Preprint

Variation-Aware Deterministic ATPG

Sauer, Matthias and Polian, Ilia and Imhof, Michael E. and Mumtaz, Abdullah and Schneider, Eric and Czutro, Alexander and Wunderlich, Hans-Joachim and Becker, Bernd

Proc. 19th IEEE European Test Symposium (ETS'14), Paderborn, Germany, May 26-30 , pp. 87-92

doi: 10.1109/ETS.2014.6847806

This is the author's "personal copy" of the final, accepted version of the paper published by IEEE.

IEEE COPYRIGHT NOTICE

© 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Variation-Aware Deterministic ATPG

Matthias Sauer†, Ilia Polian‡, Michael E. Imhof‡, Abdullah Mumtaz‡, Eric Schneider‡, Alexander Czutro*, Hans-Joachim Wunderlich‡, Bernd Becker*

*University of Freiburg, Georges-Köhler-Allee 51, 79110 Freiburg, Germany
†Faculty of Informatics and Mathematics, University of Passau, Instr. 43, 94032 Passau, Germany
‡ Institute of Computer Architecture and Computer Engineering
University of Stuttgart, Pfaffenwaldring 47, 70569 Stuttgart, Germany

Abstract—In technologies affected by variability, the detection status of a small-delay fault may vary among manufactured circuit instances. The same fault may be detected, missed or provably undetectable in different circuit instances. We introduce the first complete flow to accurately evaluate and systematically maximize the test quality under variability. As the number of possible circuit instances is infinite, we employ statistical analysis to obtain a test set that achieves a fault-efficiency target with an user-defined confidence level. The algorithm combines a classical path-oriented test-generation procedure with a novel waveform-accurate engine that can formally prove that a small-delay fault is not detectable and does not count towards fault efficiency. Extensive simulation results demonstrate the performance of the generated test sets for industrial circuits affected by uncorrelated and correlated variations.

Keywords-Variation-aware test, fault efficiency, ATPG

I. INTRODUCTION

Recent semiconductor manufacturing technology nodes are heavily affected by massive parameter variations which create severe challenges for design and test of state-of-the-art circuits [1, 2]. Manufacturing defects interact with the effects of variations in complex and often unpredictable ways. This holds in particular for small-delay faults, which are defined as fixed amounts of extra delay at a given gate. Traditional fault-coverage metrics and test-generation methods are insufficient for circuit populations affected by variations. They ignore the fact that a given test set could detect a fault in some manufactured instances of the circuit but miss the same fault in other instances. The fault may also be provably undetectable in some of the instances, and such faults should not count towards fault efficiency. However, the parameters affected by variability, i.e., delays, are continuous and the number of possible circuit instances is infinite.

A further substantial difficulty in variation-aware testing is the reliance of traditional test methods on the concept of path sensitization [3, 4]. It is assumed that, in order to detect a small-delay fault \( f \) on a given gate, it is sufficient to find a test pair that sensitizes the longest path that goes through that gate. However, even in absence of variations, such procedures are not suited to identify undetectable faults. They enforce certain sensitization conditions, including robust, non-robust and functional sensitization. If no robustly sensitizable path of sufficient length (that is, cumulative delay) exists through the fault location, the fault is not proven undetectable, as detection through non-robustly or functionally sensitized path could be possible. On the other hand, non-robust or functional sensitization is prone to invalidation, and faults for which such a path has been found may actually not be detected [5]. This situation is even worse under variations where the same path may have a length that is sufficient for detection in one instance but not sufficient in another.

In this paper, we present a statistical metric that generalizes the concept of fault efficiency \( (FE) \) to circuits under variations. We assume that variations are described by a known probability distribution, which may be correlated or uncorrelated. The metric consists of three parts: fault efficiency bound \( FE_{\text{min}} \), probability \( c \) and confidence \( \gamma \). An individual circuit instance \( i \) has fixed delays, and the fault efficiency \( (FE) \) of a test (pair) set \( T \) with respect to an instance \( i \) is the number of faults detected by \( T \) divided by the number of detectable faults in \( i \). Under variations, there are infinitely many instances \( i \), and the same test set \( T \) may have different \( FE \) on different instances. \( T \) achieves fault efficiency \( FE_{\text{min}} \) with probability \( c \) if the share of instances on which the fault efficiency of \( T \) is equal to or greater than \( FE_{\text{min}} \) is at least \( c \). In other words, if a large number \( N \) of random circuit instances is drawn according to the known probability distribution, at least \( c \cdot N \) of them will have \( FE \geq FE_{\text{min}} \) and at most \( (1 - c) \cdot N \) may have \( FE < FE_{\text{min}} \).

We introduce an automatic test generation algorithm that produces a test set that achieves \( FE_{\text{min}} \) with probability \( c \). This property of the test set holds with confidence level \( \gamma \). The probability distribution and the parameters \( FE_{\text{min}}, c \) and \( \gamma \) are inputs of the algorithm. Larger values of these parameters result in a larger test set. In contrast to previous variation-aware test generation approaches [6–12], the proposed algorithm relies on accurate fault efficiencies of the generated test sets. For this reason, the algorithm employs a very recently introduced SAT-based test-generation engine WaveSAT [5] that works with waveform precision and does not rely on path sensitization. WaveSAT is able to accurately classify a given fault, that is, to generate a test pair that detects it or to prove that the fault is undetectable and should be removed from fault efficiency calculation.

We applied our algorithm to mid-size industrial circuits and were able to generate test sets for large values \((\geq 0.98)\) of \( FE_{\text{min}}, c \) and \( \gamma \). The approach is validated on uncorrelated and correlated probability distributions. The scalability is achieved by complementing WaveSAT by a fast timing-aware path-
sensitization procedure PHAETON [13, 14] and an efficient
parallel delay fault simulator fsim running on general-purpose
graphics processing units (GPGPUs) [15].

The remainder of the paper is organized as follows. Section II
provides background on variation-aware test and the metrics
used in this paper. It also introduces the statistical model of
fault detection with a guaranteed confidence level. Section III
starts with a brief review of procedures which generate
test pairs for faults in circuit instances with fixed delays.
The discussion of these procedures and their properties is
essential as they are the basis of the adaptive confidence-guided
algorithm, which is presented in the remainder of Section III.
Experimental results are reported in Section IV. Section V
concludes the paper.

II. VARIATION-AWARE TEST

A. Circuit model under variations

We assume that variability affects the delays of all gates of
a circuit while leaving its topology unchanged. Therefore, gate
delays are random variables described by a probability distribu-
tion. While uncorrelated variations have been predicted to be
dominant in the nanoscale technologies [16–18], our approach
is based on Monte-Carlo experiments and therefore works
independent of the distribution. We will report experimental
results for both uncorrelated and uncorrelated variations. In
this paper, we assume circuits composed of primitive gates and
a simplified timing model: each gate $g_i$ has an input-to-output
delay $p_j$. Each $p_j$ is a random variable, and a circuit $C$ with
$n$ gates is completely described by a parameter configuration
$p = (p_1, \ldots, p_n)$. This model can be easily extended by
incorporating different rising and falling delays, pin-to-pin
delays and pattern-dependent delays, resulting in more random
variables per circuit. It is also possible to include travel times
on wires and pulse filtering.

A circuit $t = C[p]$ where all gate delays have fixed values
$(p_1, \ldots, p_n)$ is called a circuit instance or simply an instance.
The nominal instance $i_0$ has all delays set to the expected
values of their distributions. When circuits are manufactured,
each produced circuit is described by an instance. Manufac-
turing $N$ circuits corresponds to a Monte-Carlo experiment
where the parameters $(p_1, \ldots, p_n)$ of each instance are drawn
according to the distribution. It is possible to simulate this
process by drawing sets of random parameters without actually
manufacturing the circuit. This Monte-Carlo simulation yields
a set of instances called population. Properties that are valid
for the simulated population of sufficient size hold for different
populations, including the actual manufactured circuits, with
some probability called confidence level.

B. Fault detection under variations

We consider small-delay faults (SDFs) that are defined at
logic gates in the circuit. An SDF $(g, s)$ results in a slow-down
of gate $g$ by $s$ time units. An SDF is detected by a test pair
$(v_1, v_2)$ if the circuit affected by this SDF has an incorrect
value on at least one output at the observation time $t_{obs}$. We
define $t_{obs}$ as the time when the circuit’s outputs are read out.
$t_{obs}$ typically equals the clock cycle duration, and a (fault-free)
circuit instance in which all possible transitions are finished
before $t_{obs}$ is called timing-correct. We assume that the value
of $t_{obs}$ is not affected by variations and is hence equal for all
circuit instances in the population. Under normal circumstances,
the designer will set $t_{obs}$ such that the majority of (fault-free)
circuit instances in a population are timing-correct and only a
few instances have delays $(p_1, \ldots, p_n)$ that are so large that
$t_{obs}$ is exceeded. Such instances contribute to the parametric
yield loss and are not considered here.

Detection of an SDF in two individual instances of the same
circuit, $i_1$ and $i_2$, is illustrated in the left and the right part of
Figure 1, respectively. The SDF of size $\varepsilon = 3.5$ would be detected in instance $i_1$ (transition on the output at time 10.5, after $t_{obs} = 8$) and not detected in instance $i_2$ (output transition at time 7.5, before $t_{obs}$). However, this fault is detectable in instance $i_2$ through path $B$-g1-g2-g4, sensitized by a different test pair $AB = 11/10$ and inducing the output transition at time 8.5.

A fault is undetectable if there is no path of sufficient length
($\geq t_{obs} - s$) through its location. We refer to this case by the
term structural undetectability. In our example, the SDF on
g1 with size $\varepsilon = 2$ is detected in instance $i_1$ but structurally
undetectable in instance $i_2$. If a sufficiently long path does
exist, it may or may not be possible to find a test pair that
sensitizes it such as to induce the incorrect value at time $t_{obs}$.
In summary, a given test set $T$ partitions the set $F$ of all
considered SDFs in circuit instance $i$ into three parts: $F = F_{\text{det}}(i, T) \cup F_{\text{nd}}(i) \cup F_{\text{mis}}(i, T)$, where $F_{\text{det}}(i, T)$, $F_{\text{nd}}(i)$ and $F_{\text{mis}}(i)$ are the sets of faults detected by $T$ in instance $i$, faults that are provably undetectable in instance $i$, and faults that are
detectable in $i$ but have been missed by $T$, respectively. Note
that $F_{\text{nd}}(i)$ depends on the instance (the same fault could be
undetectable in some instances and detectable in others) but
not on test set $T$. The fault efficiency of a test set $T$ with
respect to a fault set $F$ in an instance $i$ is

$$ FE(F, T, i) = \frac{|F_{\text{det}}(i, T)|}{|F \setminus F_{\text{nd}}(i)|}. $$

(1)

It is important to point out the significance of the set $F_{\text{nd}}(i)$
of undetectable faults. When stuck-at faults are considered,
most circuits have no or very few undetectable faults. In case
of small-delay faults considered here, the detection depends
on the relationship between the slack of the sensitized path
(the difference between $t_{obs}$ and its length) and the size $s$
of the SDF. If many SDFs have a size that is close to the
slack of a sensitizable path, they may be detectable in some
Circuit instances but undetectable in others. Fault coverage
metrics employed in previous work [19] that did not incorporate
accurate detectability status yielded quite low coverages (5% to 35%) for test sets that achieve 76% to 86% fault efficiency according to Eq. 1. Since our test generation is controlled by $FE$ and we aim at rather high values (in excess of 98%), we employ a sophisticated undetectability check to each missed fault to decide whether it counts towards the denominator in Eq. 1 or not.

C. Statistical fault efficiency for circuit populations

The fault efficiency metric defined above is valid for one specific circuit instance $i$ described by a given parameter configuration $(p_1, \ldots, p_n)$. In this section, we introduce a statistical metric that holds for an entire population of circuit instances that share the same probability distribution of random variables $(p_1, \ldots, p_n)$. As already mentioned in the introduction, this metric evaluates a test set $T$ with respect to the three values $FE_{min}$, $c$ and $\gamma$. For $N$ randomly drawn circuit instances of a given parameter space, test set $T$ achieves a fault efficiency of $FE_{min}$ with probability $c$ for an instance $i$ (of the same distribution), if at least $\lceil (c \cdot N) \rceil$ out of the $N$ instances have a fault efficiency that matches or exceeds $FE_{min}$:

$$\Pr[FE(F, T, i) \geq FE_{min}] \geq c.$$  \hspace{1cm} (2)

Recall that calculating the fault efficiency (Eq. 1) requires classification of all considered faults into sets $F_{det}(i, T)$, $F_{not}(i)$ and $F_{min}(i)$.

Since the number of different instances in the parameter space is infinite, it is impossible to evaluate Eq. 2 directly. Instead, $FE_{min}$ will be estimated with respect to the full parameter space by the following Monte-Carlo experiment:

- Draw $k$ random instances of the circuit using the known probability distribution.
- If the fault efficiency of each of the $k$ instances is equal to or larger than $FE_{min}$, assume that Eq. 2 holds.

However, this conclusion is not universal but rather holds with a certain confidence $\gamma$ that is related to the number of repetitions $k$ and the probability $c$. For example, if $k = 1$, only one random instance is evaluated and if its fault efficiency exceeds $FE_{min}$, the test set fulfills Eq. 2. There is still a probability that this particular random instance simply happened to have an excessively high $FE$ by chance and that the performance of the test set $T$ on other instances is worse than $FE_{min}$. Therefore, the confidence $\gamma$ for guaranteeing the probability of achieving $FE_{min}$ is considerably low. On the other hand, for $k = 100$ the test set must exceed $FE_{min}$ on 100 random instances in a row which raises the confidence. We will now establish the formal relationship between $k$, $c$ and $\gamma$.

Assume that the actual probability for test set $T$ having a fault efficiency of $FE_{min}$ with respect to $N$ random instances is $\tilde{c}$. In other words, this means that out of $N$ instances, $FE$ of $T$ will exceed $FE_{min}$ for $(\tilde{c} \cdot N)$ instances. Running the Monte-Carlo experiment as described above will lead to a positive result and find a sequence of $k$ instances where the $FE$ of each instance is larger than or equal to $FE_{min}$ with a probability of $\tilde{c}^k$. Now, the confidence level $\gamma$ captures the probability that this positive result of the Monte-Carlo experiment implies Eq. 2. This is best formulated using the opposite event, namely that the outcome of the Monte-Carlo experiment is positive but the actual probability $\tilde{c}$ is less than the desired probability $c$. Hence, the result of Monte-Carlo experiment was positive and misled to the belief that the probability is no less than $c$ while in reality it was $\tilde{c} < c$.

The probability for this opposite event is $(1 - \gamma)$. Therefore, the probability of the positive result of the Monte-Carlo experiment, or $\tilde{c}^k$, must be less than $(1 - \gamma)$ for all $\tilde{c} < c$. From $\tilde{c}^k \leq 1 - \gamma$ for all $\tilde{c} < c$, it follows $\tilde{c}^k \leq 1 - \gamma$. This implies that in order to obtain a desired confidence level $\gamma$, the following number $k$ of instances must be considered:

$$k \geq \left\lceil \frac{\ln(1 - \gamma)}{\ln(c)} \right\rceil.$$  \hspace{1cm} (3)

Eq. 3 delivers $k$ with the following property. If a test set $T$ is applied to $k$ randomly generated instances and exceeds fault efficiency $FE_{min}$ for all of them, then the probability that an arbitrary randomly generated circuit instance will exceed fault efficiency $FE_{min}$ is $c$ with confidence $\gamma$. This property will be utilized in the adaptive test generation procedure in Section III.

III. CONFIDENCE-GUIDED TEST GENERATION

The confidence-guided test generation procedure aims at producing a test set $T$ that achieves a certain desired fault efficiency $FE_{min}$ on any arbitrary instance with probability $c$ and confidence $\gamma$. As explained in Section II-C, a test set $T$ fulfills this condition if it exceeds $FE_{min}$ for $k$ randomly generated circuit instances, where $k$ is calculated using Eq. 3.

The pseudo-code of the procedure is found in Algorithm 1. Before the algorithm is explained, key sub-routines called by the method and their relevant properties are outlined.

- **PHAETON** is a SAT-based path-oriented small-delay test pattern generator [13, 14]. Given a set of faults $F$ and a circuit instance $i$, it searches, for each fault $f \in F$, for the longest path through the fault location that can be sensitized in $i$ and calculates the test pair that does the sensitization. In this work, PHAETON uses robust path
sensitization that is not prone to invalidations but may miss detectable faults.

- WaveSAT is a small-delay ATPG engine that generates timed sequences of rising and falling transitions (waveforms) on the circuit lines [5] that lead to the detection of the fault. Since WaveSAT does not necessarily sensitize the longest possible path, its generated test pair do not tend to detect many faults. However, WaveSAT allows accurate classification: if no test pair is found, it is guaranteed that none exists, i.e., the fault is undetectable under the given timing assumptions. WaveSAT is used for two purposes: close the gaps in fault coverage by pinpointedly generating test pairs for faults missed by PHAETON test pair sets, and prove that a fault is undetectable in a specific circuit instance and can be excluded from fault-efficiency calculation.

- fsim is a delay fault simulator that runs in parallel on general-purpose graphic processing units (GPGPUs) [15]. The high degree of parallelization is beneficial when grading a large number of circuit instances that have the same structure but different delays.

To construct $T$ with the desired property, the nominal instance $i_0$ and a number of further instances $i_1, i_2, \ldots$, called the training set, are considered (the number of instances in the training set is not known ahead of time, as new instances are added to the training set until the test set reaches the required quality). The initial $T$ is generated to obtain 100% fault efficiency on the nominal instance $i_0$ in Lines 1 through 4 of Algorithm 1. PHAETON is used to quickly generate a good test set and WaveSAT to cover the missed faults identified by fsim. In Line 5, the number $k$ of circuit instances that need to be considered to obtain the fault-efficiency estimate with the desired confidence $\gamma$ is calculated according to Eq. 3. Then, Lines 7 through 16 are repeated until the test set $T$ fulfills the $FE$ target, that is, exceeds $FE_{\text{min}}$ for $k$ consecutive random instances $i$. This is implemented as follows.

The random instance $i$ is generated in Line 8, and the faults $F_{\text{det}}$ detected by $T$ are determined by fault simulation in Line 9. The set of undetectable faults $F_{\text{nd}}$ is obtained by running WaveSAT and collecting all faults shown to be undetectable. In addition, WaveSAT generates top-up test pairs for faults not detected by the initial $T$; these pairs are collected in set $T_{\text{tu}}$. If at least one of $k$ instances does not reach fault efficiency of $FE_{\text{min}}$, the top-up pairs are added to $T$ and another $k$ instances are considered (this is achieved by setting the loop variable $j$ to 0). The iterations stop when $T$ exceeds $FE_{\text{min}}$ for $k$ instances in a row. This means that $T$ is of sufficient quality and fulfills the specification; therefore, this $T$ is returned in Line 18.

**Algorithm 1** Confidence-guided test generation algorithm.

**Procedure** conf_testgen($C, F, FE_{\text{min}}, c, \gamma$)

**Inputs:** Circuit $C$ with random parameter distribution, fault set $F$, fault efficiency target $FE_{\text{min}}$, probability $c$, confidence $\gamma$.

**Output:** Test set $T$ with fault efficiency $\geq FE_{\text{min}}$ with probability $c$ and confidence $\gamma$.

1. $i := \text{nominal\_instance}(C)$;
2. $T := \text{PHAETON}(i, F)$;
3. $F_{\text{det}} := \text{fsim}(i, T, F)$;
4. $T := T \cup \text{WaveSAT}(i, F \setminus F_{\text{det}})$;
// $T$ has 100% fault efficiency on the nominal instance
5. $k := \left\lceil \frac{\ln(1 - \gamma)}{-\ln(c)} \right\rceil$;
6. $T_{\text{tu}} := \emptyset$; $j := 1$;
7. while ($j \leq k$) begin
8.  $i := \text{random\_instance}(C)$;
9.  $F_{\text{det}} := \text{fsim}(i, T, F)$;
10. $T := \text{WaveSAT}(i, F \setminus F_{\text{det}})$; $T_{\text{tu}} := T_{\text{tu}} \cup T$;
11. $F_{\text{nd}} := F \setminus \text{fsim}(i, T, F \setminus F_{\text{det}})$;
12. if ($\frac{|F_{\text{det}}|}{|F|} < FE_{\text{min}}$) then begin
// Insufficient $FE$ for one of $k$ instances, enlarge $T$
13. $T := T \cup T_{\text{tu}}$; $T_{\text{tu}} := \emptyset$;
14. $j := 0$; // Evaluate $k$ new instances
15. end if
16. $j := j + 1$;
17. end while // Sufficient $FE$ for $k$ instances in a row, $T$ finished
18. return $T$;
end conf_testgen

For each circuit, we generated one nominal instance $i_0$ with all delays set to their nominal values, and 1000 instances $i_1, \ldots, i_{1000}$ with delays described by independent random variables according to a Gaussian distribution with standard deviation of 20%. We considered small-delay faults at 100 randomly chosen but fixed fault locations (gates) with nine different sizes equiprobably distributed across the clock cycle, i.e., 10% of $t_{\text{obs}}, 20\%$ of $t_{\text{obs}}$, through 90% of $t_{\text{obs}}$. A substantial portion of these faults are provably undetectable, as no sensitizable path of sufficient length exists through their location. We generated a number of test sets $T_N$ with complete coverage of detectable faults (fault efficiency of 100%) on the instances $i_0$ through $i_N$. This has been done by running PHAETON on $i_0$, simulating the generated test set on instances $i_0, \ldots, i_N$, and using WaveSAT to either detect all undetected faults or classify them as provably undetectable. For example, test set $T_{10}$ detects all detectable faults in the nominal instance $i_0$ and ten further instances $i_1$ through $i_{10}$.

The quality of test sets $T_N$ is reported in Columns 6–15 of Table 1 (the numbers for 5-detect and 10-detect timing-aware transition fault test sets generated by a commercial tool by sensitizing the longest path through a fault location are given in Columns 4 and 5 for reference). Column 3 contains the number of detectable faults aggregated over all 1001 instances. For each test set, the size of the test set $|T|$, the number of detected faults in all instances and the fault efficiency $FE$, obtained by dividing this number by the value in Column 3, are quoted. It can be seen that the fault efficiency of variation-

**IV. EXPERIMENTAL RESULTS**

**A. Explicit test generation for random circuit instances**

To evaluate the need for considering variations during test generation, we performed the following experiment for combinational cores of industrial circuits provided by NXP.
The size of instances with instances. The pattern generation time (pattern generation time $T$) is about twice the size of the 10-detect test set, while reaching a fault efficiency of more than 99%. The test set size $T_N$ quickly saturates with growing $N$, exceeding 99% for low double-digit values of $N$. The test set size $|T_N|$ increases only moderately with the size of the training set, that is, the number of instances is considered. Moreover, the fault efficiency of test set $T_N$ increases among these 200 random circuit instances. The percentage $\tilde{c}$ of individual instances for which the test sets exceed $F_{E_{\text{min}}}$ is always much larger than the target value $c$, and in most cases even reaches 100%. The runtimes associated with the confidence-driven test generation are moderate and do not exceed a few hours, even for large circuits under high confidence and probability targets. Recall that the test sets are evaluated on random circuit instances unrelated to the ones for which they have been generated. Interestingly, while the number of considered instances $I_{\gamma}$ appears to scale with $k$, the differences between the test set sizes and fault efficiencies reached for a fixed circuit are rather small.

We also applied the test sets (that were obtained assuming uncorrelated variations) to a population that consists of 1025 correlated instances. The correlated instances were drawn from a two-dimensional parameter space that describes gate delay distributions with multiple gradients and different orientations in order to model spatial correlations. This parameter space has the same mean and variance over all instances as compared to the random cases.

The results can be found in Table IV. Columns $FE$ quote the average fault efficiency of the generated test set among all the correlated instances, and columns $\tilde{c}$ contain the percentage of instances with $FE$ exceeding $F_{E_{\text{min}}}$.

It can be seen that the obtained test sets are of extremely high quality. The average $FE$ substantially exceeds the target $F_{E_{\text{min}}}$ in all cases. The percentage $\tilde{c}$ of individual instances for which the test sets exceed $F_{E_{\text{min}}}$ is always much larger than the target value $c$, and in most cases even reaches 100%. The runtimes associated with the confidence-driven test generation are moderate and do not exceed a few hours, even for large circuits under high confidence and probability targets. Recall that the test sets are evaluated on random circuit instances unrelated to the ones for which they have been generated. Interestingly, while the number of considered instances $I_{\gamma}$ appears to scale with $k$, the differences between the test set sizes and fault efficiencies reached for a fixed circuit are rather small.

### B. Confidence-driven test generation

The application of the confidence-guided test-generation algorithm $conf_{\text{testgen}}$ is reported in Table III. Recall that the algorithm takes the circuit, the fault set (same as in the first experiment), the fault-efficiency target $F_{E_{\text{min}}}$, the confidence level $\gamma$ and the probability $c$ as inputs, where $c$ stands for likelihood that the generated test set $T$ exceeds $F_{E_{\text{min}}}$ on a randomly generated circuit instance. Results are reported for a number of $\gamma$ values (groups of columns) and a number of $F_{E_{\text{min}}}$ with $c$ set to the same value (groups of rows). Since both $\gamma$ and $c$ affect the number of instances $k$ that have to be considered in the Monte-Carlo experiments (Eq. 3), the respective value of $k$ is denoted in Table II for each combination.

Table III shows six columns for each experiment performed: The size $|I_{\gamma}|$ of the training set, that is, the number of random circuit instances considered before test $T(I_{\gamma})$ passed the statistical test of Eq. 3; the size $|T(I_{\gamma})|$ of this test set; the pattern generation time $t_p$; and the simulation time $t_s$; the fault efficiency $FE$ achieved by this test set on 200 new random instances (not included in the training set); and the percentage $\tilde{c}$ of instances with $FE \geq F_{E_{\text{min}}}$ among these 200 random instances.

### Table II

Value of $k$ depending on target probability $c$ and confidence $\gamma$.

<table>
<thead>
<tr>
<th>$c$</th>
<th>$k_1$</th>
<th>$k_2$</th>
<th>$k_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.98</td>
<td>149</td>
<td>194</td>
<td>228</td>
</tr>
<tr>
<td>0.985</td>
<td>199</td>
<td>259</td>
<td>305</td>
</tr>
</tbody>
</table>

### Table I

Test set size $|T|$, numbers of detected faults $|F|$ and fault efficiency $FE$ [%] over 1001 circuit instances (9 fault sizes).

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Gates Detectable</th>
<th>Timing-aware for 10 faults</th>
<th>Variation-aware test sets $T_N$ generated for instances $i_0, \ldots, i_N$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S-detected</td>
<td>10-detected</td>
<td>$T_0$</td>
</tr>
<tr>
<td>p45k</td>
<td>25679</td>
<td>210290</td>
<td>[49x233]</td>
</tr>
<tr>
<td>p78k</td>
<td>70475</td>
<td>434292</td>
<td>[32x245]</td>
</tr>
<tr>
<td>p89k</td>
<td>58638</td>
<td>155318</td>
<td>[64x262]</td>
</tr>
<tr>
<td>p100k</td>
<td>61066</td>
<td>201883</td>
<td>[128x289]</td>
</tr>
</tbody>
</table>

All experiments have been conducted on a host system equipped with Intel Xeon processors clocked at 2.8GHz, 256GB RAM and NVIDIA GeForce GTX 480 graphics processing cards with 1.6GB RAM.
TABLE III
NUMBER OF CONSIDERED INSTANCES $|I_c|$, TEST SET SIZE $|T(I_c)|$, PATTERN GENERATION TIME $t_p$, SIMULATION TIME $t_s$, FAULT EFFICIENCY $FE$ AND ACTUAL PERCENTAGE OF SUFFICIENTLY COVERED INSTANCES $\tilde{c}$ FOR FAULT-EFFICIENCY TARGET $FE_{\text{min}}$, TARGET PROBABILITY $c$ AND CONFIDENCE $\gamma$.

<table>
<thead>
<tr>
<th>Fault Efficiency Target $FE_{\text{min}}$</th>
<th>Target Probability $c$</th>
<th>$\gamma = 0.95$</th>
<th>$\gamma = 0.98$</th>
<th>$\gamma = 0.99$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit</td>
<td>$</td>
<td>I_c</td>
<td>$</td>
<td>$</td>
</tr>
<tr>
<td>p45k</td>
<td>248</td>
<td>547</td>
<td>1109</td>
<td>808</td>
</tr>
<tr>
<td>p98k</td>
<td>248</td>
<td>458</td>
<td>1322</td>
<td>1519</td>
</tr>
<tr>
<td>p45k</td>
<td>301</td>
<td>517</td>
<td>1108</td>
<td>983</td>
</tr>
<tr>
<td>p78k</td>
<td>352</td>
<td>1337</td>
<td>532</td>
<td>2606</td>
</tr>
<tr>
<td>p89k</td>
<td>314</td>
<td>430</td>
<td>1321</td>
<td>2050</td>
</tr>
<tr>
<td>p100k</td>
<td>575</td>
<td>755</td>
<td>6532</td>
<td>3801</td>
</tr>
</tbody>
</table>

TABLE IV
TEST SET VALIDATION ON 1025 CORRELATED INSTANCES.

<table>
<thead>
<tr>
<th>Fault Efficiency Target $FE_{\text{min}}$</th>
<th>Target Probability $c$</th>
<th>$\gamma = 0.95$</th>
<th>$\gamma = 0.98$</th>
<th>$\gamma = 0.99$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit</td>
<td>$FE$</td>
<td>$\tilde{c}$</td>
<td>$FE$</td>
<td>$\tilde{c}$</td>
</tr>
<tr>
<td>p45k</td>
<td>99.276</td>
<td>91.4</td>
<td>99.276</td>
<td>91.4</td>
</tr>
<tr>
<td>p78k</td>
<td>99.825</td>
<td>99.9</td>
<td>99.825</td>
<td>99.9</td>
</tr>
<tr>
<td>p89k</td>
<td>99.308</td>
<td>85.2</td>
<td>99.308</td>
<td>85.2</td>
</tr>
<tr>
<td>p100k</td>
<td>99.342</td>
<td>96.6</td>
<td>99.342</td>
<td>96.6</td>
</tr>
<tr>
<td>p45k</td>
<td>99.257</td>
<td>81.9</td>
<td>99.257</td>
<td>81.9</td>
</tr>
<tr>
<td>p78k</td>
<td>99.835</td>
<td>99.9</td>
<td>99.835</td>
<td>99.9</td>
</tr>
<tr>
<td>p100k</td>
<td>99.668</td>
<td>97.6</td>
<td>99.668</td>
<td>97.6</td>
</tr>
</tbody>
</table>

V. Conclusions

We presented the first test generation method that produces test sets with proven statistical performance under variations. Its core procedure iteratively enriches the test set by explicitly considering random circuit instances, guided by an estimated confidence level. The method employs the latest SAT-based timing-aware test generation engines which can provide complete testability characterization of all faults in a circuit instance, thus extending the concept of fault efficiency to variation-aware testing for the first time. The obtained test sets guarantee a minimum fault efficiency on all instances with user-defined probability and confidence bounds. The method is applicable to industrial circuits.

VI. Acknowledgment

Parts of this work were supported by the German Research Foundation (DFG) under grants BE 1176/14-2, PO 1220/2-2, GRK 1103 and WU 245/11-1. We thank Tobias Schubert (University of Freiburg) for providing the SAT-solver *antom* [20] and support.

REFERENCES
