Incremental Computation of Delay Fault Detection Probability for Variation-Aware Test Generation

Wagner, Marcus; Wunderlich, Hans-Joachim

Proceedings of the 19th IEEE European Test Symposium (ETS'14) Paderborn, Germany, 26-30 May 2014

doi: http://dx.doi.org/10.1109/ETS.2014.6847805

Abstract: Large process variations in recent technology nodes present a major challenge for the timing analysis of digital integrated circuits. The optimization decisions of a statistical delay test generation method must therefore rely on the probability of detecting a target delay fault with the currently chosen test vector pairs. However, the huge number of probability evaluations in practical applications creates a large computational overhead. To address this issue, this paper presents the first incremental delay fault detection probability computation algorithm in the literature, which is suitable for the inner loop of automatic test pattern generation methods. Compared to Monte Carlo simulations of NXP benchmark circuits, the new method consistently shows a very large speedup and only a small approximation error.

Preprint

Incremental Computation of Delay Fault Detection Probability for Variation-Aware Test Generation

Marcus Wagner, Hans-Joachim Wunderlich
Institute of Computer Architecture and Computer Engineering
University of Stuttgart, Pfaffenwaldring 47, D-70569 Stuttgart, Germany

***Abstract*—Large process variations in recent technology nodes present a major challenge for the timing analysis of digital integrated circuits. The optimization decisions of a statistical delay test generation method must therefore rely on the probability of detecting a target delay fault with the currently chosen test vector pairs. However, the huge number of probability evaluations in practical applications creates a large computational overhead.**

To address this issue, this paper presents the first incremental delay fault detection probability computation algorithm in the literature, which is suitable for the inner loop of automatic test pattern generation methods. Compared to Monte Carlo simulations of NXP benchmark circuits, the new method consistently shows a very large speedup and only a small approximation error.

***Index Terms*—delay test, process variations, delay test quality**

I. Introduction

Process variations and high defect densities in recent technology nodes have emerged as new challenges for the timing analysis and the delay test of digital integrated circuits [1]–[3]. The uncertainty in the delays of all circuit components severely degrades the quality and reliability of all delay tests, leading to many test escapes [4]. More stringent path sensitization conditions can reduce the risk of test escapes, but these conditions are not satisfiable for a large number of paths [5].

The detection of a target delay fault by a single test vector pair occurs with a certain *delay fault detection probability*, which must be increased by the application of additional test vector pairs to minimize the risk of test escapes [6]. This probability depends on many delay test parameters [7]–[13], which must be optimized to find a suitable compromise between the final delay test quality and the test cost. Apart from the test set itself, these parameters include the test clock cycle time [9] and the masking of the combinational network outputs for faster-than-at-speed testing [10]. To guide the delay test parameter optimization, the delay fault detection probabilities for all affected target delay faults must be estimated after every delay test parameter modification. To avoid a huge computational overhead, the knowledge of previous probability estimation results must be exploited to incrementally compute the delay fault detection probability after every parameter modification.

The large runtime of the delay fault detection probability estimation has prompted some research in a related but simpler problem [11], which is obtained by neglecting all structural and spatial correlations. However, the resulting approximation error is acceptable only for tiny delay variations in the mature and classical manufacturing technology space [14].

On the other hand, Monte Carlo simulations naturally consider structural and spatial correlations even in very complex circuit models. To avoid simulating the entire test set after every removal or insertion of a test vector pair, [12] proposed to store the random delay values and test results for all iterations and test vector pairs. However, a Monte-Carlo simulation of the entire circuit is inefficient because only sufficiently long paths, which are also sensitized by the new test vector pair, can have a significant impact on the fault detection probability.

In [9], the authors presented a sensitization analysis in combination with a block based approach to compute the test vector pair delay distribution. While the block based approach could easily be extended for incremental computations, it also requires all gate and interconnect delays to have a normal distribution. However, this requirement is not satisfied for low power applications [15] and localized delay faults, which are often assumed to have an exponentially distributed fault size.

To alleviate these limitations, the authors of [13] introduced a path based approach which only requires normally distributed path delays. However, the efficiency of this approach deteriorates with the test set size, because a multivariate normal integral of increasingly high dimension must be approximated after every delay test parameter modification.

To address this problem, the paper at hand presents a new path based algorithm, which incrementally approximates the fault detection probability after every delay test parameter update. The algorithm minimizes the runtime for the probability computations by efficiently adapting an approximation of the test subset delay distribution, which is defined by the joint delay distribution of all test vector pairs in the test subset. The accuracy of the delay fault detection probability approximation only depends on the new values of the delay test parameters and is not affected by the delay test parameter update type or the order, in which these updates are applied.

The incremental analysis is based on a new extension of the Clark approximation [16], which introduces an efficient method to compute the covariance $\mathrm{Cov}(U, V)$ between the two maxima $U = \max(X_1, X_2)$ and $V = \max(X_3, X_4)$ of jointly normally distributed random variables $X_1, \ldots, X_4$.

The remainder of this paper is organized as follows. The proposed incremental computation of the delay fault detection probability is presented in section II. Section III describes the extension of the Clark maximum approximation method. The experimental results for several NXP benchmark circuits are shown in section IV and conclusions are drawn in section V.

II. Incremental Computation of the Delay Fault Detection Probability

It is assumed that the whole circuit is subject to delay variations, which may cause path delay faults by itself or in combination with a single gate delay fault. The subset of all test vector pairs in the test set, which are applicable for the detection of the gate delay fault, is referred to as *test subset*. Furthermore, the time between the application of a test vector pair to the combinational network inputs and the stabilization of all network outputs is called *test vector pair delay*. The following subsections focus on the insertion and the removal of a test vector pair from the test subset. The update of other delay test parameters is addressed in subsection II-E.

Following an update of the test subset, the proposed algorithm incrementally approximates the probability of capturing an incorrect logic value into at least one scan flip-flop. The incremental computation is based on the efficient approximation and adaptation of the *test subset delay distribution*, which is the joint delay distribution of all test vector pairs in the test subset. Only relatively few distribution parameter entries are affected by a delay test parameter update, which yields substantial savings in runtime. Moreover, the small dimension of the distribution allows the efficient approximation of the timing failure probability, which is a tight upper bound for the delay fault detection probability. A timing failure occurs if at least one combinational network output has not stabilized to its final logic value within the test clock cycle time.

All major steps of the algorithm are shown in Fig. 1. After the insertion of a new test vector pair, a sensitization analysis of this test vector pair is performed, which yields all complete transition paths along which transitions propagate to the combinational network outputs (A). The delay of the new test vector pair is then described by a normal distribution, which approximates the maximum of all transition path delays (B). This normal distribution subsequently extends the multivariate normal approximation of the test subset delay distribution by one dimension (C). Finally, the timing failure probability approximation is obtained by numerical integration over the resulting multivariate normal distribution (D).

After the removal of a test vector pair, the corresponding parameter entries of the multivariate normal distribution are deleted (C). A numerical integration over the reduced distribution yields the timing failure probability approximation (D).

The following subsections present a detailed description of all major steps of the algorithm.

A. Sensitization Analysis of New Test Vector Pair

The sensitization analysis is based on a single pass event-driven timing simulation of the new test vector pair using only nominal delay values [9], [13]. A set of complete transition paths is subsequently identified by tracing all transitions at the outputs of the combinational networks back to the inputs of the networks. The delay of a transition path is the sum of the delays of all circuit components along the path, where each component delay is defined as the sum of the nominal delay value, the inter-die and the intra-die part of the delay variation.

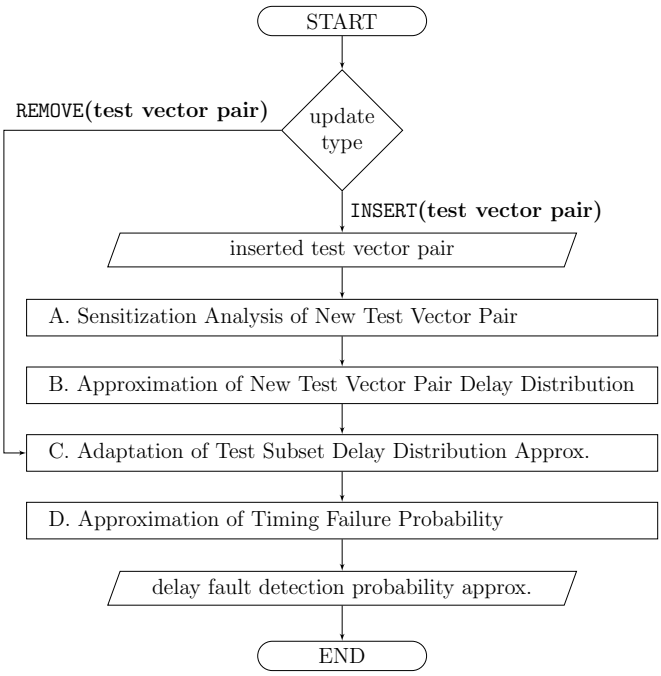

Fig. 1. Flowchart after the insertion or the removal of a test vector pair.

Any complete transition path, which terminates at an unmasked combinational network output, is called long transition path if its probability of exceeding the test clock cycle time is above a user defined threshold. In the following, $X_1, X_2, \ldots, X_n$ will denote the delays of the resulting n long transition paths.

The sum of all component delays along a complete transition path converges rapidly to a normal distribution for most practical models of correlation, especially if the distributions of the gate delays are close to a normal distribution [17]. This property is a major advantage of the path based approach and also justifies the assumption, that the joint distribution of all long transition path delays can accurately be described by a multivariate normal distribution.

B. Approximation of New Test Vector Pair Delay Distribution

The delay of the new test vector pair, which extends the current test subset from size $k-1$ to size k, is defined as

$$Y_k = \max(X_1, \ldots, X_n), \tag{1}$$

and approximated using the extended Clark approximation method, which is presented in section III.

The additional flexibility of the extension allows the formation of balanced tree like dataflow graphs (Fig. 2b), in which only those maximum operations, which lie on the path to the final result, must be recomputed if modifications to the long transition path delays are made.

On the other hand, the Clark approximation always forms a single chain of maximum operations (Fig. 2a), so that on average half of the maximum operation results depend on the value of a single variable.

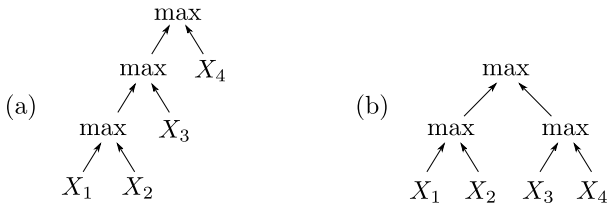

Fig. 2. Data flow graphs for the approximation of $\max(X_1, X_2, X_3, X_4)$ using (a) the Clark approximation and (b) the extended Clark approximation.

C. Adaptation of Test Subset Delay Distribution Approx.

The joint distribution of all test vector pair delays is called test subset delay distribution. For greater efficiency, this distribution is approximated by a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

After the insertion of a new test vector pair, the length of the mean vector and the dimension of the covariance matrix increases from $k-1$ to k. The values of the new entry of $\boldsymbol{\mu}$ and the new diagonal entry of $\boldsymbol{\Sigma}$ equal the mean and the variance of the new test vector pair delay Y_k. The remaining new entries $\text{Cov}(Y_k, Y_1), \ldots, \text{Cov}(Y_k, Y_{k-1})$ of $\boldsymbol{\Sigma}$ represent the covariances between Y_k and the delays $Y_1, \ldots, Y_{k-1}$ of the remaining test vector pairs in the test subset.

The Clark approximation does not provide a function to compute the covariance $\text{Cov}(Y_k, Y_i)$ between the two test vector pair delays Y_k and Y_i, with $i \in \mathbb{N}^+, i < k$. Therefore, the covariance computation method of the extended Clark approximation is required, which is presented in section III. Similar to the approximation of the maximum $\max(Y_k, Y_i)$, the computation of $\text{Cov}(Y_k, Y_i)$ starts from the long transition path delays and proceeds level by level until Y_k and Y_i have been reached. However, most of the intermediate results are still available from the approximation of Y_k and Y_i and only some additional covariances must be computed.

After the removal of the ith test vector pair, the dimension of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ reduces from k to $k-1$ by deleting all entries, which correspond to the ith test vector pair delay Y_i.

The time for the extension of the distribution increases linearly with the test subset size, assuming that all test vector pairs sensitize the same number of long transition paths.

D. Approximation of Timing Failure Probability

Finally, this step approximates the probability of a timing failure, which occurs if at least one test vector pair delay $Y_i \in \{Y_1, \ldots, Y_k\}$ exceeds the test clock cycle time T_{clk}. The timing failure probability is computed as

$$\Psi = 1 - \text{Pr}(Y_1 \leq T_{clk}, \ldots, Y_k \leq T_{clk}), \tag{2}$$

and approximated by a multivariate normal integral over the test subset delay distribution approximation

$$\Psi \approx 1 - \int_{-\infty}^{T_{clk}} \cdots \int_{-\infty}^{T_{clk}} \phi(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) dy_1 ... dy_k, \tag{3}$$

where $\phi(\cdot)$ denotes the probability density function of the multivariate normal distribution.

The probability Ψ is efficiently approximated using recent numerical integration algorithms [18], [19]. To ensure the positive definiteness of $\boldsymbol{\Sigma}$ for linearly dependent test vector pair delays and approximation errors, all diagonal entries of $\boldsymbol{\Sigma}$ are multiplied with a very small constant greater than one.

E. Update of other Delay Test Parameters

The set of complete transition paths, which was stored for every test vector pair in the test subset, can be reused because it does not depend on the test clock cycle time T_{clk} or the masking of the transition path outputs. However, some complete transition paths may become long transition paths if T_{clk} is reduced or some combinational network outputs are unmasked. Likewise, some long transition paths may be removed if outputs are masked or T_{clk} is increased, although the latter case may have little impact on the test subset delay distribution approximation.

If the masking or unmasking of transition path outputs changes the set of long transition paths, the algorithm proceeds as if a new test vector pair had been added to the test subset. However, instead of extending the mean vector and the covariance matrix, all entries corresponding to the modified test vector pair delay are replaced.

III. Extended Clark Approximation Method

Let $U = \max(X_1, X_2)$ and $V = \max(X_3, X_4)$ denote two maxima of jointly normal random variables. The extended Clark approximation uses the formulas for the exact moments of U and V, presented by Clark [16], but extends the flexibility of the approximation by providing a method for the accurate computation of the covariance $\text{Cov}(U, V)$.

The classical Clark approximation computes $\text{Cov}(U, X_3)$ and $\text{Cov}(U, X_4)$ using the formula $\text{Cov}(\max(Y_1, Y_2), Y_3) = \text{Pr}(Y_1 > Y_2)\text{Cov}(Y_1, Y_3) + \text{Pr}(Y_1 \leq Y_2)\text{Cov}(Y_2, Y_3)$. However, the new random vector (X_3, X_4, U) does not have a trivariate normal distribution. As a consequence, this formula does not compute the accurate value of $\text{Cov}(\max(X_3, X_4), U)$.

This extension computes the accurate covariance from

$$\text{Cov}(U, V) = \text{E}[UV] - \text{E}[U]\text{E}[V], \tag{4}$$

where the first raw moments $\text{E}[U]$ and $\text{E}[V]$ are available from the Clark approximation [16]. The cross-moment $\text{E}[UV]$ is

$$\begin{aligned}\text{E}[UV] = &\text{E}[X_1X_3 | X_1 > X_2, X_3 > X_4]\text{Pr}(X_1 > X_2, X_3 > X_4)\\ &+ \text{E}[X_1X_4 | X_1 > X_2, X_3 \leq X_4]\text{Pr}(X_1 > X_2, X_3 \leq X_4)\\ &+ \text{E}[X_2X_3 | X_1 \leq X_2, X_3 > X_4]\text{Pr}(X_1 \leq X_2, X_3 > X_4)\\ &+ \text{E}[X_2X_4 | X_1 \leq X_2, X_3 \leq X_4]\text{Pr}(X_1 \leq X_2, X_3 \leq X_4).\end{aligned}$$

To simplify the solution for the above expression, let the random vector $\boldsymbol{W}$ be a linear transformation of $\boldsymbol{X}$, defined as

$$(W_1, W_2, W_3, W_4) = (X_1, X_1 - X_2, X_3, X_3 - X_4),$$

The transformed vector $\boldsymbol{W}$ has a multivariate normal distribution with mean vector

$$\boldsymbol{\theta} = (\mu_1, \mu_1 - \mu_2, \mu_3, \mu_3 - \mu_4)^T$$

and covariance matrix

$$\boldsymbol{A}=\begin{bmatrix} \sigma_{1,1} & \sigma_{1,1}-\sigma_{1,2} & \sigma_{1,3} & \sigma_{1,3}-\sigma_{1,4} \\ & \sigma_{1,1}+\sigma_{2,2}-2\sigma_{1,2} & \sigma_{1,3}-\sigma_{2,3} & \sigma_{1,3}-\sigma_{1,4}-\sigma_{2,3}+\sigma_{2,4} \\ & & \sigma_{3,3} & \sigma_{3,3}-\sigma_{3,4} \\ & & & \sigma_{3,3}+\sigma_{4,4}-2\sigma_{3,4} \end{bmatrix},$$

where μ_i denotes the mean of X_i and $\sigma_{i,j}$ denotes the covariance $\mathrm{Cov}(X_i, X_j)$ with $i, j \in \{1, \ldots, 4\}$. Only the upper triangular part of the covariance matrix $\boldsymbol{A} = \{a_{i,j}\}$ is presented here due to spacial limitations. The lower triangular part is defined by symmetry $\boldsymbol{A} = \boldsymbol{A}^T$.

Suppose that the component W_i has mean θ_i, standard deviation $a_i = \sqrt{a_{i,i}}$ and correlation $\rho_{i,j} = a_{i,j}/(a_i a_j)$ with another component W_j. By introducing the following notations

$$\alpha_i = -\theta_i / a_i \tag{5}$$

$$\beta_2 = (\alpha_4 - \rho_{2,4}\alpha_2)/\sqrt{1-\rho_{2,4}^2} \tag{6}$$

$$\beta_4 = (\alpha_2 - \rho_{2,4}\alpha_4)/\sqrt{1-\rho_{2,4}^2} \tag{7}$$

and using the property

$$\mathrm{E}[W_i W_j] = a_{i,j} + \theta_i \theta_j, \tag{8}$$

the formulas for the truncated multivariate normal distribution, presented in [20] and [21], can be simplified to

$$\begin{aligned} \mathrm{E}[UV] = {} & \mathrm{E}[W_1 W_3] \\ & - \mathrm{E}[W_1 W_4]\Phi_1(\alpha_4) + (\theta_1 - \theta_2 \Phi_1(\beta_4)) a_4 \phi_1(\alpha_4) \\ & - \mathrm{E}[W_2 W_3]\Phi_1(\alpha_2) + (\theta_3 - \theta_4 \Phi_1(\beta_2)) a_2 \phi_1(\alpha_2) \\ & + \mathrm{E}[W_2 W_4]\Phi_2(\alpha_2, \alpha_4; \rho_{2,4}) \\ & + (1-\rho_{2,4}^2) a_2 a_4 \phi_2(\alpha_2, \alpha_4; \rho_{2,4}) \end{aligned} \tag{9}$$

with

$$\mathrm{E}[U] = \theta_1 - \theta_2 \Phi_1(\alpha_2) + a_2 \phi_1(\alpha_2) \tag{10}$$

$$\mathrm{E}[V] = \theta_3 - \theta_4 \Phi_1(\alpha_4) + a_4 \phi_1(\alpha_4), \tag{11}$$

where ϕ_k denotes the probability density function and Φ_k denotes the cumulative distribution function of the k-variate standard normal distribution. Hence, the computation of the covariance $\mathrm{Cov}(\max(X_1, X_2), \max(X_3, X_4))$ involves several univariate normal integrals but only one bivariate normal integral, which can be evaluated using a fast double precision approximation method, recently presented by Genz [19].

For the special case $\rho_{2,4} = 0$ the computation simplifies to

$$\mathrm{Cov}(U, V) = a_{1,3} - a_{1,4}\Phi_1(\alpha_4) - a_{2,3}\Phi_1(\alpha_2) \tag{12}$$

Finally, if either U or V has a normal distribution, the proposed extension simplifies to the special case considered by the Clark approximation formula for $\mathrm{Cov}(\max(Y_1, Y_2), Y_3)$.

IV. Experimental Results

Several NXP benchmark circuit were speed-optimized using a commercial synthesis tool and mapped to the NanGate 45nm Open Cell Library [22]. The gate model for this approach conforms to the Verilog HDL [23] standard, but instead of real numbers, every delay value X of a gate has a normal distribution with variance $\mathrm{var}(X) = (c_v \mathrm{E}[X])^2$ and mean $\mathrm{E}[X]$ equal to the nominal delay value from the standard delay format description of the synthesized circuit. Based on predictions for future process technology nodes [14], a variation coefficient of $c_v = 0.25$ was selected. Interconnect delays and spatial correlations would require the analysis of full chip layouts, which had not been designed to avoid an unnecessary complex experimental setup. The test clock cycle time T_{clk} was determined, such that delay variations would not cause timing failures in 95% of the fault free chips.

For every circuit, a set of 20000 randomly chosen single gate delay faults was created. Next, a test subset for each delay fault was generated by a suitable example ATPG algorithm. At first, the 1000 longest paths through the fault site were found by a commercial static timing analysis tool. Thereby, the number of paths terminating at the same combinational network output was limited to 100. The resulting set of paths was sensitized with a commercial ATPG tool. Finally, the delay fault size was chosen, such that the delay of the longest complete transition path through the fault site was equal T_{clk}.

To evaluate the incremental probability computation, the example ATPG algorithm extended an initially empty test subset by inserting single test vector pairs in decreasing order of the greatest complete transition path delay. The removal of a randomly chosen test vector pair only requires a rerun of the numerical integration, which is also presented in Table I. Only the results of those update operations are shown, which resulted in a test subset of size 1, 5, 10 or 20 test vector pairs. Alternative delay test parameter updates are evaluated by the results of the corresponding individual steps of the algorithm.

The reference fault detection probability P was computed by crude full-circuit Monte Carlo simulations of 10^4 iterations, which results in a sufficiently small estimated upper bound of 0.0098 for the half length of the 95% confidence interval. A fault is said to be detected if during one iteration a combinational network output has an unexpected logic value after the test clock cycle time T_{clk}. The runtime of a Monte Carlo simulation is dominated by the large number of random delay values, which are required for each iteration. The Verilog HDL standard distinguishes between different pin-to-pin, asymmetric rising/falling and conditional path delays. Therefore up to eight delay values are required for a two-input gate. The random number generation utilizes high-performance implementations of the Box-Muller transform and the Mersenne Twister pseudo-random number generator from the Intel Math Kernel Library [24]. All programs were implemented in C, C++ and FORTRAN and executed on Intel Core i7-2600K processor workstations with 20GB RAM.

Column (1) gives the name of the NXP benchmark circuit, and column (2) shows the number of test vector pairs $|\Theta|$ in the test subsets Θ after the insertion or removal of a test vector pair. All following columns (3)-(14) present average values over all randomly chosen delay faults in each circuit.

The average runtime of the crude Monte Carlo simulation is shown in column (3). The following columns (4)-(11) and (12)-(14) present the results for the insertion and the removal of a test vector pair, respectively.

NXP circuit	$\lvert\Theta\rvert$	Reference	**INSERT(test vector pair)**								**REMOVE(test vector pair)**		
		crude MC	sensitization analysis (A)			dist. ext. (B,C)		Ψ (D)	*Complete Update*		Ψ (D)	*Complete Update*	
		t_{MC} [ms]	$\lvert\Pi\rvert$	$\lvert\delta\rvert$ 10^{-2}	t_{SA} [ms]	$\lvert\varepsilon\rvert$ 10^{-2}	t_{DE} [ms]	t_{NI} [ms]	$\lvert\epsilon\rvert$ 10^{-2}	**Speedup**	t_{NI} [ms]	$\lvert\epsilon\rvert$ 10^{-2}	**Speedup**
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
p35k	1	28 957	6.80	4.89	4.67	0.21	0.10	< 0.01	4.88	**6 071**	< 0.01	4.88	$>$ **10^7**
	5	59 490	30.28	3.74	3.75	0.76	0.22	1.27	3.70	**11 343**	1.27	3.70	**46 891**
	10	92 331	57.88	2.98	3.73	0.72	0.36	12.65	2.93	**5 517**	12.65	2.93	**7 301**
	20	156 858	109.20	2.16	3.71	0.59	0.61	24.86	2.13	**5 375**	24.86	2.13	**6 309**
p45k	1	29 753	3.32	3.45	5.86	0.12	0.08	< 0.01	3.50	**5 006**	< 0.01	3.50	$>$ **10^7**
	5	62 473	13.33	3.44	5.09	0.65	0.17	1.16	3.72	**9 736**	1.16	3.72	**53 966**
	10	99 064	25.63	3.06	5.11	0.77	0.42	10.20	3.39	**6 298**	10.20	3.39	**9 713**
	20	172 082	45.46	2.42	5.11	0.80	0.45	18.06	2.73	**7 285**	18.06	2.73	**9 530**
p77k	1	49 087	3.64	2.07	8.94	0.09	0.08	< 0.01	2.11	**5 443**	< 0.01	2.11	$>$ **10^7**
	5	115 037	13.82	2.17	8.89	0.52	0.21	1.05	2.46	**11 334**	1.05	2.46	**109 504**
	10	190 105	25.24	1.93	8.76	0.69	0.39	8.76	2.32	**10 614**	8.76	2.32	**21 707**
	20	345 268	44.27	1.96	8.77	0.89	0.69	12.60	2.51	**15 650**	12.60	2.51	**27 405**
p78k	1	184 414	3.46	6.70	19.71	0.17	0.06	< 0.01	6.77	**9 325**	< 0.01	6.77	$>$ **10^8**
	5	350 785	15.60	4.65	20.11	0.56	0.09	1.37	4.66	**16 255**	1.37	4.66	**255 495**
	10	557 799	28.76	3.47	20.03	0.59	0.13	11.99	3.46	**17 349**	11.99	3.46	**46 518**
	20	971 327	50.74	2.73	19.76	0.60	0.20	22.00	2.69	**23 147**	22.00	2.69	**44 159**
p81k	1	295 960	2.68	3.95	13.14	0.06	0.06	< 0.01	3.97	**22 424**	< 0.01	3.97	$>$ **10^8**
	5	407 829	10.65	2.99	13.05	0.39	0.12	1.18	3.06	**28 411**	1.18	3.06	**344 233**
	10	539 772	19.39	2.42	13.04	0.45	0.20	11.10	2.50	**22 175**	11.10	2.50	**48 618**
	20	805 824	33.73	1.96	13.05	0.47	0.37	19.92	2.05	**24 171**	19.92	2.05	**40 453**
p100k	1	147 445	3.15	3.87	13.32	0.10	0.07	< 0.01	3.90	**11 015**	< 0.01	3.90	$>$ **10^8**
	5	213 606	12.50	3.35	13.24	0.49	0.15	1.14	3.55	**14 698**	1.14	3.55	**187 609**
	10	302 103	23.10	2.89	13.20	0.57	0.22	9.51	3.13	**13 178**	9.51	3.13	**31 779**
	20	540 130	44.52	2.74	13.30	0.61	0.44	16.68	3.01	**17 752**	16.68	3.01	**32 373**
p267k	1	560 594	4.21	2.97	35.90	0.13	0.10	< 0.01	3.00	**15 574**	< 0.01	3.00	$>$ **10^8**
	5	767 491	16.13	2.66	35.95	0.60	0.32	1.20	2.87	**20 485**	1.20	2.87	**639 586**
	10	1 007 033	29.30	2.18	35.98	0.70	0.65	10.80	2.45	**21 229**	10.80	2.45	**93 203**
	20	1 476 696	53.29	1.73	36.00	0.72	1.36	19.77	2.03	**25 846**	19.77	2.03	**74 678**
p330k	1	773 594	7.61	4.32	51.88	0.22	0.29	< 0.01	4.42	**14 827**	< 0.01	4.42	$>$ **10^8**
	5	1 105 557	31.61	3.29	51.89	0.74	1.59	1.21	3.63	**20 213**	1.21	3.63	**911 752**
	10	1 503 512	56.47	2.69	51.77	0.81	3.76	10.30	3.05	**22 840**	10.30	3.05	**145 940**
	20	2 299 519	104.69	2.18	51.72	0.81	7.25	18.14	2.51	**29 824**	18.14	2.51	**126 773**

TABLE I
RUNTIME AND ABSOLUTE ERROR OF INCREMENTAL ANALYSIS AFTER INSERTION OR REMOVAL OF A TEST VECTOR PAIR (NXP BENCHMARK CIRCUITS)

Columns (4)-(6) present the results of the sensitization analysis. A complete transition path with delay X was considered long, if $\mathrm{E}[X] + 3\sqrt{\mathrm{var}(X)} \geq T_{clk}$. The average number of long transition paths $|\Pi|$, identified for all test vector pairs in the test subset Θ, is shown in column (4). The approximation error δ of the sensitization analysis is estimated by

$$\delta = \big(1 - \Pr\big(X_1 \leq T_{clk}, \ldots, X_{|\Pi|} \leq T_{clk}\big)\big) - P$$

and the average absolute value of δ is presented in column (5). In general, the sensitization of a long transition path is restricted to a subset of all possible delay realizations, due to hazards and variations in the arrival times of the transitions at the off-path inputs. Hence, the nominal delay value based sensitization analysis might identify a set of long transition paths, which are not representative for the arrival time of the last transition at the combinational network outputs. However, the resulting error δ becomes very small as the number of long transition paths increases. The average runtime of the sensitization analysis is shown in column (6).

The following steps of the algorithm were omitted, unless at least one long transition path was found. Columns (7) and (8) show the results from the extended Clark approximation for the computation of the test vector pair delay distribution and the extension of the multivariate normal distribution. The extended Clark approximation error is estimated by

$$\begin{aligned}\varepsilon = &(1 - \Pr(\, Y_1 \leq T_{clk}, \ldots, Y_{|\Theta|} \leq T_{clk})) \\ &- (1 - \Pr(X_1 \leq T_{clk}, \ldots, X_{|\Pi|} \leq T_{clk})).\end{aligned}$$

The average absolute error, shown in column (7), demonstrates the superior accuracy of the extended Clark approximation for the proposed test subset delay distribution extension. However, experiments with more general statistical timing analysis problems have shown its high susceptibility to skewed intermediate results, which resulted in high approximation errors.

The average runtime for the delay distribution extension, shown in column (8), is very small and increases almost linear with the size of the test subset.

The test subsets for p330k contains some test vector pairs, which sensitize thousands of long transition paths, with many paths being sensitized by multiple test vector pairs of the same test subset. To minimize the runtime of this step, it would be possible to ignore all recurrences of a long transition path delay without affecting the timing failure probability, unless this path delay is also deleted by a subsequent removal of a test vector pair or the masking of a transition path output.

The timing failure probability Ψ was approximated using the FORTRAN routines `BVND`, `TVTL` and `MVNDST`, which were developed by Genz [18] [19]. The average runtime, shown in column (9) and (12), increases rapidly for small and slowly for larger dimensions $|\Theta| \geq 10$. The parameters of `MVNDST` were chosen, such that the estimated relative error was below $5 \cdot 10^{-3}$ while using at most 10^5 points. These routines were also used for the presentation of the approximation errors δ and ε of the individual steps.

Columns (10)-(11) and (13)-(14) present the average runtime and the accuracy of the complete algorithm after the insertion or the removal of a test vector pair, respectively.

The approximation error of the final result is estimated by $\epsilon = \Psi - P$, where Ψ denotes the approximation of the timing failure probability and P denotes the fault detection probability. The average absolute incremental analysis error, shown in columns (10) and (13), is small and mainly determined by the accuracy of the sensitization analysis.

Columns (11) and (14) highlight the very large speedup of the proposed algorithm. Compared to the previous approach [13], the new algorithm is between 1.2 and 26.5 times faster. The speedup is defined by the ratio of the average runtime of the Monte Carlo simulation (column (3)) and the average runtime of the incremental computation over all delay faults.

Similary, the speedup for other delay test parameter updates can be computed from the sum of the average runtimes of the individual steps, which are required for the particular update. For example, the runtime after the reduction of the test clock cycle time or the modification of the combinational network output masks equals the sum of columns (8) and (9).

V. Conclusion

Delay variations in recent technology nodes reduce the quality and reliability of all delay tests. To find a suitable trade-off between the statistical delay test quality and the test cost, statistical test generation methods must optimize all delay test parameters based on the probability of detecting a delay fault. To efficiently evaluate the huge number of probability estimations, an efficient incremental algorithm has been presented, which is suitable for the inner loop of automatic test pattern generation methods. The approach was compared with results of extensive Monte Carlo simulations and has consistently shown a very large speedup with only a small loss of accuracy.

VI. Acknowledgement

This work has been supported by the German Research Foundation (DFG) under grant WU245/9-1 (Intesys). The authors would like to thank NXP for providing the industrial circuits.

References

[1] B. Becker, S. Hellebrand, I. Polian, B. Straube, W. Vermeiren, and H.-J. Wunderlich, "Massive statistical process variations: A grand challenge for testing nanoelectronic circuits," in *Int. Conf. on Dependable Systems and Networks Workshops (DSN-W)*, Jun. 2010, pp. 95–100.

[2] S. Kundu and A. Sreedhar, "Modeling manufacturing process variation for design and test," in *Proc. Design, Automation and Test in Europe (DATE)*, Mar. 2011, pp. 1–6.

[3] E. Yilmaz, S. Ozev, O. Sinanoglu, and P. Maxwell, "Adaptive testing: Conquering process variations," in *IEEE European Test Symp. (ETS)*, May 2012, pp. 1–6.

[4] U. Ingelsson, B. Al-Hashimi, S. Khursheed, S. Reddy, and P. Harrod, "Process variation-aware test for resistive bridges," *IEEE Trans. Computer-Aided Design*, vol. 28, no. 8, pp. 1269–1274, Aug. 2009.

[5] C. Lin and S. Reddy, "On delay fault testing in logic circuits," *IEEE Trans. Computer-Aided Design*, vol. 6, no. 5, pp. 694–703, Sep. 1987.

[6] M. Favalli and M. Dalpasso, "High quality test vectors for bridging faults in the presence of IC's parameters variations," in *Int. Symp. on Defect and Fault-Tolerance in VLSI Syst. (DFT)*, Sep. 2007, pp. 448–456.

[7] H. Hao and E. McCluskey, "Very-low-voltage testing for weak CMOS logic ICs," in *Proc. Int. Test Conf. (ITC)*, Oct. 1993, pp. 275–284.

[8] C.-W. Tseng, E. J. McCluskey, X. Shao, J. Wu, and D. M. Wu, "Cold delay defect screening," in *Proc. VLSI Test Symp. (VTS)*. IEEE Comput. Soc, May 2000, pp. 183–188.

[9] B. Lee, L.-C. Wang, and M. S. Abadir, "Reducing pattern delay variations for screening frequency dependent defects," in *Proc. VLSI Test Symp. (VTS)*, May 2005, pp. 153–160.

[10] B. Lee, H. Li, L.-C. Wang, and M. S. Abadir, "Hazard-aware statistical timing simulation and its applications in screening frequency-dependent defects," in *Proc. Int. Test Conf. (ITC)*, Nov. 2005, pp. 91–100.

[11] Z. Wang and D. M. Walker, "Compact delay test generation with a realistic low cost fault coverage metric," in *Proc. VLSI Test Symp. (VTS)*, May 2009, pp. 59–64.

[12] A. Czutro, M. E. Imhof, J. Jiang, A. Mumtaz, M. Sauer, B. Becker, I. Polian, and H.-J. Wunderlich, "Variation-aware fault grading," in *Proc. IEEE Asian Test Symp. (ATS)*, Nov. 2012, pp. 344–349.

[13] M. Wagner and H.-J. Wunderlich, "Efficient variation-aware statistical dynamic timing analysis for delay test applications," in *Proc. Design, Automation and Test in Europe (DATE)*, Mar. 2013, pp. 1–6.

[14] Y. Ye, S. Gummalla, C.-C. Wang, C. Chakrabarti, and Y. Cao, "Random variability modeling and its impact on scaled CMOS circuits," *Journal of Computational Electronics*, vol. 9, no. 3-4, pp. 108–113, Oct. 2010.

[15] S. Hanson, D. Blaauw, and D. Sylvester, "Analysis and mitigation of variability in subthreshold design," in *Proc. Int. Symp. on Low Power Electronics and Design (ISLPED)*, Aug. 2005, pp. 20–25.

[16] C. E. Clark, "The greatest of a finite set of random variables," *Operations Research*, vol. 9, no. 2, pp. 145–162, Mar. 1961.

[17] S. Louhichi, "Rates of convergence in the CLT for some weakly dependent random variables," *Theory of Probability and its Applications*, vol. 46, no. 2, pp. 297–315, Jul. 2002.

[18] A. Genz, "Numerical computation of multivariate normal probabilities," *Journal of Computational and Graphical Statistics*, vol. 1, no. 2, pp. 141–149, Jun. 1992.

[19] A. Genz, "Numerical computation of rectangular bivariate and trivariate normal and t probabilities," *Statistics and Computing*, vol. 14, no. 3, pp. 251–260, Aug. 2004.

[20] Z. W. Birnbaum and P. L. Meyer, "On the effect of truncation in some or all coordinates of a multinormal population," Washington University and Seattle Laboratory of Statistical Research, Tech. Rep., Nov. 1951.

[21] G. Tallis, "The moment generating function of the truncated multinormal distribution," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 23, no. 1, pp. 223–229, 1961.

[22] "Nangate 45nm open cell library," Aug. 2011. Available: http://si2.org/openeda.si2.org/projects/nangatelib/

[23] IEEE Standards Department, "Verilog hardware description language," *IEEE Std 1364-2005*, Apr. 2006.

[24] Intel Corporation, "Intel math kernel library reference manual," pp. 1–3326, 2013.