. 3. Defect and defect-free delay histograms of a single NAND and an embedded NAND cell in front of a chain of 16 inverters

The points where the red and green curves of Fig. 2 get close are reflected here as those delay bins where red and green bins overlap. Exactly these bins form the challenge for any classification technique to be addressed in the next two sections. Clearly, the overlap bins are wider and include more instances in the combinational circuit of the embedded cell comparing to a single cell case. Therefore, not only combinational circuit masking impacts are the challenges which should be covered, but also more instances are dealt with in the overlap delay range which need to be classified.

IV. Data Generation

The analysis procedure followed in this paper consists of the four phases depicted in Fig. 4. The first two phases create models for the defect and defect-free circuit as well as corresponding random samples reflecting variations. These two phases are described in detail in this section. The third phase selects and applies appropriate statistical learning schemes to the data set, and finally, the performance of these schemes is validated as described in the next section.

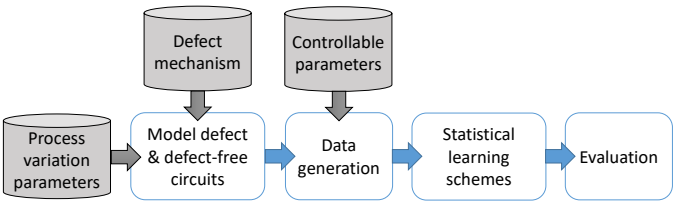

Fig. 4. Classifier generation flow

A. Defect-free circuit model

The study at hand concentrates on a combinational circuit which includes a NAND cell as the embedded cell and an inverter chain as well as parasitic elements to consider fanouts. All cells are from NanGate 15nm OCL which has FinFET transistors [15]. The analysis of a complete library is beyond the scope of the paper and left to an industrial exploitation of the encouraging results presented in Section VI.

The netlist is shown in Fig. 5 where also two of the possible resistive opens are marked. The output of the NAND cell is connected to a chain of λ inverters as already seen in Fig. 3. All transistor delays of the NAND cell and the inverter chain as shown in Fig. 5 are subject to individual random variations following a Gaussian distribution $N(\mu_\tau, \sigma_\tau)$. A more complex modeling with chains of more complex cells and even with individual distributions is possible [26] [27], but does not affect the arguments below.

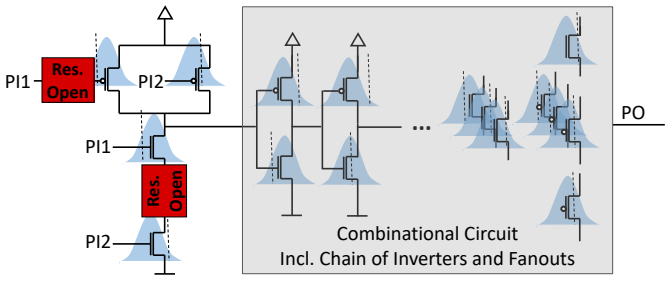

Fig. 5. Embedded NAND cell with 2 possible resistive open locations in a combinational circuit

If μ_{emb} is the nominal delay of the embedded NAND cell and μ_{Inv} is the nominal delay of each inverter, then:

$$\mu_{\tau_{out}} = \mu_{\tau_{emb}} + \lambda * \mu_{\tau_{Inv}} \quad (3)$$

is the expected delay of the entire circuit. For the corresponding standard deviations $\sigma_{\tau_{emb}}$ and $\sigma_{\tau_{Inv}}$, the standard deviation of the circuit output is:

$$\sigma_{\tau_{out}} = \sqrt{\sigma^2_{\tau_{emb}} + \lambda * \sigma^2_{\tau_{Inv}}} \quad (4)$$

In summary, a large data set of modules will be generated for different chain lengths (λ), where all the transistors have individual random delays.

B. Defect circuit model

Resistive opens of varying sizes are injected individually into the embedded NAND cell, at the possible locations as it is depicted in Fig. 5. The defect sizes have to be chosen such that the resulting timing from the circuit primary output overlaps with the possible delay of a defect-free instance. If the delays were too large, there would be no challenge for any Machine Learning (ML) scheme, and standard delay testing would solve the job as well. The minimum defect size considered here is based on the delay of the embedded NAND cell $\tau_f = 3\sigma_{\tau_{emb}}$. It means, if such a defect is injected into a fast instance with the delay $\mu_{\tau_{emb}} - 3\sigma_{\tau_{emb}}$, the defect instance still reaches the nominal speed and is hard to be identified.

The upper limit of the delay size which is interesting to be analyzed is determined by Eq. 3 and Eq. 4. The additional delay τ_f should not exceed the delay increase of an inverter chain with slow transistors: $\tau_f \leq 3\sigma_{\tau_{out}}$. The corresponding defects have to be inserted into the SPICE netlist as a resistor.

C. Monte Carlo simulation

The data set passed to the machine learning schemes is generated by Monte Carlo simulation. For each class of defect and defect-free circuits, N instances are generated in which all transistors may have different process parameter values. The channel lengths (L) and gate widths (W) of transistors are assumed to be individual Gaussian random values in which the mean value (μ_{pv}) is the nominal value of (L) and (W), and the standard deviation (σ_{pv}) equals 20% of μ_{pv} [30]. The data set is generated for four different chain lengths $\lambda = 2, 4, 8, 16$.

The timing behavior of each defect and defect-free instance is simulated under $\omega = 13$ various voltages by transient analysis in Spice. The supply voltage as a controllable parameter gets values between 0.4 V and 1 V with the step of 0.05 V. The defect and defect-free instances which produce a delay in the overlapped range are extracted from the total simulation results and form the data set of size N' which will be further used for the ML training and testing. The data set corresponding to each chain length is presented by a table of N' tuples (rows) of delays, and 13 attributes (columns) for the applied voltages, and a label to indicate defect and defect-free instances. Each circuit with $\lambda = 2, 4, 8, 16$ inverters is handled separately.

V. Classification by supervised learning

A. Learning schemes

Four different statistical learning schemes are trained. The setup used for each scheme is described below. The focus of this work is on investigating the defect classification. The statistical learning scheme setup can be further optimized which is beyond the scope of this work.

- **Support Vector Machines (SVM)**: The Support Vector Machine algorithm separates data points with the largest margin into classes by constructing a hyper-plane between them. SVM is an effective supervised learning method in high dimensional spaces, and the kernel method is useful in implementing non-linear classification [31]. In this work, the Support Vector Classifier (SVC) with *rbf* [32] kernel is used.

- **k-Nearest Neighbors (kNN)**: The k-Nearest Neighbors algorithm finds the distances of a new data point to the k-nearest data points and votes for the most frequent label. kNN algorithm is among the simplest and yet most efficient classification rules and is widely used in practice [33]. The distance is computed by the Euclidean metric.

- **Decision Tree (DT)**: The Decision Tree is a tree whose internal nodes can be taken as tests on input data patterns and whose leaf nodes can be taken as categories of these patterns [34]. Tree-based algorithms empower predictive models and map well to non-linear relationships as well as imbalanced data sets. Moreover, they do not get influenced by outliers to a fair degree and have high execution speed [35]. The Gini coefficient is used as the decision criteria, and a maximum depth of the tree is set to avoid over-fitting.

- **Random Forest (RF)**: The Random Forest scheme is a bagging method which builds up a set of trees and gets the final result based on majority votes. It undertakes dimensionality reduction methods to handle large data sets with higher dimensionality [36]. The same setup as for the DT above is used for this approach and the number of trees in the forest is set based on achieving the highest classification result.

B. Evaluation

The classification quality for the four schemes above is evaluated with respect to the standard metrics used in statistical learning [37]. The attribute P of an instance means "defect" (positive), and N means "no-defect" (negative). A correct classification is true (T), otherwise false (F). The test instances are partitioned into four groups, which the classification quality metrics are calculated based on.

- True Positive (TP): Defect instances correctly identified as defect.
- True Negative (TN): Defect-free instances correctly identified as defect-free.
- False Positive (FP): Defect-free instances wrongly classified as defect.
- False Negative (FN): Overlooked defect instances wrongly classified as defect-free.

Precision indicates the ratio of the classified instances, which are correctly classified. Usually different values for the class defect (D) and no-defect (ND) are observed:

$$Prec_D := \frac{|TP|}{|TP| + |FP|}, Prec_{ND} := \frac{|TN|}{|TN| + |FN|} \quad (5)$$

Recall represents the ratio of the instances in a class, which are classified correctly. Similar to precision, different values for the class defect (D) and no-defect (ND) are observed:

$$Recall_D := \frac{|TP|}{|TP| + |FN|}, Recall_{ND} := \frac{|TN|}{|TN| + |FP|} \quad (6)$$

Accuracy denotes how much of the test data are in total classified correctly. Accuracy is prone to get biased by imbalanced data sets. Assuming N' as the size of the overlap test data, accuracy is defined as below.

$$Accuracy := \frac{|TP| + |TN|}{N'} \tag{7}$$

F1-score is defined as the harmonic mean of the corresponding precision and recall for the defect (D) and no-defect (ND). A high F1-score shows that the classification is not disturbed by false alarms.

$$F1 - score_{(D,ND)} := \frac{2}{\frac{1}{Recall_{(D,ND)}} + \frac{1}{Precision_{(D,ND)}}} \tag{8}$$

All the metrics mentioned above are applied to the four ML schemes. To generate a robust and fair evaluation, K-fold cross-validation is used [38]. The data set of N' instances is partitioned into $k = 10$ disjoint randomly selected subsets. In a round-robin fashion, the label is removed from one subset in order to be used as test set, and the union of the remaining sets is used for training. The metrics above are obtained as the average outcome of these $k = 10$ experiments.

VI. Simulation Results

In the first subsection, the four ML schemes are compared according to the different metrics mentioned above. For the sake of brevity, only the outcomes with respect to one circuit and defect size are compared. The second subsection reports the results of the best scheme for varying circuit and defect sizes.

The minimum defect size $3\sigma_{\tau_{emb}}$ and the maximum interesting defect size $3\sigma_{\tau_{out}}$ are defined by Eq. 4, which result in:

$$DSize(i) = 3\sigma_{\tau_{emb}} + i(\sigma_{\tau_{out}} - \sigma_{\tau_{emb}}) \quad \text{for } i = 0, .., 3 \tag{9}$$

The resistors are injected into the cell, such that the additional delays of Eq. 9 are produced.

A. Comparing ML schemes

To present the comparison of the four schemes, a specific relatively small defect size ($DSize(1)$) and moderate chain length ($\lambda = 8$) are targeted. Table. I shows the results obtained.

Almost all the schemes reach an accuracy significantly larger than 0.8. It is being observed the best results are obtained by the Random Forest (RF) scheme. This is due to the fact that tree-based classification schemes map well to non-linear relationships, as well as imbalanced data sets and get less influenced by outliers. In addition, RF can boost the tree-based models by executing bias-variance trade-off analysis. It also selects the features based on classification scores, which enables it to handle a big set of data. The results from RF scheme will be discussed in further detail. It has to be noted that only N' hard to classify instances, which can be seen as the overlapped range in Fig. 3 are under investigation. Without the proposed technique to distinguish defect from defect-free, this part of the produced chips (N') should be completely removed from the final production to prevent ELF, which means a high and expensive yield loss. By using the proposed technique based on RF scheme, $Prec_D = 0.946$ indicates that only 5.4% of chips classified as defect are misclassified. Depending on the yield, this metric can also be used as reliability binning. $Prec_{ND} = 0.873$ means that in 12.7% of the cases where a warning is not given, it should have been done. Here, it should be considered that the experiment is performed with a balanced data set. In practice, the defect parts should be much less than 50%, and the portion of overlooked cases will be much less as well. $Recall_D = 0.961$ means that only 3.9% of marginal chips will pass, and $Recall_{ND} = 0.838$ denotes that 84% of the marginal chips which are still in the specification, can be sorted out.

TABLE I
Classification quality metrics for DSize(1) and $\lambda = 8$

	SVM	kNN	DT	RF
$Prec_D$	0.76	0.887	0.933	0.946
$Recall_D$	0.886	0.984	0.937	0.961
$F1_D$	0.812	0.931	0.935	0.954
$Prec_{ND}$	0.681	0.95	0.8	0.873
$Recall_{ND}$	0.465	0.619	0.794	0.838
$F1_{ND}$	0.526	0.731	0.78	0.849
Accuracy	0.818	0.898	0.902	0.931

B. Impact of circuit and defect sizes

For the four defect sizes according to Eq. 9 and for the three circuit sizes $\lambda = 0$ (no inverter chain), $\lambda = 4$ and $\lambda = 16$, all the relevant metrics are reported in Table. II.

The general trend is that with increasing defect size the performance of the RF scheme is increasing further. Anomalies ($\lambda = 16$, $DSize(1)$) are due to the frequency distributions of the defect and defect-free instances. It is observed that the timing propagation through an inverter chain of 16 still allows classification.

C. Application scenarios

According to the presented results, the RF scheme is able to identify resistive opens with high classification quality. This result is encouraging enough to train such a ML model on real silicon data for which industrial support is needed.

VII. Conclusion and further work

Resistive opens can be identified in a cell, even if it is deeply embedded into a combinational circuit and it does not change the circuit behavior beyond the specification. A machine learning scheme based on Random Forest is able to classify these cells under variations with a very high accuracy. The encouraging result can be used for quality screening, binning, and a diagnosis with a negligible impact on the yield.

Acknowledgements

This work is part of the project grant WU 245/19-1 (FAST) funded by the German Research Foundation (DFG), and is supported by Advantest as part of the Graduate School "Intelligent Methods for Test and Reliability" (GS-IMTR) at the University of Stuttgart.

TABLE II
RF CLASSIFICATION QUALITY METRICS FOR VARYING DSIZE AND VARYING λ

λ	Defect	$Prec_D$	$Recall_D$	$F1_D$	$Prec_{ND}$	$Recall_{ND}$	$F1_{ND}$	Accuracy
0	DSize(0)	0.917	0.950	0.932	0.971	0.943	0.958	0.952
	DSize(1)	0.957	0.983	0.968	0.962	0.889	0.916	0.959
4	DSize(0)	0.925	0.936	0.93	0.880	0.827	0.849	0.914
	DSize(1)	0.953	0.969	0.962	0.866	0.841	0.846	0.941
	DSize(2)	0.98	0.991	0.986	0.983	0.961	0.974	0.988
16	DSize(0)	0.888	0.912	0.900	0.898	0.866	0.879	0.894
	DSize(1)	0.927	0.957	0.939	0.812	0.671	0.710	0.904
	DSize(2)	0.937	0.963	0.951	0.958	0.926	0.941	0.974
	DSize(3)	0.968	0.978	0.973	0.967	0.929	0.946	0.980

REFERENCES

[1] F. Forero, J. Galliere, M. Renovell, and V. Champac, "Detectability Challenges of Bridge Defects in FinFET Based Logic Cells," 2018.

[2] A. Czutro, N. Houarche, P. Engelke, I. Polian, M. Comte, M. Renovell, and B. Becker, "A Simulator of Small-Delay Faults Caused by Resistive-Open Defects," in *Proc. IEEE European Test Symp. (ETS)*, 2008, pp. 113–118.

[3] P. Engelke, I. Polian, M. Renovell, S. Kundu, B. Seshadri, and B. Becker, "On Detection of Resistive Bridging Defects by Low-Temperature and Low-Voltage Testing," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 2, pp. 327–338, 2008.

[4] S. Hellebrand, T. Indlekofer, M. Kampmann, M. Kochte, C. Liu, and H. Wunderlich, "FAST-BIST: Faster-than-at-Speed BIST Targeting Hidden Delay Defects," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 2014, pp. 1–8.

[5] F. Hapke, J. Schloeffel, W. Redemund, A. Glowatz, J. Rajski, M. Reese, J. Rearick, and J. Rivers, "Cell-Aware Analysis for Small-Delay Effects and Production Test Results from Different Fault Models," in *Proc. IEEE Int'l Test Conf. (ITC)*, Sep. 2011, pp. 1–8.

[6] C. Liu, E. Schneider, M. Kampmann, S. Hellebrand, and H. Wunderlich, "Extending Aging Monitors for Early Life and Wear-Out Failure Prevention," in *Proc. IEEE Asian Test Symp. (ATS)*, Oct. 2018, pp. 92–97.

[7] H. Hao and E. J. McCluskey, "Very-Low-Voltage Testing for Weak CMOS Logic ICs," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 1993, pp. 275–284.

[8] J. T. Chang and E. J. McCluskey, "Detecting Delay Flaws by Very-Low-Voltage Testing," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 1996, pp. 367–376.

[9] P. Engelke, I. Polian, M. Renovell, B. Seshadri, and B. Becker, "The Pros and Cons of Very-Low-Voltage Testing: An Analysis Based on Resistive Bridging Faults," in *Proc. IEEE VLSI Test Symp. (VTS)*, Apr. 2004, pp. 171–178.

[10] X. Qian, Chao Han, and A. D. Singh, "Detection of Gate-Oxide Defects with Timing Tests at Reduced Power Supply," in *Proc. IEEE VLSI Test Symp. (VTS)*, Apr. 2012, pp. 120–126.

[11] S. Mhira, V. Huard, A. Benhassain, F. Cacho, S. Naudet, A. Jain, C. Parthasarathy, and A. Bravaix, "Dynamic Adaptive Voltage Scaling in Automotive environment," in *Proc. IEEE Int'l Reliability Physics Symp. (IRPS)*, Apr. 2017, pp. 3A–4.1–3A–4.7.

[12] R. C. Aitken, "Defect or Variation? Characterizing Standard Cell Behavior at 90 nm and Below," *IEEE Trans. on Semiconductor Manufacturing (TSM)*, vol. 21, no. 1, pp. 46–54, Feb. 2008.

[13] X. Qian and A. D. Singh, "Distinguishing Resistive Small Delay Defects from Random Parameter Variations," in *Proc. IEEE Asian Test Symp. (ATS)*, Dec. 2010, pp. 325–330.

[14] A. Karel, F. Azais, M. Comte, J. Galliere, M. Renovell, and K. Singh, "Detection of Resistive Open and Short Defects in FDSOI under Delay-Based Test: Optimal VDD and Body Biasing Conditions," in *Proc. IEEE European Test Symp. (ETS)*, May 2017, pp. 1–2.

[15] NanGate Inc. (2017) NanGate15nm Open Cell Library. [Online]. Available: http://www.nangate.com/

[16] Z. Najafi-Haghi, M. Hashemipour Nazari, and H. Wunderlich, "Variation-Aware Defect Characterization at Cell Level," in *Proc. IEEE European Test Symp. (ETS)*, May 2020, pp. 1–6.

[17] L. W. Nagel and D. O. Pederson, "SPICE (Simulation Program with Integrated Circuit Emphasis)," EECS Department, University of California, Berkeley, Tech. Rep. UCB/ERL M382, Apr. 1973.

[18] E. Schneider and H.-J. Wunderlich, "SWIFT: Switch Level Fault Simulation on GPUs," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Syst. (TCAD)*, vol. 38, no. 1, pp. 122–135, Jan. 2019.

[19] E. Schneider and H.-J. Wunderlich, "GPU-accelerated Time Simulation of Systems with Adaptive Voltage and Frequency Scaling," Feb. 2020.

[20] K. D. Wagner and E. J. McCluskey, "Effect of Supply Voltage on Circuit Propagation Delay and Test Applications," in *Proc. IEEE Int'l Conf. on Computer-Aided Design (ICCAD)*, Nov. 1985, pp. 42–44.

[21] J. Chin and M. Nourani, "SoC Test Scheduling with Power-time Tradeoff and Hot Spot Avoidance," in *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, vol. 1, 2004, pp. 710–711.

[22] X. Lu, Z. Li, W. Qiu, D. M. H. Walker, and W. Shi, "Longest Path Selection for Delay Test under Process Variation," in *Proc. IEEE Asia-South Pacific Design Automation Conf. (ASP-DAC)*, 2004, pp. 98–103.

[23] N. B. Zain Ali, M. Zwolinski, B. M. Al-Hashimi, and P. Harrod, "Dynamic Voltage Scaling Aware Delay Fault Testing," in *Proc. IEEE European Test Symp. (ETS)*, May 2006, pp. 15–20.

[24] F. Forero, J. Galliere, M. Renovell, and V. Champac, "Analysis of Short Defects in FinFET Based Logic Cells," in *Proc. IEEE Latin American Test Symp. (LATS)*, Mar. 2017, pp. 1–6.

[25] M. Sauer, I. Polian, M. E. Imhof, A. Mumtaz, E. Schneider, A. Czutro, H.-J. Wunderlich, and B. Becker, "Variation-Aware Deterministic ATPG," in *Proc. IEEE European Test Symp. (ETS)*, May 2014, pp. 1–6.

[26] M. Wagner and H.-J. Wunderlich, "Efficient Variation-Aware Statistical Dynamic Timing Analysis for Delay Test Applications ," in *Proc. ACM/IEEE Conf. on Design, Automation Test in Europe (DATE)*, Mar. 2013, pp. 276–281.

[27] M. Wagner and H.-J. Wunderlich, "Probabilistic Sensitization Analysis for Variation-Aware Path Delay Fault Test Evaluation," in *Proc. IEEE European Test Symp. (ETS)*, May 2017, pp. 1–6.

[28] J. García Gervacio and V. Champac, *Timing Performance of Nanometer Digital Circuits under Process Variations*, Apr. 2018.

[29] V. P. Yanambaka, S. P. Mohanty, E. Kougianos, D. Ghai, and G. Ghai, "Process Variation Analysis and Optimization of a FinFET-Based VCO," *IEEE Trans. on Semiconductor Manufacturing (TSM)*, vol. 30, no. 2, pp. 126–134, May 2017.

[30] T. Matsukawa, S. O'uchi, K. Endo, Y. Ishikawa, H. Yamauchi, Y. X. Liu, J. Tsukada, K. Sakamoto, and M. Masahara, "Comprehensive Analysis of Variability Sources of FinFET Characteristics," in *Proc. IEEE Symp. on VLSI Technology*, 2009, pp. 118–119.

[31] J. A. Gualtieri and S. Chettri, "Support Vector Machines for Classification of Hyperspectral Data," in *IEEE Int'l Geoscience and Remote Sensing Symp.(IGARSS)*, vol. 2, 2000, pp. 813–815 vol.2.

[32] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

[33] J. Laaksonen and E. Oja, "Classification with Learning K-nearest Neighbors," in *Proc. of Int'l Conf. on Neural Networks (ICNN)*, vol. 3, 1996, pp. 1480–1483 vol.3.

[34] A. Navada, A. N. Ansari, S. Patil, and B. A. Sonkamble, "Overview of Use of Decision Tree Algorithms in Machine Learning," in *IEEE Control and System Graduate Research Colloquium*, 2011, pp. 37–42.

[35] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[36] Tin Kam Ho, "Random Decision Forests," in *Proc. of Int'l Conf. on Document Analysis and Recognition*, vol. 1, 1995, pp. 278–282 vol.1.

[37] S. Marsland, *Machine Learning: An Algorithmic Perspective*, 2nd ed. Chapman and Hall/CRC, 2014.

[38] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," in *Proc. Int'l Joint Conf. on Artificial Intelligence (IJCAI)*, Aug. 1995, pp. 1137–1143.