PAPER

LAPS: Layout-Aware Path Selection for Post-Silicon Timing Characterization

Yu HU[†,††a)], *Member*, Jing YE[††b)], Zhiping SHI[†,†††c)], *and* Xiaowei LI[††d)], *Nonmembers*

SUMMARY Process variation has become prominent in the advanced CMOS technology, making the timing of fabricated circuits more uncertain. In this paper, we propose a Layout-Aware Path Selection (LAPS) technique to accurately estimate the circuit timing variation from a small set of paths. Three features of paths are considered during the path selection. Experiments conducted on benchmark circuits with process variation simulated with VARIUS show that, by selecting only hundreds of paths, the fitting errors of timing distribution are kept below 5.3% when both spatial correlated and spatial uncorrelated process variations exist.

key words: *process variation, timing variation, sample, path selection, least square*

1. Introduction

As the feature size in advanced VLSI technology continuous to shrink, process variation produces more uncertainty in circuit timing behavior. Accurate timing characterization plays an important role in a variety of applications [1]–[6], such as statistical timing analysis, post-silicon tuning, post-silicon reliability analysis, and IC identification.

In general, process variation can be divided into two categories: correlated systematical variation and uncorrelated random variation. The correlated systematical variation tends to affect the closely placed gates or routed wires in a similar manner, making the gates and wires more likely to have similar process variations than those placed far apart [7], which is called the spatial correlation. On the other hand, the uncorrelated random variation means the random variation involves spatial uncorrelation. Transistor timing is heavily impacted by both the correlated systematical variation and uncorrelated random variation [8].

Figure 1-(1) shows the distribution of gate timing variations under spatially correlated process variations, generated by the VARIUS tool [9], where the deeper color represents higher timing variation. It can be seen that the timing variations of nearby gates are similar. Wires also have this kind of correlation [10], [11]. Therefore, timing characterization methods [12]–[21] usually first divide a circuit layout into grids to reduce the sample complexity, as shown in Fig. 1-(2) and Fig. 1-(3). When the grid is small enough, gates in a grid are considered to have the same timing variation. Then, by sampling the representative gates in grids or paths across grids, the timing variation of each grid can be profiled. The ways of sampling the timing information include (1) invasive measurement and (2) non-invasive testing.

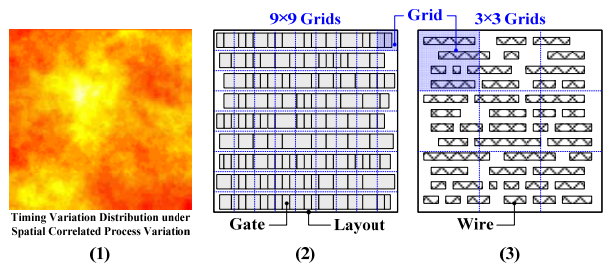


Fig. 1 Examples of variations, layouts and grids.

For invasive measurement, on-chip monitors such as ring oscillator [12], NMOS/PMOS transistor chain [13] and slew-rate monitor [14] are inserted into grids to measure their timing. The advantage is that the timing information of grids can be directly measured. However, the more grids the layout is divided into, the higher hardware overhead it costs. Recent works [15]–[17] tried to solve the hardware overhead issue, but the accuracy of timing characterization is still more or less sacrificed.

Non-invasive testing leverages the paths in circuits to get the timing information. As the paths normally go across several grids, the path delay information must be post-processed by compressed sensing or least square techniques to obtain the timing of each grid [18]–[21]. Obviously, the accuracy of timing characterization is heavily dependent on the sampled paths. Path selection strategies have been well studied in delay fault testing works, but critical paths with long delays are the main concern. Without considering the distribution of paths on the circuit layout, these paths hardly reflect the whole timing distribution of the circuit.

In this paper, we propose a path selection method specifically for non-invasive timing characterization. This paper is an extension of our prior work in [21]. The proposed method has the following contributions:

Manuscript received May 2, 2016.
Manuscript revised September 21, 2016.
Manuscript publicized October 25, 2016.

[†]The authors are with Beijing Advanced Innovation Center for Imaging Technology, Capital Normal University, Beijng, 100048, China.
[††]The authors are with the State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, China.
[†††]The author is with College of Information Engineering, Capital Normal University, Beijng, 100048, China.
a) E-mail: huyu@ict.ac.cn
b) E-mail: yejing@ict.ac.cn
c) E-mail: shizp@cnu.edu.cn (Corresponding author)
d) E-mail: lxw@ict.ac.cn

DOI: 10.1587/transinf.2016EDP7184

- Sufficiency and uniformity of the sampled data for fitting timing variations are analyzed;
- A layout-aware path selection method named LAPS is proposed to sample paths for estimating the circuit timing variation.

The rest of the paper is organized as follows. Section 2 reviews the problem formulation of estimating the timing variation based on the sampled path delays. Section 3 analyzes the sufficiency and the uniformity of the sampled data, while Sect. 4 proposes the LAPS method. The experimental results are given in Sect. 5, followed by the conclusion section.

2. Problem Formulation

In this section, we will briefly review the problem formulation of estimating timing variation with sampled path delays [18], [21]. Taking the timing variation into consideration, the delay of a gate or a wire is represented by:

$$D(K) = D^{nom}(K) \times v(K) \tag{1}$$

where K represents a gate or a wire. $D(K)$ represents the actual delay of K. $D^{nom}(K)$ represents the nominal delay of K. $v(K)$ represents the timing variation of K.

If a path is a single-sensitized path, then its delay can be approximately considered as the accumulated delay of the gates and wires along the path [18]–[22]. Please notice that, different from the path selection in delay testing, we have no requirement on path length, so either long path or short path is feasible in our work.

Since the gates or wires nearby at layout have similar timing variations, their timing variations in the same layout grid are considered to be similar, especially when the grid is small. This approximation will introduce some fitting errors due to the uncorrelated random process variations, and we will show the fitting errors are still acceptable in the experiment section.

Because long wire can cross multiple grids, we divide the long wire into several segments so that every segment belongs to only one grid. The total delay of the wire is then considered as the sum of its segments. Different metal layers of a chip may have different timing variation distributions, so each layer will be accordingly divided into optimal number of grids.

For example, in Fig. 2, P_0 is a single-sensitized path when $e = 1$, $f = 0$, and $g = 0$. The gates H and I are both in the grid G_1, so they are approximately considered with the same timing variation $v(H) = v(I) = v(G_1)$; J is in G_2 with $v(J) = v(G_2)$. The wire b belongs to two grids so it is divided into b_1 and b_2; a and b_1 are both in G_3 with $v(a) = v(b_1) = v(G_3)$; b_2, c, and d are all in G_4 with $v(b_2) = v(c) = v(d) = v(G_4)$. So the actual delay of P_0 is:

$$\begin{aligned} D(P_0) = {} & [D^{nom}(H) + D^{nom}(I)] \times v(G_1) \\ & + D^{nom}(J) \times v(G_2) \\ & + [D^{nom}(a) + D^{nom}(b_1)] \times v(G_3) \\ & + [D^{nom}(b_2) + D^{nom}(c) + D^{nom}(d)] \times v(G_4) \end{aligned} \tag{2}$$

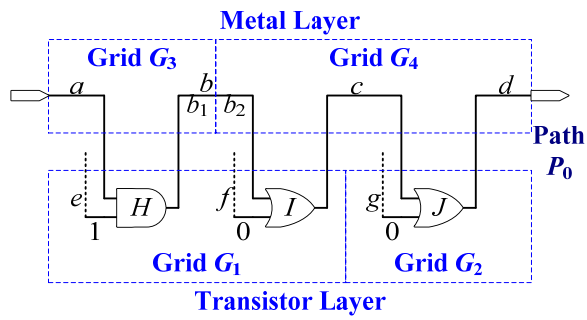


Fig. 2 An example of a single-sensitized path.

where $D^{nom}(H)$ denotes the nominal delay of the gate H.

Therefore, if the number of single-sensitized paths is N_{SSP}, then the timing characterization problem can be formulated to the following equation set:

$$D = \Pi \times V \qquad D = \begin{bmatrix} D(P_1) \\ D(P_2) \\ \vdots \\ D(P_{N_{SSP}}) \end{bmatrix} \qquad V = \begin{bmatrix} v(G_1) \\ v(G_2) \\ \vdots \\ v(G_{N_{Grid}}) \end{bmatrix}$$

$$\Pi = \begin{bmatrix} \pi_{11} & \pi_{12} & \cdots & \pi_{1N_{Grid}} \\ \pi_{21} & \pi_{22} & \cdots & \pi_{2N_{Grid}} \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{N_{SSP}1} & \pi_{N_{SSP}2} & \cdots & \pi_{N_{SSP}N_{Grid}} \end{bmatrix} \qquad \begin{aligned} \pi_{pg} &= \sum D^{nom}(K) \\ K &\in P_p \cap G_g \end{aligned} \tag{3}$$

where N_{Grid} is the total number of grids. K is a gate or wire which is located at the path P_p and also belongs to the grid G_g.

In Eq. (3), as shown in [18]–[24], $\boldsymbol{D}$ can be collected using the Automatic Test Equipment (ATE) or Built-In Self Testing (BIST). Since we focus on how to select effective paths for characterizing timing variations, the testing approaches to obtain $\boldsymbol{D}$ will not be extended. Please refer to literatures [18]–[24] to find detailed explanations. As the measurement accuracy has an impact on the fitting errors, we have considered the measurement errors in the experiments. The parameter $\boldsymbol{\Pi}$ can be obtained by Electric Design Automation (EDA) tools. And $\boldsymbol{V}$ is the unknown timing variation distribution to be calculated.

Please notice that, some single-sensitized paths can propagate both a rising and a falling transition. As rising delay and falling delay of the same gate usually are different, each path can provide two equations. Thus in the rest of the paper, one such path is counted as two paths.

As Eq. (3) is formulated based on the assumption that the timing variations of gates in the same grid are similar, a solution $\boldsymbol{V}$ that satisfies every equation will not exist. Therefore, we use the method of least squares to calculate $\boldsymbol{V}$:

$$\min \|\Pi \times V - D\|_2^2 \tag{4}$$

3. Data Sampling

The proposed key idea is shown in Fig. 3. We will solve the above mentioned least square problem to characterize the timing variation distribution. As the quality of timing characterization is determined by the sampled data, this section will analyze the sufficiency and uniformity of the sampled data. Then in the next section, we will elaborate how to select the paths to obtain the sampled data.

The fitting error metric can evaluate the accuracy of the fitted timing distribution. The relative fitting error of a grid g, represented by $E(g)$, is:

$$\overline{v}(g) = \frac{\sum v(k)}{N_{Gate}(g)} \quad k \in g \quad E(g) = \left| \frac{v_F(g) - \overline{v}(g)}{\overline{v}(g)} \right| \tag{5}$$

where $N_{Gate}(g)$ represents the number of gates or wires in the grid g, and k is a gate or wire in g, so $\overline{v}(g)$ represents the mean timing variations of the gates or wires in g, and $v_F(g)$ denotes the fitted timing variation of g.

As Eq. (4) contains N_{Grid} unknowns, so at least N_{Grid} paths should be selected for constructing N_{Grid} equations, and they are expected to have no linear correlation, i.e. without redundant equations. According to the sampling theory [25], the sampled data should be sufficient and uniform. Sampling sufficiency means at least how many gates or wires in grids should be sampled, while sampling uniformity means the sampled gates or wires should be evenly distributed in the grids.

Obviously, to accurately fit timing variations, every grid should be sampled. In general, higher fitting accuracy can be achieved with more samples. Meanwhile, the decrease of fitting error will dramatically slow down after the number of samples reaches a point. Assume the number of data is N, and the data is in the Gaussian distribution with a standard deviation σ. To achieve an absolute fitting error d with a confidence level u, at least N_S samples should be randomly selected [25]:

$$N_S = \frac{N \times u^2 \times \sigma^2}{N \times d^2 + u^2 \times \sigma^2} \tag{6}$$

For example, assuming a grid contains $N = 100$ gates, and the σ of their timing variations is 0.01, then to achieve the absolute fitting error $d = 0.01$ with the confidence level $u = 1.96$ (confidence coefficient 95%), at least $N_S = 4$ gates should be sampled. The theoretical analysis can help us to set the number of samples.

In regard to the sampling uniformity, to reflect the overall timing variation of the circuit, the sampled gates or wires are expected to be evenly distributed in the grids.

Fig. 3 Overview of the proposed idea.

In summary, the selected paths are expected to meet the following constraints.

***C*-1:** To meet the requirement of the Eq. (5), the selected paths must be single-sensitized paths;

***C*-2:** For the sampling sufficiency, it is expected that at least N_{Grid} paths should be selected to cover all the grids, and every grid is expected to be covered by different paths and through different gates or wires for reducing possibilities of constructing linearly correlated equations in the Eq. (4);

***C*-3:** For the sampling uniformity, the gates or wires covered by the selected paths are expected evenly distributed in each grid.

Among them, C-1 is the strictest constraint that must be met, while C-2 and C-3 are loose constraint that may not be met completely, but our proposed LAPS method will try to achieve them as much as possible.

4. Layout Aware Path Selection

The pseudo-code of the LAPS method is given in Algorithm 1. To cover a grid, the main-function *LAPS* selects a gate located in the grid, and then calls the sub-function *Select_One_PATH* to generate a complete path across the gate. We will use the example in Fig. 4 to introduce the algorithm.

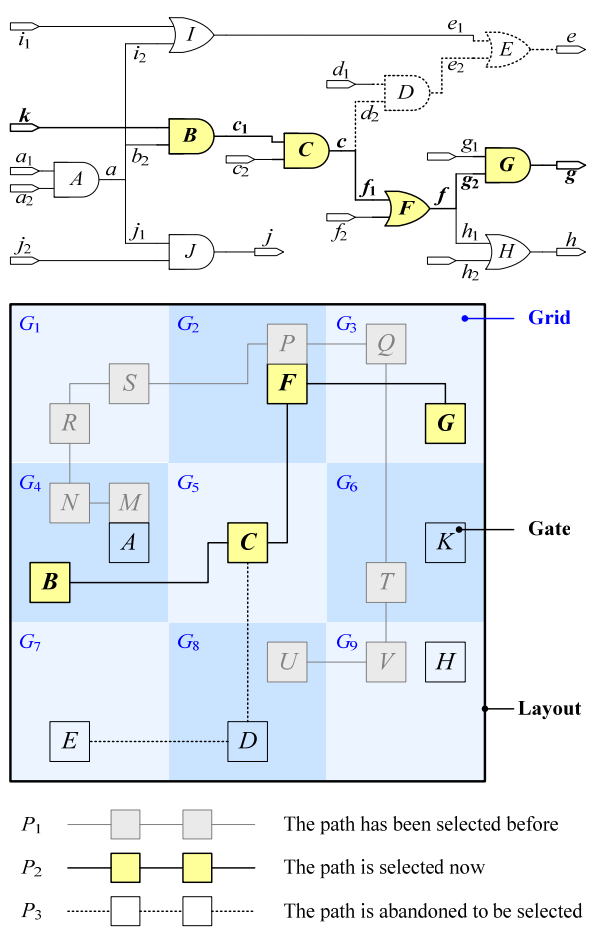


Fig. 4 An example of LAPS.

Algorithm 1 LAPS

```
Void LAPS( )
1   Define N_Grid;
2   Define N_SSP;
3   Set Num_Selected_Path=0;
4   While (Num_Selected_Path<N_SSP)
5   {
6   //Select one path
7   G_T=Select_A_Grid();
8   K_T=Select_A_Gate(G_T);
9   Select an input K_I that can arrive at K_T;
10  Set Success_Select=Select_One_Path(K_I, K_T);
11  If (Success_Select)
12  Num_Selected_Path++;
13  }

Bool Select_One_Path(K_I, K_T)
14  Set Success_Select=0;
15  Initial a stack K_Stack=ϕ;
16  Push K_I to K_Stack;
17  While (K_Stack≠ϕ)
18  {
19  Pop a gate K_s from K_Stack;
20  If (K_s is not an output)
21  Select and push K_s' succeeding gates to K_Stack;
22  Else
23  {
24  Store the path of the selected gates that
    go from K_I to K_T and finally to K_s;
25  Success_Select=1;
26  Break;
27  }
28  }
29  Return Success_Select;

Int Select_A_Grid( )
30  For (g=0;g<N_Grid;g++)
31  {C[g]=the number of selected paths that cover the grid g;}
32  Min_C=the minimum value of array C;
33  For (g=0;g<N_Grid;g++)
34  {If (C[g]==Min_C) Add g to G_MIN;}
35  g_r=randomly select a grid from G_MIN;
36  Return g_r;

Int Select_A_Gate(int G_T)
37  For (each gate g in G_T)
38  {C[g]=the number of selected paths that cover the gate g;}
39  Min_C=the minimum value of Array C;
40  For (each gate g in G_T)
41  {If (C[g]==Min_C) Add g to G_MIN;}
42  If (G_MIN contains only one gate g_r) Return g_r;
43  Else
44  {
45  For (each gate g in G_MIN) {
46  A[g]=the average layout coordinate of g and the selected gates in G_T;
47  D[g]=the distance between A(g) and the center of G_T;
48  }
49  Min_D=the minimum value of Array D;
50  For (each gate g in G_MIN)
51  {If (D[g]==Min_D) Add g to D_MIN;}
52  g_r=randomly select a grid from D_MIN;
53  Return g_r;
54  }
```

For the sake of simplicity, only the transistor layout is illustrated in the figure. The layout is divided into 9 grids, and there is a selected path P_1.

Line-1 and Line-2 of Alg. 1 present two user-defined parameters N_{Grid} and N_{SSP}. From Line-6 to Line-12, one single path selection is tried each time until the total number of selected paths reaches N_{SSP}. If no more single-sensitized paths can be found, the function *LAPS* ends. Based on the constraint C-2, all the grids are expected to be covered by the selected paths, and the grids are expected to be covered as many times as possible through different gates. Thus, in Line-7, the LAPS starts from selecting a grid G_T, which has been covered by the fewest number of selected paths, and then we will try to select a path across the G_T. If more than one grid satisfies this requirement, then one of them is randomly selected. The sub-function *Select_A_Grid* is shown in Line-30 to Line-36. Next in Line-8, a gate K_T is selected. It belongs to the G_T and has been covered by the fewest number of selected paths. If more than one candidate gates are selected, then the constraint C-3 which considers the uniform distribution, is used to guide the further selection of K_T from them. Based on C-3, if a gate can make the average layout coordinate of selected gates closest to the center of its grid, the gate is selected as K_T. The sub-function *Select_A_Gate* is shown in Line-37 to Line-54. As K_T may not be an input, in Line-9, an input K_I, which can arrive at K_T, is selected by the similar strategy based on C-2 and C-3. The major difference between selecting K_T and selecting K_I is that, instead of analyzing every gate of G_T in Line-37 and Line-40, every gate which can arrive K_T is analyzed for selecting K_I.

The sub-function *Select_One_Path* is called in Line-10 selects a path that goes from K_I and through K_T. At the beginning, the path only contains an input gate K_I. Then its succeeding gates are gradually selected and connected to the path one by one until arriving at an output gate. The detailed steps are as follows. Firstly, to go through K_T, the succeeding gates that are not in the logic cone of K_T are removed. Secondly, the succeeding gates that cannot satisfy the constraint C-1 are removed. Finally, the constraints C-2 and C-3 are used to determine the order of the remaining succeeding gates to be pushed into the stack *K_Stack*. The *K_Stack* is used for backtracing when the currently selected partial path cannot satisfy the C-1. If no more gates are left in the *K_Stack*, then no single-sensitized paths are available, and this path selection fails. The failed paths will be recorded to avoid unnecessary try in the future. Otherwise, anytime the K_s is an output gate, the selection is success.

For example, in Fig. 4, at the beginning, both the grids G_5 and G_7 have not been covered by the path P_1 yet, so we randomly select one grid from them: assuming G_5 is selected as the G_T. Then, in G_5, since the gate C has not been covered yet, it is selected as the K_T. Two input gates A and B can arrive at the C. Both of them are in the same grid. In G_4, two other gates M and N have already been covered. The average layout coordinate of B, M, and N is closer to the center of G_4 than that of A, M, and N, so B is selected as K_I. Now, K_T is C and K_I is B, so the currently selected partial path is k => B => C. Here C has two fanout gates D and F. According to C-1, the partial path k => B => C => D is a single-sensitized path when $b_2 = 1$, $c_2 = 1$, and $d_1 = 1$, while k => B => C => F is a single-sensitized path when $b_2 = 1$, $c_2 = 1$, and $f_2 = 0$. Although both D and F satisfy C-1, the average layout coordinate of D and U is closer to the center of G_8 than that of F and P to the

center of G_2. Hence D will be selected as the gate K_s in the next loop, and the currently selected partial path becomes k => B => C => D. The D has only one succeeding gate E. The partial path k => B => C => D => E is a single-sensitized path when $b_2 = 1$, $c_2 = 1$, $d_1 = 1$, and $e_1 = 0$. However, to make $e_1 = 0$, i_1 and i_2 must be 0 too, which conflicts $b_2 = 1$. Thus, E does not satisfy the C-1. In this case, another partial path k => B => C => F will be tried. Finally, the complete path k => B => C => F => G => g is successfully selected.

5. Experimental Results

5.1 Experiment Flow

The experiment flow is shown in Fig. 5. First of all, commercial EDA tools are used to generate layouts and extract normalized delays of gates and wires for these benchmark circuits. In our experiments, four ISCAS'89 benchmark circuits s13207, s15850, s35932, and s38417 and two largest ITC'99 benchmark circuits b18 and b19 are used. We use TSMC 65nm technology library to conduct synthesis, placement and routing. For each benchmark circuit, 10^6 paths are randomly selected. On average, the gates contribute more than 99% of the total delay of a path, so for the benchmark circuits, the wires have a much smaller effect on path delays. Thus in this experiment, only the timing variations of gates are considered. It should be understood that in some other circuits, the wires can contribute significant amount of path delays. As explained before, the proposed method can also characterize their timing variation distributions.

Secondly, the VARIUS model [16] is used to simulate the process variations of transistor threshold voltage and channel length. In [16], for the spatially correlated process variations, a multivariate normal distribution with a spherical spatial correlation structure is applied. In the VARIUS, the *phi*, correlation distance relative to the grid width, is set to 0.5. The spatially uncorrelated process variations are generated with Gaussian distribution. The size of grid of VARIUS is set to let every gate in the circuit has its own variation.

Next, the proposed LAPS method is used to select the paths for timing characterization. An open-source tool [26] is adopted to check whether a partial path or a complete path is a single-sensitized path or not.

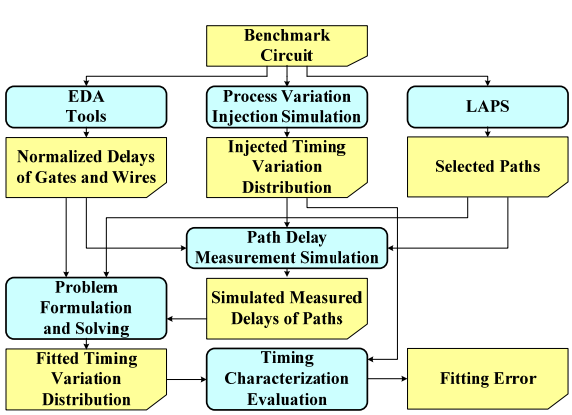


Fig. 5 Experiment flow.

With the injected timing variation distribution, the timing variations of every gate are obtained, so the measurement of path delays ($\boldsymbol{D}$ in Eq. (4)) can be simulated. Meanwhile, the matrix $\boldsymbol{\Pi}$ can be derived through the normalized delays of gates. With $\boldsymbol{D}$ and $\boldsymbol{\Pi}$, the least square problem of Eq. (4) is then solved by a commercial mathematical tool to fit the timing variation distribution $\boldsymbol{V}$. Finally, by comparing the injected timing variation distributions with the fitted distributions, the fitting error $E(g)$ is calculated for every grid g.

5.2 Selected Paths

The selected paths are expected to meet the three constraints C-1, C-2 and C-3, so N_{Grid}, N_{SSP}, and $R_{P/G} = N_{SSP}/N_{Grid}$ are set to different values according to the different scales of benchmark circuits. For ISCAS'89 benchmark circuits, all the grids can be covered. For ITC'99 benchmark circuits, when the number of grids is large, some grids are not covered, but the percentage of covered grids is still greater than 99%. The sampling sufficiency is shown in Table 1.

In Table 1, column D1 gives the average number of selected paths to cover a grid. For example, when $N_{Grid} = 100$ and $N_{SSP} = 100$, a grid is covered by 14.78 selected paths on average.

Column D2 gives the average number of selected paths to cover a gate. As explained before, if a single-sensitized path can propagate both a rising transition and a falling transition, this path is counted as two paths of N_{SSP}. When N_{Grid} and N_{SSP} are small, the selected paths indeed cover different gates in a grid. When N_{Grid} and N_{SSP} increase, as the total number of single-sensitized paths is limited by the circuit structure, the selected paths are more likely to cover the same gates. On average, when $N_{Grid} = 100$ and $N_{SSP} = 100$, a gate is only covered by 2.03 selected paths; when $N_{Grid} = 6400$ and $N_{SSP} = 19200$, a gate is covered by 4.68 paths.

The column D3 and D4 give the average number and the percentage of gates covered by the selected paths in a grid, respectively. For example, when $N_{Grid} = 100$ and $N_{SSP} = 100$, on average, 7.28 gates are covered in a grid. In general, Table 1 shows the selected paths achieve the expected sampling sufficiency required by C-2.

As for the third constraint C-3, the selected gates are expected uniformly distributed in each grid. Figure 6 illustrates the distributions of selected gates. For each circuit, every grid is divided into $5 \times 5 = 25$ sub-grids. The height of every column represents the accumulated number of selected gates in each sub-grid. We can see the sub-grids of a circuit contains a similar number of selected gates.

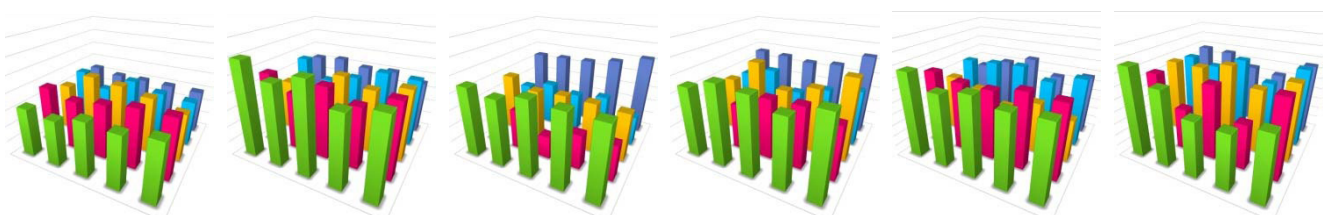
The runtime for selecting one path depends on the complexity of the circuit, so the runtime increases nearly linearly with the increasing of N_{SSP}. For the circuit s13207, it costs only less than 1 second to select a path, while for the circuit

Table 1 Sampling sufficiency of select paths

AN: Average number of gates in a grid; D1: Average number of selected paths to cover a grid; D2: Average number of selected paths to cover a gate; D3: Average number of gates covered by the selected paths in a grid; D4: Average percentage of gates covered by the selected paths in a grid.

	s13207 Number of Gates: 8176							s15850 Number of Gates: 9873							s35932 Number of Gates: 16385						
$R_{P/G}$	N_{Grid}	N_{SSP}	AN	D1	D2	D3	D4	N_{Grid}	N_{SSP}	AN	D1	D2	D3	D4	N_{Grid}	N_{SSP}	AN	D1	D2	D3	D4
1	10*10 =100	100	81.76	13.3	2.1	6.3	7.7%	10*10 =100	100	98.73	17.2	2.0	8.6	8.7%	10*10 =100	100	163.85	14.4	2.0	7.0	4.3%
2		200		29.4	2.5	12.1	14.8%		200		36.1	2.2	16.4	16.6%		200		24.5	2.1	12.0	7.3%
3		300		44.2	2.6	17.4	21.3%		300		53.3	2.4	22.5	22.8%		300		37.1	2.1	17.7	10.8%
1	20*20 =400	400	20.44	14.1	2.5	5.7	28.1%	20*20 =400	400	24.68	16.6	2.5	7.0	28.2%	20*20 =400	400	40.96	11.8	2.1	5.7	13.8%
2		800		27.6	3.0	9.3	45.7%		800		33.9	3.1	11.4	46.0%		800		23.0	2.3	10.2	24.9%
3		1200		40.9	3.5	12.0	58.8%		1200		51.7	3.7	14.2	57.6%		1200		32.7	2.5	13.3	32.4%
1	30*30 =900	900	9.08	14.9	3.3	4.7	52.0%	30*30 =900	900	10.97	17.1	3.3	5.5	49.7%	30*30 =900	900	18.21	12.7	2.4	5.5	30.2%
2		1800		29.7	4.7	6.6	72.2%		1800		34.9	4.6	7.7	70.4%		1800		23.5	2.9	8.4	46.0%
3		2700		44.6	6.1	7.4	81.8%		2700		53.4	6.4	8.5	77.4%		2700		33.2	3.4	10.0	54.7%

	s38417 Number of Gates: 22503							b18 Number of Gates: 62304							b19 Number of Gates: 124705						
$R_{P/G}$	N_{Grid}	N_{SSP}	AN	D1	D2	D3	D4	N_{Grid}	N_{SSP}	AN	D1	D2	D3	D4	N_{Grid}	N_{SSP}	AN	D1	D2	D3	D4
1	10*10 =100	100	225.03	14.2	2.0	7.2	3.2%	20*20 =400	400	155.76	13.9	2.18	7.94	5.1%	30*30 =900	900	138.56	13.8	2.16	5.68	4.1%
2		200		30.4	2.1	14.6	6.5%		800		29.2	2.37	13.08	8.4%		1800		28.7	2.31	9.70	7.0%
3		300		46.9	2.2	21.6	9.6%		1200		39.8	2.52	18.54	11.9%		2700		40.9	2.44	14.27	10.3%
1	20*20 =400	400	56.26	15.5	2.2	7.3	12.9%	40*40 =1600	1600	38.94	13.4	2.54	6.19	15.9%	55*55 =3025	3025	41.22	13.4	2.48	6.10	14.8%
2		800		31.5	2.5	12.9	22.9%		3200		29.8	2.88	9.35	24.0%		6050		30.2	2.79	9.15	22.2%
3		1200		47.1	2.8	17.4	31.0%		4800		42.8	3.35	12.85	33.0%		9075		44.0	3.16	12.53	30.4%
1	30*30 =900	900	25.00	14.9	2.4	6.3	25.1%	60*60 =3600	3600	17.31	12.9	3.31	4.69	27.1%	80*80 =6400	6400	19.49	13.1	3.09	4.64	23.8%
2		1800		29.8	2.9	10.4	41.4%		7200		30.8	4.09	7.37	42.6%		12800		30.5	3.69	7.85	40.3%
3		2700		44.6	3.4	13.4	53.4%		10800		42.5	5.40	9.85	56.9%		19200		44.2	4.68	10.58	54.3%



s13207(N_{Grid}=100) s15850(N_{Grid}=100) s35932(N_{Grid}=100) s38417(N_{Grid}=100) b18(N_{Grid}=400) b19(N_{Grid}=900)

Fig. 6 Sampling uniformity of selected paths ($R_{P/G}$ = 3).

b19, it costs about 15 seconds to select one path.

5.3 Fitting Errors

For simplicity, the following characters are used in the legends of subsequent figures:

$$\Delta : \frac{\textit{Delay under Process Variation} - \textit{Normalized Delay}}{\textit{Normalized Delay}} \tag{7}$$

MAX: Maximum MIN: Minimum
AVG: Average SD: Standard Deviation

First of all, the fitting errors are evaluated when only spatial correlated process variations are injected. The experimental results are shown in Fig. 7. Generally speaking, the fitting errors range from 1.72% to 4.36%. Under the same number of grids, with more selected paths, more gates in each grid are covered, so lower fitting errors are achieved. When N_{Grid} = 100 and N_{SSP} = 100, the average fitting error is 3.18%; with N_{SSP} = 300, the average fitting error is 2.42%.

This data shows the proposed method can achieve high accuracy in confronting with spatial correlated process variations. In reality, spatial uncorrelated process variations also exist. Figure 8 illustrates the fitting errors, where the R_{UNC} is used to represent the percentage of contributions of spatial uncorrelated process variations to the timing variations of gates. Generally speaking, when R_{UNC} = 30%, the fitting errors range from 2.82% to 5.29%. Similarly, with more selected paths, lower fitting errors are achieved. As more spatial uncorrelated process variations may result in larger standard deviation of gate timing variations in each grid, same number of covered gates in a grid would result in a higher fitting error.

Besides of spatial uncorrelated process variations, in reality, measurement errors also exist. Though repeating measurement can reduce these errors [27], the delays of selected paths can still not be obtained accurately [16], [18], [27]. Hence, the fitting errors are further evaluated when measurement errors with Gaussian distribution (SD ≈ 1.6%) exist. Figure 9 illustrates the fitting errors when R_{UNC} = 30% of Fig. 8. With the increasing of the selected paths, the fitting errors decrease too. When $R_{P/G}$ = 3.0, the average fitting error reduces to 5.47%. Roughly comparing with the previous work [18] whose average fitting error is 8.35%, the proposed method effectively improves the accuracy.

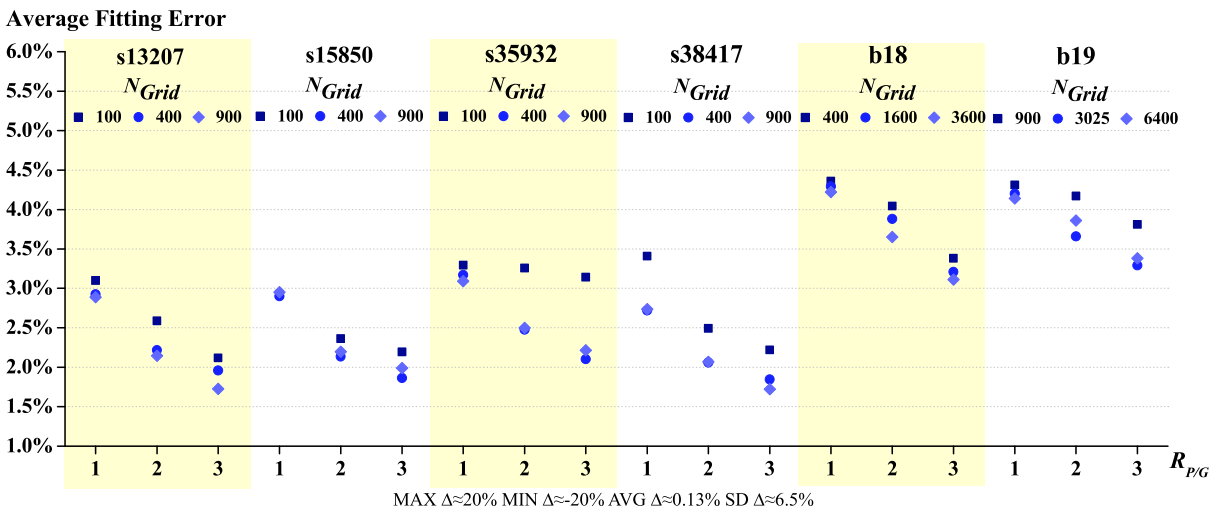


Fig. 7 Fitting errors for spatial correlated process variations.

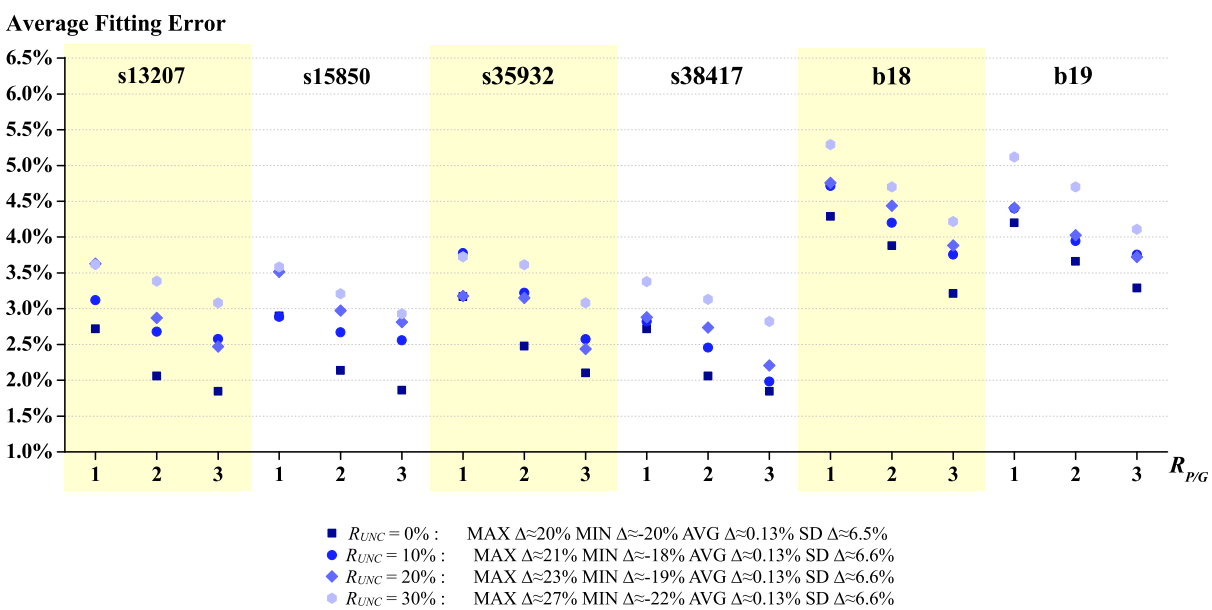


Fig. 8 Fitting errors for spatial uncorrelated process variations (N_{Grid} is 400 for s13207, s15850, s35932, and s38417; N_{Grid} is 1600 for b18; N_{Grid} is 3025 for b19)

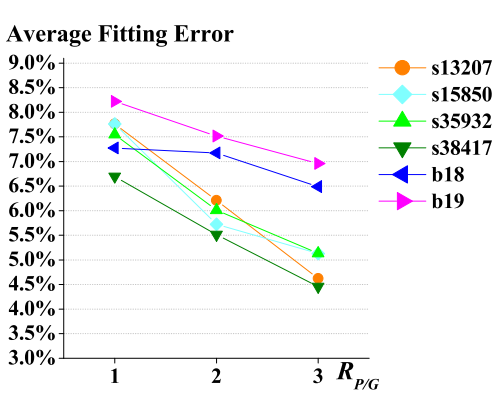


Fig. 9 Fitting errors for measurement errors

6. Conclusion

This paper proposes a LAPS method for post-silicon timing characterization. We analyze the sufficiency and the uniformity of sampled data for effectively fitting timing variations. Then we select paths with consideration of single-sensitize, sampling sufficiency and sampling uniformity. Experiments on benchmark circuits show that, by selecting only hundreds of paths, we can keep the fitting errors of timing distribution below 4.4% when only spatial correlated process variations exist, below 5.3% when spatial uncorrelated process variations also exist, and below 8.2% when measurement errors are involved.

Acknowledgments

This work is supported in part by National Natural Science Foundation of China (NSFC) under grant No. 61532017, 61376043, and 61274030.

References

[1] C. Zhuo, K. Agarwal, D. Blaauw, and D. Sylvester, "Active Learning Framework for Post-Silicon Variation Extraction and Test Cost Reduction," ICCAD, pp.508–515, 2010.
[2] Q. Liu and S.S. Sapatnekar, "Synthesizing a Representative Critical Path for Post-Silicon Delay Prediction," ISPD, pp.183–190, 2009.
[3] C. Zhuo, D. Blaauw, and D. Sylvester, "Post-Fabrication Measurement-Driven Oxide Breakdown Prediction and Management," ICCAD, pp.441–448, 2009.
[4] F. Koushanfar and M. Potkonjak, "CAD-Based Security, Cryptography, and Digital Rights Management," DAC, pp.268–269, 2007.
[5] J. Ye, Y. Hu, and X. Li, "OPUF: Obfuscation Logic Based Physical Unclonable Function," IOLTS, pp.156–161, 2015.
[6] J. Ye, Y. Hu, and X. Li, "DCPUF: Placement and Routing Constraint based Dynamically Configured Physical Unclonable Function on FPGA," FPGA, poster 17, pp.279–279, 2016.
[7] D. Blaauw, K. Chopra, A. Srivastava, and L. Scheffer, "Statistical Timing Analysis: From Basic Principles to State of the Art," IEEE TCAD, vol.27, no.4, pp.589–607, 2008.
[8] J. Deng, A. Rahman, R. Thoma, P.W. Schneider, J. Johnson, H. Trombley, N. Lu, R.Q. Williams, H.M. Nayfeh, K. Zhao, R. Robison, X. Guan, N. Zamdmer, S. Shuma, B. Worth, J.E. Sundquist, E.A. Foreman, S.K. Springer, and R. Wachnik, "SOI FinFET nFET-to-pFET Tracking Variability Compact Modeling and Impact on Latch Timing," IEEE TED, vol.62, no.6, pp.1760–1768, 2015.
[9] S.R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas, "VARIUS: A Model of Parameter Variation and Resulting Timing Errors for Microarchitects," IEEE TSM, vol.21, no.1, pp.3–13, 2008.
[10] J. Fan, N. Mi, S.X..-D. Tan, Y. Cai, and X. Hong, "Statistical Model Order Reduction for Interconnect Circuits Considering Spatial Correlations," DATE, pp.1–6, 2007.
[11] R. Zheng, L. Xi, H. Shaojian, C. Shoumian, Z. Yuhang, and S. Yanling, "A Correlation-Considered Variation-Aware Interconnect Parasitic Profile Extraction Method," IEEE TED, vol.59, no.9, pp.2321–2326, 2012.
[12] L. Milor, L. Yu, and B. Liu, "Logic Product Speed Evaluation and Forecasting During the Early Phases of Process Technology Development using Ring Oscillator Data," IWSM, pp.20–23, 1997.
[13] H.K. Alidash, A. Calimera, A. Macii, E. Macii, and M. Poncino, "On-Chip Process Variation-Tracking Through an All-Digital Monitoring Architecture," IET CDS, vol.6, no.5, pp.366–373, 2012.
[14] A. Ghosh, R.M. Rao, J.-J. Kim, C.-T. Chuang, and R.B. Brown, "Slew-Rate Monitoring Circuit for On-Chip Process Variation Detection," IEEE TVLSI, vol.21, no.9, pp.1683–1692, 2013.
[15] X. Li, R.R. Rutenbar, and R.D. Blanton, "Virtual Probe: A Statistically Optimal Framework for Minimum-Cost Silicon Characterization of Nanoscale Integrated Circuits," ICCAD, pp.433–440, 2009.
[16] W. Zhang, X. Li, and R.A. Rutenbar, "Bayesian Virtual Probe: Minimizing Variation Characterization Cost for Nanoscale IC Technologies via Bayesian Inference," DAC, pp.262–267, 2010.
[17] M. Li, A. Davoodi, and L. Xie, "Custom On-Chip Sensors for Post-Silicon Failing Path Isolation in the Presence of Process Variations," DATE, pp.1591–1596, 2012.
[18] F. Koushanfar, P. Boufounos, and D. Shamsi, "Post-Silicon Timing Characterization by Compressed Sensing," ICCAD, pp.185–189, 2008.
[19] M. Potkonjak, "Non-Invasive Timing Characterization Integrated Circuits Using Sensitizable Signal Paths Sparse Equations," US Patent US008176454B2, 2012.
[20] E.J. Jang, J. Chung, A. Gattiker, S. Nassif, and J.A. Abraham, "Post-Silicon Timing Validation Method Using Path Delay Measurements," ATS, pp.232–237, 2011.
[21] X. Zhang, J. Ye, Y. Hu, and X. Li, "Capturing Post-Silicon Variation by Layout-Aware Path-Delay Testing," DATE, pp.288–291, 2013.
[22] E.J. Jang, A. Gattiker, S. Nassif, and J.A. Abraham, "Efficient and Product-Representative Timing Model Validation," VTS, pp.90–95, 2011.
[23] X. Wang, M. Tehranipoor, S. George, D. Tran, and L. Winemberg, "Design and Analysis of a Delay Sensor Applicable to Process/Environmental Variations and Aging Measurements," IEEE TVLSI, vol.20, no.8, pp.1405–1418, 2012.
[24] A. Drake, R. Senger, H. Deogun, G. Carpenter, S. Ghiasi, T. Nguyen, N. James, M. Floyd, and V. Pokala, "A Distributed Critical-Path Timing Monitor for a 65 nm High-Performance Microprocessor," ISSCC, pp.398–399, 2007.
[25] S. Feng, et. al., SampleSurvey Theory and Methods, China Statistics Press, 2nd ed., 2012. http://www.biblio.com/book/sample-survey-theory-methods-2nd-edition/d/657140208
[26] http://www.princeton.edu/~chaff/zchaff.html
[27] J. Aarestad, C. Lamech, J. Plusquellic, D. Acharyya, and K. Agarwal, "Characterizing Within-Die and Die-to-Die Delay Variations Introduced by Process Variations and SOI History Effect," DAC, pp.534–539, 2011.

Yu Hu received the B.S., M.S., and Ph.D. degrees all in electrical engineering from University of Electronic Science and Technology of China (UESTC), in 1997, 1999, and 2003, respectively. She is currently a professor at the Institute of Computing Technology, Chinese Academy of Sciences. Her research interests generally include architectural-level and circuit-level reliable design and fault diagnosis techniques. She is a member of ACM, IEEE, IEICE, and CCF.

Jing Ye received the Ph.D degree in the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS) from University of Chinese Academy of Sciences (UCAS) in 2014. He received the B.S. degree in School of Electronics Engineering and Computer Science from Peking University in 2008. He is currently an assistant professor at the State Key Laboratory of Computer Architecture, ICT, CAS. His research interests generally include VLSI test, diagnosis, and hardware security.

Zhiping Shi received the Ph.D. degree in computer software and theory from Institute of Computing Technology, Chinese Academy of Sciences, China, in 2005. Currently he is an associate professor at the Capital Normal University, Beijing, China. His research interests include formal verification and visual information analysis. He has been a research staff at Institute of Computing Technology, Chinese Academy of Sciences from 2005 to 2010.

Xiaowei Li received the B.Eng. and M.Eng. degrees in computer science from Hefei University of Technology, China, in 1985 and 1988, respectively, and his Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), in 1991. From 1991 to 2000, he was an assistant professor and an associate professor (since 1993) in the Department of Computer Science and Technology, Peking University, China. He joined the ICT, CAS as a professor in 2000. He is now the deputy director of the State Key Lab. of Computer Architecture, ICT, CAS. Dr. Li's research interests include VLSI testing, design for testability, design verification, dependable computing, and wireless sensor networks. Dr. Li serves as chair of Technical Committee on Fault Tolerant Computing, CCF (China Computer Federation) since 2008. He serves as vice chair of IEEE Asian Pacific Regional TTTC (Test Technology Technical Council) since 2004. He serves as the steering committee Chair of IEEE Asian Test Symposium since 2011. In addition, he serves on the Technical Program Committee of several IEEE and ACM conferences, including VTS, DATE, ASP-DAC, PRDC, etc. He also serves as member of editorial board of JCST, JOLPE, JETTA, etc.