Layout-Aware Critical Path Delay Test Under Maximum Power Supply Noise Effects

Junxia Ma, *Member, IEEE,* and Mohammad Tehranipoor, *Senior Member, IEEE*

***Abstract*—As technology shrinks, gate sensitivity to noise increases due to supply voltage scaling and limited scaling of the voltage threshold. As a result, power supply noise (PSN) plays a greater role in sub-100 nm technologies and creates signal integrity issues. It is vital to consider supply voltage noise effects: 1) during design validation to apply sufficient guardbands to critical paths, and 2) during path delay test to ensure the performance and reliability of the chip. In this paper, a novel layout-aware pattern generation procedure is proposed to maximize PSN effects on critical paths considering the impact of local voltage drop. The proposed pattern generation and validation flow is implemented on the ITC'99 b19 benchmark. Experimental results for both wire-bond and flip-chip packaging styles are presented. Results demonstrate that our proposed method is fast, significantly increases switching around the functionally testable critical paths, and induces large voltage drop on cells placed on the critical paths which results in increased path delay. The proposed method eliminates the very time consuming pattern validation phase that is practised in industry.**

***Index Terms*—Path delay test, pattern generation, power supply noise, signal integrity.**

I. Introduction

SCALING technology has been continuing to push for more complex designs. The power supply voltage has also been scaling to reduce power consumption when moving to lower technology nodes. However, the reduced supply voltage also reduces the noise immunity which in turn reduces signal integrity and negatively impacts performance and reliability.

In general, power supply noise (PSN) refers to voltage spikes or droops on the power and ground distribution network. This can be induced by large currents drawn through the resistive power distribution network (PDN), creating IR-drop, or sudden changes in current, causing $L\frac{dI}{dt}$ effects. Supply voltage noise compromises the cell driving capability and thus slows down the cell transition, potentially violating setup and hold times and creating clock skew. All of these effects would degrade the circuit performance or potentially result in functional failure.

Manuscript received April 16, 2011; revised May 12, 2011; accepted June 24, 2011. Date of current version November 18, 2011. This work was supported in part by Semiconductor Research Corporation, under Contract 2053, and by NSF, under Grants CCF-0811632 and ECCS-0823992. This paper was recommended by Associate Editor S. Mitra.

J. Ma is with LSI Corporation, Milpitas, CA 95035 USA (e-mail: junxia@engr.uconn.edu).

M. Tehranipoor is with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269 USA (e-mail: tehrani@engr.uconn.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCAD.2011.2163159

The sensitivity of the cell delay to the PSN increases as supply voltage scales while there is little or no threshold voltage scaling [1]. It has been shown that a 10% voltage drop increases the gate propagation delay by 8% in a 180 nm design [2] and 30% in 130 nm technology [3]. In 90 nm technology, 1% voltage variation causes approximately 4% change in gate delay [4]. These examples highlight the increasing impact of PSN on cell delays as technology scales.

Path delay test is commonly used to test the accumulated delay effects resulting from physical defects and environmental noises. It is useful for characterizing the timing of critical paths. For the path under test, various filling methods cause different PSN that indicates that PSN is pattern dependent. The 0/1/random/adjacent-fill methods as well as functional patterns usually used in practice may not necessarily cause sufficient PSN on paths under test, which would allow the chip to pass manufacturing test but fail in the field (known as escape).

Power supply noise also depends on the layout. Both the location of the switching gates and the design of power distribution network can impact the PSN. For the same pattern generated using the same netlist, when applied to different layout designs, the distribution of the switching gates can be different. This may result in entirely different hot-spots on the chip and supply noise levels around the critical path. The structure of the power distribution network and the location of power pads/power bumps would also make a big difference to the on-chip PSN. Therefore, it would be difficult to determine the worst-case path delay induced by PSN using patterns generated by conventional automatic test pattern generation (ATPG) tools because they are layout-unaware.

To address this issue, a pattern generation method is proposed in this paper to maximize the PSN on the critical paths by considering actual physical layout information. With the proposed method, a worst-case test pattern is generated for each path under test while the total test power can be maintained at a specific level (e.g., functional power level).

A. Related Prior Work

Several approaches have been proposed for PSNmodeling, analysis, and estimation in recent years in [5]–[13].

Closed form equations for calculating simultaneous switching noise were derived in [5]. The authors in [6] discussed the challenges in power and ground integrity. Modeling of PSN on distributed on-chip networks was described in [7]. Mitra *et al.* [8] defined power supply droop fault model and

generated stuck-at patterns for it. Two metrics that quantify the PSN effects on gate delay were described and validated in [9]. The DVD_{avg} and DVD_{max} are the average and peak values of the dynamic voltage drop (DVD) profile in the timing cycle, respectively. The authors proved that the effect of the voltage drop profile on a digital path timing performance is equivalent to applying a fixed supply voltage of $VDD - DVD_{\text{avg}}$. A similar simulation method is used in this paper to simulate the voltage drop impact on path delay. Wang *et al.* [10] developed power noise models for array-bond and wire-bond chips during delay testing. The models were used to compact test vectors to meet noise or delay constraint. In [11] and [12], the impact of PSN on circuit performance was studied. IR-drop related failures due to highly localized peak test power have been observed in [13].

Techniques to minimize the test power in test capture mode were presented in [14]–[18]. Their goal is to reduce yield loss incurred by excessive PSN, which is an important issue during production test.

A vector-less analysis approach to compute the maximum path delay under power supply fluctuations was proposed in [3]. The path delay maximization problem was formulated as a constrained linear optimization problem considering the PSN effect. Nourani *et al.* [19] proposed a pattern generation method to maximize the PSN by maximizing the switching activity in the first few levels of logic with lower fan-outs. It uses ATPG and power simulators to evaluate the gate-level netlist to find patterns that cause maximal switching activity.

Several genetic-algorithm-based methods were proposed to find patterns that maximize supply noise [20]–[23]. In [20] and [21], randomly-filled patterns are generated first, then waveform simulations are performed to find the pattern that yields the largest supply noise. A combination of Monte Carlo and genetic algorithm was implemented in [23] to search for the worst-case input vector pairs that induce maximal switching noise.

B. Motivations, Contribution, and Paper Organization

Most of the test pattern generation methods previously proposed only use gate-level information and have no knowledge about the physical location of the critical paths or any other gates. They may have addressed the noise maximization problem, but the patterns generated by these methods do not necessarily result in the worst-case critical path delays nor chip performance because the layout information such as cell placement and on-chip power distribution network is not considered.

In this paper, we propose a layout-aware pattern generation method to increase the switching activity around the critical paths. By increasing the switching activity of the neighboring cells around the critical paths, we can generate a pattern that maximizes supply noise effects on cells in the path under test. The increase in switching occurs only around the critical paths while the total switching activity in the circuit can be kept below a predefined functional threshold by taking advantage of its don't-care bits. These patterns can be used to verify design margins during validation or to determine the chip performance and reliability during production test. It enables the test engineers to apply the most effective patterns during design validation/first silicon validation and later deploy them in production test as effective screens. This procedure also helps the following.

1) Address the issue of mis-binning. Traditionally, speed binning was done only for microprocessors. Over the past few years, conducting speed binning for application-specific integrated circuit designs has gained significant attention due to excessive variations in final products' frequency. Since patterns selected by this procedure can generate the worst-case supply noise on target paths, they can be used to characterize the speed path and help address the mis-binning problem.
2) Minimize test escapes of timing marginality due to PSN conditions. Test patterns that do not have the worst-case supply noise level would cause the chip pass the test during production test. However, there may be a pattern in the field that cause larger noise and fail the chip. In this case, the chip would be considered under-tested (tested optimistically). The patterns generated by this procedure can be used to avoid such escapes for the production test.

The proposed pattern generation procedure is applicable to both launch-off-shift (LOS) and launch-off-capture (LOC) schemes. In this paper, we use LOC method to generate path delay fault (PDF) patterns. Since the second vector in a LOC vector-pair is a functional response of the first vector, with a good chance it would be functionally valid or possess a near functional property [24], [25]. Since process variations and environmental noises can impact the path delay significantly, to obtain the actual critical paths in the silicon, we suggest selecting a large number of long paths and running the proposed flow first. Then the patterns are applied to silicon on a number of parts with the maximized PSN generated using our flow. The longest paths among these parts will be selected as the most critical paths to be targeted. Note that, in this paper, we do not address the issue of critical path selection, instead we assume that critical paths have already been selected using static or statistical static timing analysis tools [26], [31].

In this paper, we use a fast and accurate method to validate the patterns generated. Based on rail analysis run with a commercial tool, we perform supply noise aware timing analysis for the target path to predict its path delay and compare it with the delays caused by a large number of randomly-filled PDF patterns. Simulation results show that the pattern generated by the proposed method, compared with randomly-filled PDF patterns, can cause a much larger voltage drop on critical path and hence path delay.

Note that a vector pair is needed to analyze PSN in integrated circuits. For the sake of simplicity, in this paper, we use term "pattern" intermittently for vector pair and PDF pattern.

The remainder of this paper is organized as follows. Section II discusses the localized IR-drop effects and the path delay caused by these effects. Section III presents the pattern generation procedure. Experimental results are shown in Section IV. The concluding remarks are given in Section V.

II. Supply Voltage Noise Induced Delay Analysis

PSN includes two major components: inductive and resistive power/ground voltage noise. The inductive noise ($L\frac{dI}{dt}$) depends on the rate of change of the instantaneous current flowing through the power/ground distribution network, where the inductance L is mainly introduced by package lead and wire/substrate parasitics. The resistive noise (IR) is contributed by the current flow and the resistance of the power/ground network. Generally, the resistive voltage drop occurring on the power network is called IR-drop, while resistive or inductive voltage increase on ground network is called ground-bounce. Both the IR-drop and ground bounce will decrease the operating voltage range of the chip and may lead to timing problems and functional failures.

Since the inductance L and resistance R of the power/ground distribution network can be considered as fixed when the final design is given, the larger the change in current and instantaneous current is, the higher would the PSN be. In typical CMOS integrated circuits, instantaneous current is mostly caused by gate switching. By increasing switching activity in a circuit, the current it draws from the PDN will also increase which in turn increases voltage drop. Although our method can potentially increase the inductive noise as well, in this paper we only focus on the resistive noise increase, i.e., total IR-drop on both power and ground networks.

A. Voltage Drop Effects on Path Delay

Path delay consists of gate delay and interconnect delay. The increase in gate delay has direct impact on the path delay. The IR-drop decreases the effective supply voltage of the gate under consideration, reduces its driving strength and thus increases the gate delay.

To illustrate the relation between path delay and its associated voltage drop, we target a functionally testable critical path in b19 benchmark. To create a pattern set with patterns having different PSN levels, a PDF pattern is generated for the targeted path with don't-care bits unfilled, then the don't-care bits in the pattern are filled randomly with bit "0" probability increasing from 0% to 100% 1000 times to generate 1000 PDF patterns. For example, pattern 1 is all 1-fill pattern, pattern 1000 is all 0-fill pattern, and pattern 500 is filled with 50% bit 0/1.

Fig. 1 shows the switching activities for the 1000 randomly-filled PDF patterns. The x-axis is the pattern index and the y-axis represents the number of total switching nets in the circuit during launch-to-capture cycle. It can be observed that the switching activities are different for PDF patterns filled with different 0/1 probabilities. The patterns with very high probability of bit "0" show lower switching activities. Path delay for the same path under the 1000 randomly-filled PDF patterns is presented in Fig. 2. Details about the simulation method for path delay with PSN are discussed in Section IV. From these results we make the following observations.

1) PDF patterns filled with different 0/1 probabilities cause different switching activities and path delay on critical path.

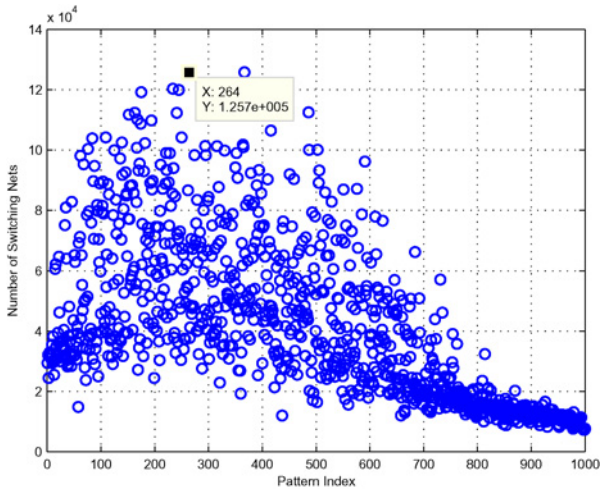

Fig. 1. Switching activities for 1000 randomly-filled PDF patterns.

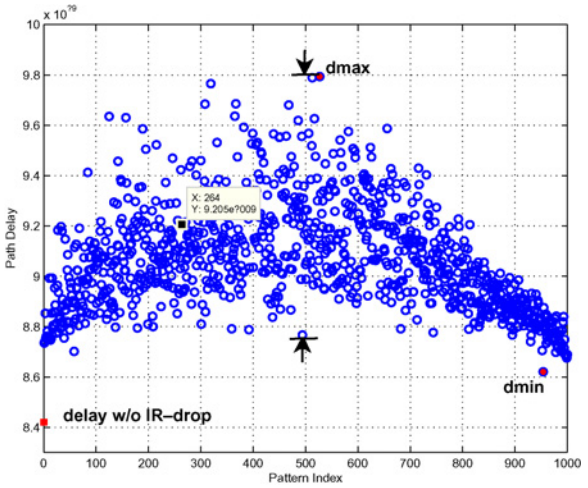

Fig. 2. Path delay of one path in *b19* benchmark for 1000 randomly-filled PDF patterns, voltage drops are different for these patterns.

2) High switching activity on the whole design does not necessarily cause large path delay for the path under test. For example, pattern 264 has the highest switching activity (see Fig. 1) but not the longest path delay (see Fig. 2).
3) The middle patterns present longer path delays compared with 0-fill or 1-fill at both ends. Even for patterns filled with around 50% bit 0/1 probabilities, which is usually the case using ATPG in commercial tool, the path delays still vary in a wide range as indicated by the two arrows in Fig. 2.

To further investigate the relationships between net switching activity, critical path IR-drop and path delay, correlation coefficients are calculated. The correlation between net switching activity and path delay is 0.54. In contrast, the average critical path IR-drop and path delay shows a much stronger correlation as 0.99. This implies that increasing the IR-drop on critical path is more efficient in increasing the path delay compared with blindly increasing the total switching activities on the whole layout design or randomly filling don't-care bits.

Clearly, pattern with *dmax* path delay (as shown in Fig. 2) is preferred over the other patterns in terms of checking the timing margin during design validation. As discussed earlier, the layout-unaware randomly filling method used by today's commercial tools cannot guarantee that the resulting pattern has *dmax* path delay. To address this issue, a pattern generation flow is proposed in this paper that generates patterns with maximal path delay. Since path delay has a strong dependance on average IR-drop of the critical path, this property will be exploited to generate patterns that maximize the IR-drop which in turn induces maximal path delay.

B. Localized IR-Drop Analysis

IR-drop in integrated circuits is a pattern and physical layout dependent effect. The on-chip power distribution network design and the switching activity distribution introduced by a pattern have direct impact on cells' IR-drop. Since in a flip-chip package, very dense power/ground bumps and wide power stripes are used in modern designs, the IR-drop resulting from package and upper level metals becomes very small; instead the local part of the power grid dominates the total IR-drop. To maximize the IR-drop effect on cells in critical path, the IR-drop effect caused by localized switching activity has to be understood.

In standard-cell based designs, cells are typically placed in rows and power is distributed among cells by power and ground rails on the top and bottom of each cell. As the cells are placed side-by-side, this forms a single rail across the row. Supply rails are usually routed in lower level metal layers (Metal 1 or Metal 2, depending on the supply design strategy used in the standard cell library) and are connected to the global power distribution network in the upper layers by power vias. The global PDN in upper level metal layers may have different structures and resistive properties for different designs. However, their resistive properties in local PDN (lower-level metal layers used in local PDN and associate vias) are similar. Fig. 3 illustrates cells and power/ground rails in local PDN, which helps analyze the IR-drop effect in localized area. In this figure, rows are placed back-to-back and share a common power rail. Power rails and ground rails are routed in Metal 1 and connected to Metal 6 through stacked vias (e.g., *ViaA* and *ViaB*). Fig. 3 also shows part of a critical path going through this region. Cell *G3* of this critical path is placed in between the power and upper ground rails as shown in the figure.

Assuming cell *G4* is the only switching cell, consuming current *I4*, and no other gates are drawing current from the supply rails in this region, *ViaA* and *ViaB* will supply most, if not all, of the current required for cell *G4*. Current *I4* can be broken down into two basic current components, current drawn from the left of *G4*, I_{4L}, and current drawn from the right of *G4*, I_{4R}. The amount of current drawn through each via depends on the distance (i.e., resistance) between the switching cell and the via. As the current drawn by cell G4 flowing through the power rail, the neighboring cells will also experience voltage drop. For example, cell *G3* will see voltage drop of $I_{4R}\,(R_{\text{global}}+R_{\text{viaA}}+R_1+R_2+R_3)$ because of the current drawn by cell G4 through *ViaA* and other neighboring vias

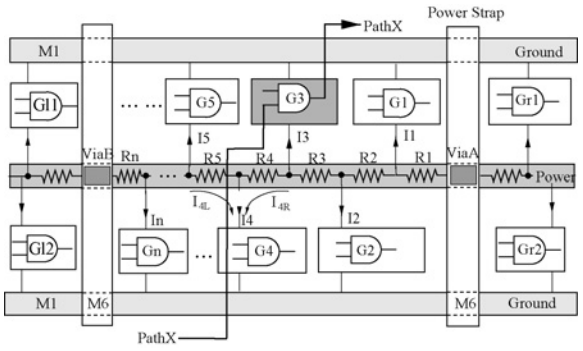

Fig. 3. Local resistive power distribution network in a standard cell design. Cell *G3* is part of the critical path *PathX*.

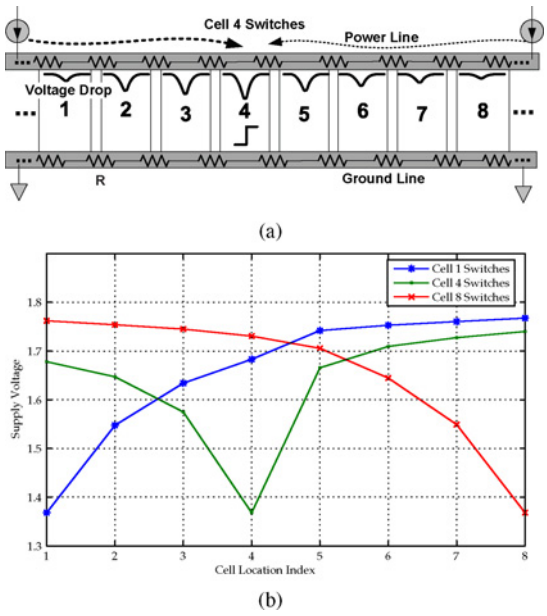

Fig. 4. Local IR-drop effect caused by single cell switching. (a) Locations of eight cells. (b) Voltage drop on each cell when cell 1, 4, or 8 switches.

from the right side of *ViaA*. Here, R_{global} is the resistance of the global PDN in top layer metals between power pads/C4 bumps to *ViaA*. The same analysis would apply for voltage drop on cells located to the left of cell *G4*.

Simulations have been performed to analyze localized IR-drop effects caused by cell switchings in close proximity. Fig. 4(a) shows the relative location of eight adjacent cells in one row that are connected to the same power rail (e.g., Metal 1). The two current sources in this figure model the location of power vias that connects to global PDNs in higher metal layers (e.g., Metal 5 and Metal 6). To investigate the impact of cell location, we use the same type of cells in this model to remove the impact of gate size. Fig. 4(b) shows the supply voltage variation for these eight cells with a single cell switching each time, which are cell 1 (closest to left power via), cell 4 (in the middle of the row), and cell 8 (closest to right power via), respectively. The simulation results show the switching cell experiences the largest voltage drop, and the neighboring cells on both sides experience smaller voltage drop. The closer the neighboring cell to the switching cell,

the larger the IR-drop. These conclusions also hold for case with different gate sizes. A larger gate needs more switching current thus causes higher voltage drop on neighboring cells when switching.

This analysis clearly demonstrates that the current induced by the switching of one cell has impact on the voltage drop of other cells connected to the same power rail. Thus, to maximize the voltage drop on one particular cell (e.g., a cell on the target critical path), we can maximize the switching activity of the neighboring cells that share the same power rail.

As shown in Fig. 3, cell *G3* is on the critical path *PathX*. To maximize the effect of PSN on *G3*, transitions can be generated on the nearby cells whose current contributes to the voltage drop on cell *G3*. The range of the neighboring cells depends on how much current cell *G3* would draw from its nearest via *ViaA* and its neighboring vias (vias from both sides of *ViaA*). In this figure, cells *G1* to *Gn* between the two power straps, some cells from the right side of *ViaA* and from the left side of *ViaB* can be considered as neighboring cells. We will discuss the range of the neighboring cells used in this paper in Section III.

Decoupling capacitance (decap) cells are usually inserted in the layout as the most common technique used to reduce IR-drop and Ldi/dt effects. The decap cells as well as the non-switching device capacitances serve as additional current sources for nearby cells. Although in Fig. 4(a) only non-switching device capacitances are considered, the above localized IR-drop analysis conclusion still holds in presence of the decaps. After the chip tape-out, the number of decaps are fixed; thus increasing switch activities around the critical path can worsen the voltage drop effect as well.

In this paper, we utilize the localized property of the voltage drop effect to introduce a large voltage drop on cells along the critical paths. The patterns generated by this method represent worst-case voltage drop for cells on critical paths and produce more accurate prediction for the worse-case timing performance. The localized property also shows that increasing switching in the entire circuit do not necessarily increase IR-drop on the target path; the switchings in those cells near critical path are more important and contribute more directly to the critical path delay.

III. Pattern Generation Procedure

We identify critical paths using a commercial timing analysis tool after the physical design and parasitic information has been extracted. For each target path, one pattern is generated using our procedure to maximize the PSN. As shown in Fig. 5, the proposed pattern generation procedure consists of three major steps: 1) cells identification; 2) virtual test points insertion (TPI); and 3) layout-aware transition delay fault (TDF) ATPG. Each of these steps is briefly described in the following.

A. Cells Identification

A design exchange format (DEF) file is used to identify the cells of the critical path on the layout. The DEF file is

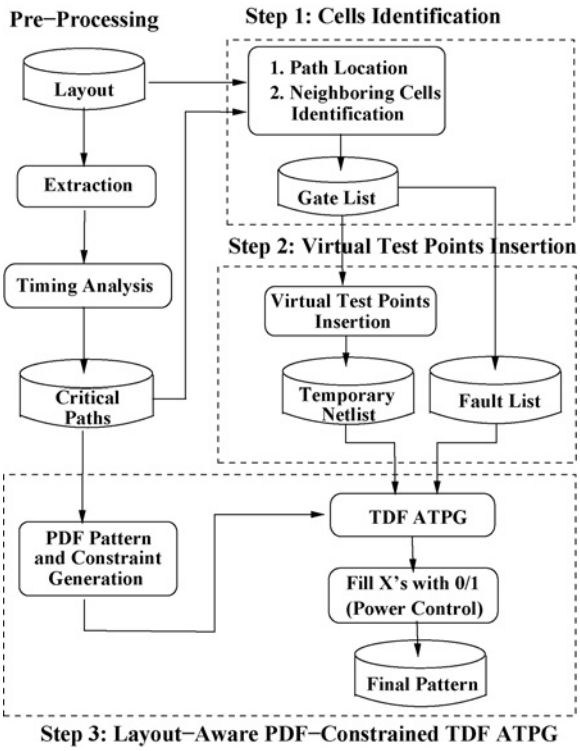

Fig. 5. Flow diagram of the layout-aware pattern generation method to maximize PSN on target critical paths.

generated from the layout design. It contains the physical placement information of the elements in the circuit. The library exchange format (LEF) file contains physical layout information of the standard cells, such as cell's width and height. We develop an in-house tool *DEFParser* in *C* programming language. The inputs to the *DEFParser* are DEF file, LEF File, and path file. By parsing the DEF file, the physical location of the cells on the specified critical path is identified. Also, the *DEFParser* gains information of the power distribution network information of the design through the DEF file. It collects information of the physical location for power ring, power straps, and power rails in the layout.

As discussed in Section II, to maximize the voltage drop effects on each cell of the critical path, an effective approach is to generate transitions on the nearby cells which share the same power/ground rail. Therefore, for each cell on the path, we also identify neighboring cells that are within a pre-defined proximity and are connected to the same power/ground rail.

Fig. 6 shows a cell *G*, referred to as victim cell, in the critical path on which we intend to generate a high IR-drop voltage on both its power and ground pins. As seen in Fig. 6, the victim cell *G* is close to via *ViaA*. The cells that have common resistors on their supply current path with that of the victim cell are called aggressor cells. The switch current of aggressor cells contributes to the voltage drop of the victim cell. The range of the aggressor cells for the victim cell *G* on critical path is shown with the dashed box in Fig. 6. Three power vias are included in this range, so we name it as range "*R3*." The aggressor cells share common power and ground rails (cells in Rows 1–3 in Fig. 6) with that of the victim cell. When victim cell *G* switches, it draws most of the current

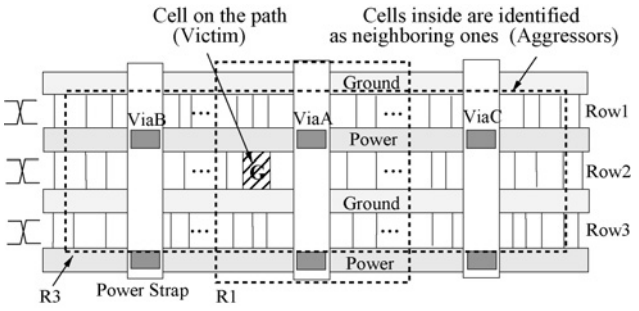

Fig. 6. Cell on critical path (victim) and its neighboring cells (aggressors) identified to be included as fault sites during layout-aware pattern generation procedure to increase the IR-drop for the cell on critical path.

from its nearest power via *ViaA* because of its lowest resistive path to the global PDN; and the two neighboring power vias *ViaB* and *ViaC* also provide small portion of the current so the cells around these two power vias are also taken as the aggressors. In this way, we extract a gate list which contains all victim cells and aggressor cells for the targeted critical path. The number of aggressors for each victim cell on critical path depends on the granularity of the power grid. In our design, there are around 20 cells in between two neighboring power vias in each row and about 180 aggressor cells in range *R3* for each victim cell.

We can improve the efficiency and reduce the complexity of the proposed procedure by reducing the total aggressors counts. There are two ways to do this: 1) setting a threshold on gate size to exclude small gates from the aggressors list, such as minimum sized inverters and buffers. By doing this, the ATPG in Step 3 would target less TDF fault sites and put more efforts on causing switching on larger gates which bring more switching current around the critical path, and 2) decreasing the range of the aggressor cells to include only the cells around the nearest power via of the victim cell. In Fig. 6, Region *R1* defines a smaller aggressor cell range for the victim cell *G*. In this way, the number of aggressor cells is reduced to about 1/3 of range *R3*. The performance of the two techniques will be presented and compared at Section IV.

B. Virtual Test Points Insertion

TDF ATPG is used to fill in don't-care bits in the target path's PDF pattern to generate a pattern with more switching on aggressor cells identified in Step 1. Outputs of all selected gates are considered as fault sites. All the flip-flops among these cells are considered as observation points during TDF pattern generation. For TDF ATPG, a transition fault will be considered detected only when it is activated at the fault site and propagated to an observation point. Since only the actual switching is necessary for this application, propagating the transition to an observation point is unnecessary and may create additional care-bits that could be better used to activate a transition at another fault site. Also, the transition may only be activated at the fault site but cannot be propagated. In this case, the pattern is not kept by the ATPG although it can induce transition at the fault site.

To avoid these issues, we alter the netlist to insert virtual test points at the output of gates for these cells. The effect of the alteration is only for TDF pattern generation and has no impact on the final layout of the chip and actual location of the gates.

The virtual test points provide new observation points to: 1) eliminate the effort that ATPG needs to propagate the transition to primary outputs or flip-flops; 2) eliminate the "not observable (NO)" ATPG untestable faults since a transition at the fault site should be immediately observable with the altered netlist; 3) reduce the number of care-bits necessary in the pattern since fewer gates need to be controlled by the ATPG; and 4) increase the number of transitions.

The new netlist with virtual test points inserted (referred to as temporary netlist) is used during ATPG for pattern generation in our proposed procedure, which is described below.

C. PDF-Constrained TDF ATPG

A PDF pattern is generated first to provide constraints for TDF pattern generation, which ensures the pattern generated using this method can test the delay of the critical path under consideration. All don't-care bits of the PDF pattern are kept unfilled. The pattern is generated using the LOC method to make it compatible with TDF pattern generation. From the PDF pattern, any state filled with a care-bit is then extracted and utilized as a constraint during TDF pattern generation.

To generate switching activity that can cause a greater IR-drop on power line and ground line, both slow-to-rise and slow-to-fall faults are added at aggressor cells. In this way, the TDF fault list from the gate list identified in Step 1 for this victim cell is generated. This procedure is conducted for all cells on a critical path. If more than one cell in a critical path are connected to the same power/ground rail, then the nearby cell (aggressor cell) region is determined based on the proximity of the victim cells. If the victim cells are very close, then their nearby cell region may overlap, resulting in some shared aggressor cells. If they are far from each other while connected to the same power/ground rail, each victim cell will have a distinct set of aggressor cells.

With the temporary netlist and the fault list, TDF ATPG using the PDF care-bits as constraints is performed to generate a pattern to test the path and increase the switching activities on its aggressor cells. However, there are several aspects need to be noted for the TDF ATPG process.

1) Not all the TDF fault sites can be activated because of circuit topology and the limited controllability on some aggressor cells.
2) Since the current sinking abilities are different for aggressor cells with different sizes, larger gates usually have large current requirements when switching.
3) For same cell size, the location of the aggressor cell, i.e., its relative distance to the victim cell and to the power via, makes its switching on victim's voltage drop vary. The aggressor cells closer to victim cell and power via are expected to induce larger voltage drop effect when switching based on our analysis in Section II-B.
4) The timing of aggressor cells' switching determines the worst IR-drop time on the victim cell; the closer the worst IR-drop time is to the victim cell's switching time,

the larger delay impact the aggressor cells have on the victim cell.

5) The timing of an aggressor cell's switching depends on how the TDF faults are activated. For a transition launched at the same clock edge on the output pin of the launch flip-flop, the number and type of gates and interconnects before the TDF fault site, determine the transition time on it. A TDF fault could be sensitized though different paths. Another issue is that timing information of the transition will not be accurate, due to the presence of large process variations especially in lower technology nodes. Due to these reasons, the transition time of the aggressor cells is not predictable.
6) There are conflict transition requirements on some of the cells and the ATPG algorithm used by the tool can only choose one to satisfy. So there will be uncertainties about which aggressor cells can be activated during the TDF ATPG.

Considering 1) and 2), we need to target large aggressor cells in close proximity could be activated by the TDF ATPG. However, to achieve this goal, TDF ATPG may sacrifice the switchings of more smaller aggressor cells, which could potentially induce larger impact on victim's voltage drop through simultaneous switching. Because of the uncertainties of the TDF ATPG stated in 3) to 5), it is impossible to designate which and how many aggressor cells to be activated that can give the best voltage drop result. A full layout-aware deterministic TDF ATPG would be extremely difficult to achieve because of the reasons mentioned above. However, it is possible to develop complex models to quantify the aggressor's switching effect on victim's voltage drop, and then use a brute-force search to find the best pattern over large number of patterns, e.g., N-detect patterns and randomly generated patterns. The model must take into account both gate size and distance to victims. We leave this to our future work. In this paper, for the aggressor cells in R3 range including three neighboring power vias, we give them equal priority during the TDF ATPG. In this way, the ATPG engine has more flexibility to activate more aggressor cells, thus larger chance to induce larger voltage drop around critical path.

The don't-care bits in this pattern can be utilized to adjust its power level by adopting various filling methods, such as using 0/1/adjacent fill methods for low power testing, or controlling the bit 0/1 ratio during filling to achieve the desired power level. There are many low power filling methods [14], [16], [27]–[29] and can be combined with the proposed procedure. However, the discussion of filling methods is beyond the scope of this paper.

IV. Experimental Results

We implemented the layout-aware pattern generation method on ITC'99 benchmark *b19*, which has 190 213 gates and 6642 flip-flops. The physical layout was designed using the 180 nm Cadence Generic Standard Cell Library [30] with 1.8 V as its nominal supply voltage. Critical paths were identified with a commercial static timing analysis tool. The program for parsing the DEF file to extract the victim and aggressor cell list was coded in *C*.

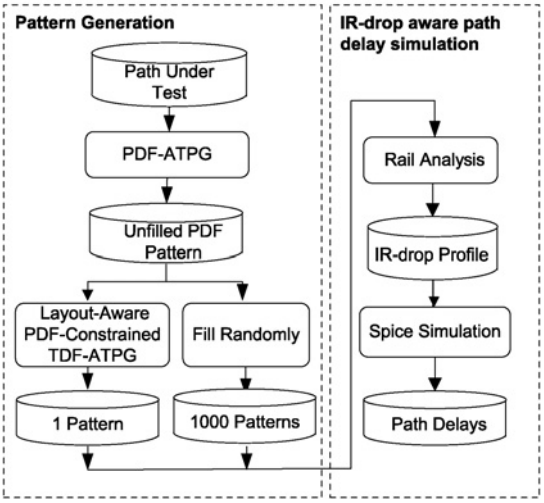

Fig. 7. Flow diagram of pattern validation process.

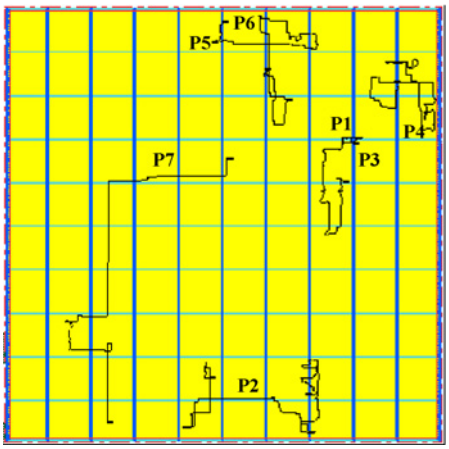

Fig. 8. Locations of seven paths under test on the layout of b19 circuit.

A pattern validation process is performed using the flow in Fig. 7 to verify that the pattern generated by the proposed procedure can induce worst-case PSN more effectively compared with that from the layout-unaware ATPG in commercial tool. As shown in the flow diagram, in the pattern generation step, we first run PDF ATPG for the path under test and generate a PDF pattern leaving the don't-care bits unfilled. Then a pattern is generated for the path under test by the proposed layout-aware pattern generation method, using the care bits as constraints. On the other hand, a random-search method is used for comparison purpose which is to find the worst-case test pattern from a large number of randomly-filled patterns. To generate these random patterns, we fill the unfilled PDF pattern randomly 1000 times to generate 1000 PDF patterns for the same path. Note that to make a fair comparison with the random-search method, we also fill the pattern randomly during the PDF-constrained TDF ATPG (Step 3 in Fig. 7).

After generating the patterns, we run IR-drop aware path delay simulation for these patterns to analyze and compare the pattern induced PSN impact on path delay. A hybrid of rail analysis and SPICE simulation is adopted in this paper, because full circuit SPICE simulation for benchmark

TABLE I
VICTIMS AND AGGRESSORS (USING *R3* RANGE) INFORMATION FOR SEVEN PATHS IN *b19* BENCHMARK

Path	# of Victims	# of Aggressors	# of Aggressor Activated	
			w/o Virtual TPI	w/ Virtual TPI
P1	45	5198	780	951 (22%)
P2	45	4111	327	410 (25%)
P3	45	5407	766	884 (15%)
P4	43	2567	198	273 (38%)
P5	42	3418	307	519 (69%)
P6	42	3627	251	416 (65%)
P7	19	2007	268	334 (25%)

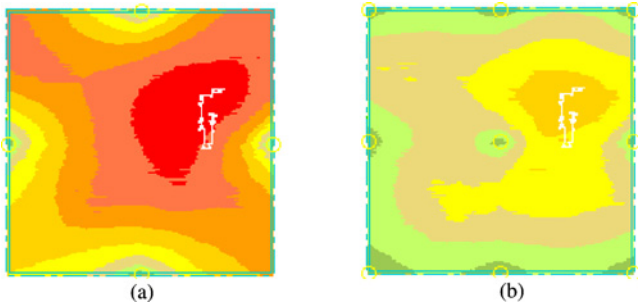

Fig. 9. IR-drop plot on power pin for path *P1* with pattern generated using our procedure in (a) wire-bond package and (b) flip-chip package.

b19 is extremely slow. We run power/ground rail analysis using a commercial tool for each pattern. Then the instance power/ground voltage profiles were extracted by a *Perl* script.

In [9], it has been proven effective and sufficiently accurate to use an average voltage drop value to represent the voltage drop effect in the timing cycle. According to the simulation waveforms, we observed that most of the switchings happen in the first half clock cycle. For each gate on the path, we subtract the average IR-drop in the first half clock cycle from the ideal supply voltage to account for the dynamic voltage drop over the timing cycle. The new supply voltages are back-annotated to SPICE netlist using a *Perl* script we developed. Using new supply voltages for each instance, SPICE simulation of the critical path was conducted to analyze the path delay. In this way, we can simulate path delay for each pattern regarding the path under test and compare their PSN effects. Note that the IR-drop aware path delay simulation is only done for the purpose of *validation* and such analysis is not necessary in practice.

After the patterns are validated, the CPU run time of the proposed procedure is presented and compared with the random-search method. At the end of this section, we also compare the path delay of patterns generated with aggressor reduction techniques, such as applying gate size threshold and using smaller aggressor range.

We have run the proposed layout-aware pattern generation procedure on seven paths placed at different locations in benchmark *b19* layout. The physical locations of the seven paths are illustrated in Fig. 8. Table I summarizes the victims (Column 2) and aggressors (Column 3) information for these paths. Aggressors are in the range of *R3* as shown in Fig. 6. Columns 4 and 5 list the number of aggressors activated by the patterns generated with/without virtual TPI. The percentage of

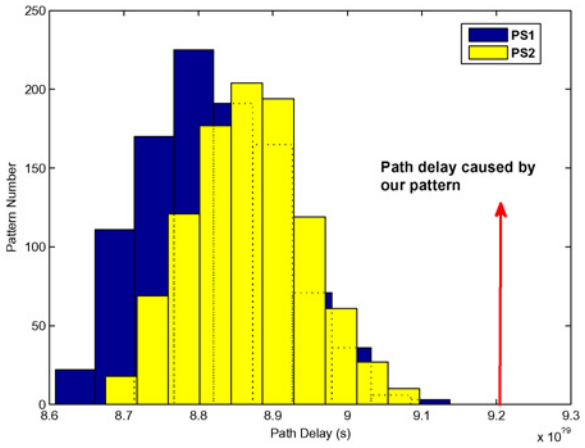

Fig. 10. Path delay distribution for path *P1* with random-filled PDF patterns from PS1, PS2, and the pattern generated by our proposed method in wire-bond package.

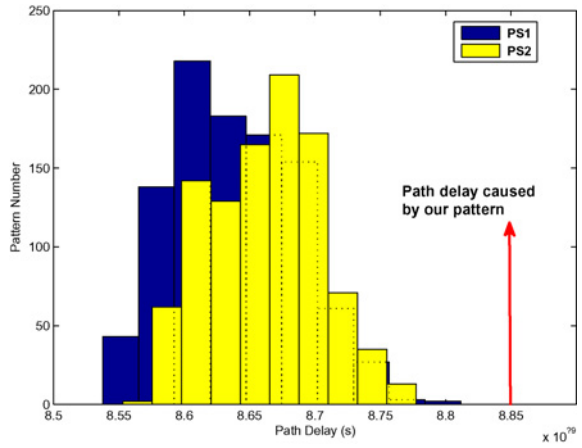

Fig. 11. Path delay distribution for path *P1* with random-filled PDF patterns from PS1, PS2, and the pattern generated by our proposed method in flip-chip package.

increase by virtual TPI is also shown in Column 5. Up to 69% more aggressors are activated with virtual TPI technique.

Fig. 9 presents the IR-drop plots for b19 using the pattern generated with the proposed method targeting critical path *P1*. IR-drop plots on power pins for both wire-bond and flip-chip packaging styles are shown in this figure. The darkest region suffers IR-drop voltage over 90 mV. In Fig. 9(a), there were four pairs of power/ground pads in the wire-bond chip, placed in the middle of each side of the chip. The effective voltage drop on both power pin and ground pin for cells in the same region is around 180 mV, which is 10% of the ideal supply voltage. Path *P1* has been highlighted in the IR-drop plots to show how the switching was targeted around it. The flip-chip, shown in Fig. 9(b), uses nine C4 bumps distributed across the chip. The color map indicates that the voltage drop on power pin is around 50 mV for cells around path *P1*.

We compared the path delay generated by our pattern with 1000 randomly-filled PDF patterns. Two randomly-filled PDF pattern sets are used for the comparison. Each pattern set consists of 1000 patterns. Patterns in pattern set 1 (PS1) are filled 1000 times with bit "0" probability increasing from 0%

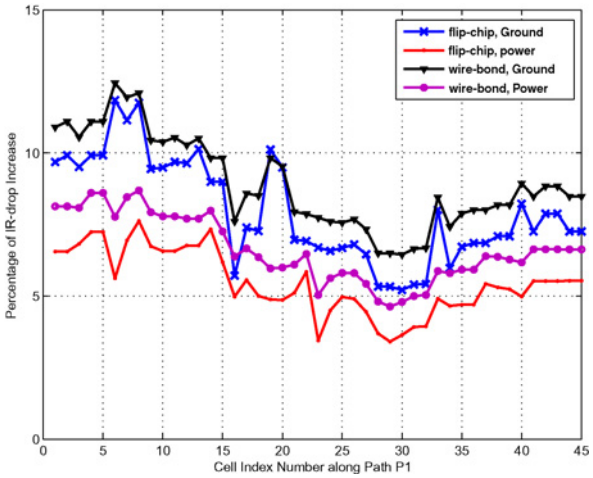

Fig. 12. Percentage of IR-drop increase (our pattern versus best random pattern from PS1) on power and ground pins of cells along path *P1* in wire-bond and flip-chip packages.

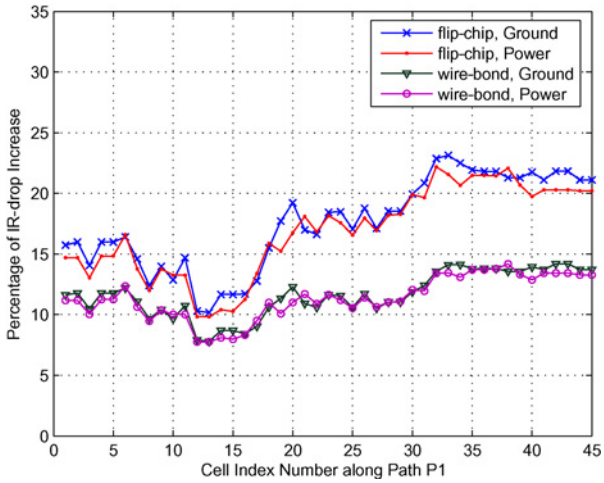

Fig. 13. Percentage of IR-drop increase (our pattern versus best random pattern from PS2) on power and ground pins of cells along path *P1* in wire-bond and flip-chip packages.

(1-fill) to 100% (0-fill); while patterns in pattern set 2 (PS2) are filled 1000 times with 50% probability of bit 0/1, which is the same 0/1 ratio used by the commercial ATPG tool. We simulate the path delay for both pattern sets for Path *P1*.

Figs. 10 and 11 illustrate the slow-to-fall path delay distribution for wire-bond and flip-chip packaging styles, respectively. Both figures show that the path delays caused by patterns generated by our method are larger than that of any random patterns. As expected, we can see that PS2 has a larger mean value, implying that 50% filling method induces a higher PSN on average. However, PS1 has a larger variance which means we will have a larger chance to find the worst-case PSN pattern by increasing the search space to a large number (i.e., number of patterns). The pattern generated by the proposed method could be one of the random patterns and would be included in the pattern set if we increase the search space when filling the PDF pattern to generate random patterns. In other words, the worst-case pattern could be one of the random patterns encountered in real application; and if the timing margin is not properly set, this pattern could fail the chip. The advantage of the proposed layout-aware pattern generation procedure is that it can generate the worst-case PSN pattern using a deterministic method in a much shorter time.

We compare the effective IR-drop voltages of cells on the critical path *P1* when applying our pattern and when applying the *best* (largest delay) pattern of 1000 random patterns for both pattern sets. Fig. 12 shows the percentage of IR-drop increase on each cell compared with the best random pattern of PS1. Results for both wire-bond and flip-chip packages are presented. Compared with the best random pattern from PS1, on average our pattern can increase the IR-drop on power and ground pin together by 15.6% for the wire-bond design and 13.4% for flip-chip design, respectively. Similar results are presented in Fig. 13 when compared with the best pattern from PS2 except for the higher percentage increase in this case. These results demonstrate the effectiveness of the proposed method on maximizing the supply noise around the critical path.

Table II shows the simulation results of average IR-drop (Columns 2, 3, 6, 7) and path delay (Columns 4, 5, 8, 9) for seven critical paths on b19 in both wire-bond and flip-chip packing styles. The average IR-drop reported is calculated as effective IR-drop on both power and ground pins across all cells on the critical path. The results for the best random pattern of PS2 and the pattern generated by the proposed method are shown. The best pattern of PS2 is taken for comparison because its filling pattern is same as the commercial ATPG tools.

For path *P1*, the path delay for circuit with ideal supply voltage (no IR-drop) is 8.422 ns. Our pattern increases path delay by 0.788 ns which is 16.9% higher than that increased by the best random pattern in wire-bond chip. In flip-chip our pattern increases path delay 19.4% higher than that increased by the best random pattern. Comparing the results for both average IR-drop and path delay, we can see our pattern induces larger PSN and longer path delay than the best random pattern for the targeted critical paths. The total switching activities (number of switching nets) for both best random patterns and our patterns are also compared in Table II. From the numbers listed in Columns 10 and 11, we can see that the patterns generated using the proposed flow have similar or lower switching activities than the best random patterns. Even though they induce larger PSN and longer path delay for the targeted critical paths, they only increase the local switching activity instead of the global switching activity.

Table III lists the number of care and don't-care bits for the layout-aware patterns generated for the seven paths. There are totally 6049 bits for both scan inputs and primary inputs for the b19 benchmark circuit. The number of care-bits for the PDF patterns are listed in Column 2. The care-bits in the PDF patterns are extracted as constraints to be used in TDF ATPG to generate the final patterns. They are used to sensitize the paths during test. More care-bits are generated during the PDF-constrained TDF ATPG to activate the aggressors. Column 3 lists the care-bits count in the final patterns. The number of don't-care bits is shown in Column 4. The ratio is also given in paired parenthesis. The ratio of don't-care bits in the final pattern is still quite high (around 90%) for the final patterns.

TABLE II
AVERAGE IR-DROP AND PATH DELAY ANALYSIS FOR LAYOUT-AWARE PDF-CONSTRAINED TDF PATTERN AND BEST RANDOM PATTERN OF PS2 IN *b19* BENCHMARK

Path	Wire-Bond				Flip-Chip				Switching Activities	
	Average IR-Drop (mV)		Path Delay (ns)		Average IR-Drop (mV)		Path Delay (ns)		(# of Switching Nets)	
	Random	Our Pattern	Random	Our Pattern	Random	Our Pattern	Random	Our Pattern	Random	Our Pattern
P1	154.5	172.1	9.096	9.210	86.6	100.3	8.777	8.846	85 639	42 088
P2	128.9	154.4	9.716	9.716	61.3	74.5	9.268	9.353	63 552	69 592
P3	154.8	181.4	8.827	8.934	82.5	105.0	8.5621	8.642	56 280	41 025
P4	151.0	168.4	8.744	8.812	72.4	78.2	8.474	8.494	71 105	49 647
P5	111.1	122.0	8.586	8.625	41.9	62.8	8.403	8.424	54 275	65 817
P6	102.9	109.9	8.582	8.609	52.9	56.6	8.408	8.420	57 000	41 361
P7	128.3	152.7	5.076	5.122	62.6	77.8	4.911	4.929	77 284	49 961

Aggressor range *R3* is used.

TABLE III
CARE AND DON'T-CARE (X) BITS (USING *R3* RANGE) INFORMATION FOR THE SEVEN PATTERNS GENERATED FOR SEVEN PATHS IN *b19* BENCHMARK

Path	PDF Pattern	Final Pattern	
	# of Care-Bits	# of Care-Bits	# of X-Bits
P1	75	617	5432 (90%)
P2	75	427	5622 (93%)
P3	75	694	5355 (89%)
P4	68	383	5666 (94%)
P5	67	605	5444 (90%)
P6	67	650	5399 (89%)
P7	85	430	5619 (93%)

TABLE IV
CPU RUN-TIME BREAKDOWN AND COMPARISON: LAYOUT-AWARE PDF-CONSTRAINED TDF ATPG VERSUS RANDOM-SEARCH IN *b19* BENCHMARK FOR SEVEN CRITICAL PATHS

Path	Layout-Aware PDF-Constrained TDF ATPG		Random
	Step 1/2/3 (s)	Total Time (s)	Search (h)
P1	2/236/43	281	49
P2	2/180/43	225	50
P3	2/251/43	296	49
P4	2/105/42	149	48
P5	2/139/43	184	49
P6	2/150/43	195	50
P7	2/82/43	127	48

These large number of don't-care bits can be exploited to control the total (global) switching activities to the level of functional mode using various X-filling techniques.

The proposed method consumes considerably less CPU run time to generate a pattern than the random-search method that may provide the longest path delay by simulating a large number (1000) of random patterns. Table IV shows the CPU run time breakdown (Column 2) for the layout-aware pattern generation method and compares its total time (Column 3) with finding the worst-case pattern from 1000 randomly-filled PDF patterns (Column 4). The run time for PDF-constrained TDF ATPG for Step 1 (aggressor identification) and Step 3 (PDF-constrained TDF ATPG) is almost a fixed value for all the seven paths. The total time of the proposed method is dominated by Step 2 (virtual TPI) which is approximately in direct proportion to the aggressors number. From the results we can see the proposed method can generate one pattern in several minutes for a critical path that is around 1000× faster than the random-search. The latter one needs days to run the hybrid SPICE simulation for a large number of random patterns and select the one with the longest delay. When targeting more paths, the average CPU run time will decrease because of the shared setup time. Note that in this paper we speed up the SPICE simulation in this paper with our hybrid simulation flow. If full-circuit SPICE simulation is used, the CPU run time will be prohibitively high.

The aforementioned results demonstrate the effectiveness of the proposed procedure when taking all the gates in range *R3* as aggressors. To investigate the impact of aggressor count reduction techniques on the performance of the proposed procedure, we run the pattern generation and validation flow twice with less aggressors by: I) setting a threshold on the gate size to target only large gates in range *R3*, and II) using the smaller aggressor range *R1* (shown in Fig. 6). Table V summarizes the performances of the two aggressor reduction techniques (Columns 5–10) and compares them with the case without aggressor reduction (Columns 2–4). Comparing the aggressor counts in Columns 5 and 8 with that in Column 2, we can see that technique II reduces the aggressors number more significantly than technique I does. As for the path delay generated, technique I has better performance for both wire-bond and flip-chip on most of the paths than technique II. However, both of them degrade the performance in terms of path delay compared with the results in Columns 3, 4 which takes all the gates in range *R3* as aggressors. In a nutshell, the reduction of complexity is achieved at the cost of performance degradation.

Note that a fair comparison with previously proposed methods [19]–[21] cannot be made since those were implemented on different benchmarks and used different technology library nodes. However, since our proposed method is not iterative, it is significantly faster than previous approaches. Another important distinction is that our method uses actual physical design information in addition to localized switching impact rather than only targeting increase in global switching activity. The pattern generated by this method only increases the

TABLE V
PERFORMANCE COMPARISON WHEN USING AGGRESSORS REDUCTION TECHNIQUES I AND II FOR SEVEN PATHS IN *b19* BENCHMARK

Path	R3 Region w/o Gate Size Threshold			I: R3 Region w/ Gate Size Threshold			II: R1 Region w/o Gate Size Threshold		
	# of Aggr.	Wire-Bond (ns)	Flip-Chip (ns)	# of Aggr.	Wire-Bond (ns)	Flip-Chip (ns)	# of Aggr.	Wire-Bond (ns)	Flip-Chip (ns)
P1	5198	9.210	8.846	2575	9.052	8.747	2068	8.999	8.748
P2	4111	9.716	9.353	2063	9.571	9.289	1588	9.496	9.269
P3	5407	8.934	8.642	2666	8.855	8.594	2134	8.854	8.588
P4	2567	8.812	8.494	1206	8.635	8.417	1201	8.531	8.386
P5	3418	8.625	8.424	1679	8.624	8.434	1328	8.480	8.377
P6	3627	8.609	8.420	1735	8.696	8.407	1448	8.442	8.283
P7	2007	5.122	4.929	1070	5.042	4.898	887	5.074	4.9143

switching activities around the target path, while the global switching activity and the test power consumption can be manipulated by filling the don't-care bits properly.

The proposed method is also applicable to scan compression environment to generate compressed test patterns. To generate patterns for compressed test, PDF ATPG is first run in bypass mode to generate unfilled PDF patterns. Then the care-bits of the PDF patterns are used as constraints in compressed mode to run TDF ATPG and generate compressed test patterns for the critical paths under test.

V. CONCLUSION

In this paper, we presented a novel layout-aware pattern generation procedure for testing critical paths under maximum PSN. It uses a deterministic method during pattern generation thus can generate a worst-case pattern per path in a short CPU time. The localized IR-drop effect and its impact on path delay have been analyzed. To maximize the IR-drop voltage on cells along critical paths, we selected cells in close proximity as aggressor cells to generate switching activity for each cell on the path. By increasing the switching activity around each cell on the critical path, the pattern generated by our method effectively increases the IR-drop effect along that path. The experimental results show that the layout-aware PDF-constrained TDF ATPG is able to increase the path delay significantly for both wire-bond and flip chips. It is considerably faster compared with those methods that select a random pattern with worst voltage drop effects from a large set of random patterns after simulation. The pattern generated by the proposed method can provide a realistic estimation of worst-case delay for the critical paths.

REFERENCES

[1] A. H. Ajami, K. Banerjee, A. Mehrotra, and M. Pedram, "Analysis of IR-drop scaling with implications for deep submicron P/G network designs," in *Proc. 4th ISQED*, 2003, pp. 35–40.

[2] R. Saleh, S. Z. Hussain, S. Rochel, and D. Overhauser, "Clock skew verification in the presence of IR-drop in the power distribution network," *IEEE Trans. Comput.-Aided Des.*, vol. 19, no. 6, pp. 635–644, Jun. 2000.

[3] S. Pant, D. Blaauw, V. Zolotov, S. Sundareswaran, and R. Panda, "Vectorless analysis of supply noise induced delay variation," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, Nov. 2003, pp. 184–191.

[4] C. Tirumurti, S. Kundu, S. K. Susmita, and Y. S. Change, "A modeling approach for addressing power supply switching noise related failures of integrated circuits," in *Proc. Des. Autom. Test Eur. Conf.*, 2004, pp. 1078–1083.

[5] R. Senthinatharr and J. L. Prince, *Simultaneous Switching Noise of CMOS Devices and Systems*. Boston, MA: Kluwer, 1994.

[6] S. Lin and N. Chang, "Challenges in power-ground integrity," in *Proc. ICCAD*, 2001, pp. 651–654.

[7] L. Zheng, B. Li, and H. Tenhunen, "Efficient and accurate modeling of power supply noise on distributed on-chip power networks," in *Proc. Int. Symp. Circuits Syst.*, 2000, pp. 513–516.

[8] D. Mitra, S. Bhattacharjee, S. Sur-Kolay, B. B. Bhattacharya, S. T. Zachariah, and S. Kundu, "Test pattern generation for power supply droop faults," in *Proc. Int. Conf. VLSI Des.*, 2006, pp. 343–348.

[9] K. Arabi, R. Saleh, and X. Meng, "Power supply noise in SoCs: Metrics, management, and measurement," *IEEE Des. Test Comput.*, vol. 24, no. 3, pp. 236–244, May–Jun. 2007.

[10] J. Wang, D. M. Walker, X. Lu, A. Majhi, B. Kruseman, G. Gronthoud, L. E. Villagra, P. J. A. M. van de Wiel, and S. Eichenberger, "Modeling power supply noise in delay testing," *IEEE Des. Test*, vol. 24, no. 3, pp. 226–234, May–Jun. 2007.

[11] J. J. Liou, A. Kristic, Y. M. Jiang, and K. T. Cheng, "Modeling, testing, and analysis for delay defects and noise effects in deep submicron devices," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 22, no. 6, pp. 756–769, Jun. 2003.

[12] Y. M. Jiang and K. T. Cheng, "Analysis of performance impact caused by power supply noise in deep submicron devices," in *Proc. 36th Annu. ACM/IEEE Des. Autom. Conf.*, Jun. 1999, pp. 760–765.

[13] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreeprakash, and M. Hachinger, "A case study of IR-drop in structured at-speed testing," in *Proc. Int. Test Conf.*, Oct. 2003, pp. 1098–1104.

[14] P. Girard, "Survey of low-power testing of VLSI circuits," *IEEE Des. Test Comput.*, vol. 19, no. 3, pp. 82–92, May–Jun. 2002.

[15] K. M. Butler, J. Saxena, T. Fryars, and G. Hetherington, "Minimizing power consumption in scan testing: Pattern generation and DFT techniques," in *Proc. Int. Test Conf.*, 2004, pp. 355–364.

[16] X. Wen, S. Kajihara, K. Miyase, T. Suzuki, K. K. Saluja, L.-T. Wang, K. S. Abdel-Hafez, and K. Kinoshita, "A new ATPG method for efficient capture power reduction during scan testing," in *Proc. VLSI Test Symp.*, 2006, pp. 58–63.

[17] V. R. Devanathan, C. P. Ravikumar, and V. Kamakoti, " On power-profiling and pattern generation for power safe scan tests," in *Proc. DATE*, 2007, pp. 534–539.

[18] J. Lee, S. Narayan, M. Kapralos, and M. Tehranipoor, " Layout-aware, IR-drop tolerant transition fault pattern generation," in *Proc. DATE*, 2008, pp. 1172–1177.

[19] M. Nourani, M. Tehranipoor, and N. Ahmed, "Pattern generation and estimation for power supply noise analysis," in *Proc. 23rd IEEE VLSI Test Symp.*, May 2005, pp. 439–444.

[20] J. J. Liou, A. Krstic, Y. M. Jiang, and K. T. Cheng, "Path selection and pattern generation for dynamic timing analysis considering power supply noise effects," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, Nov. 2000, pp. 493–496.

[21] A. Krstic, Y. M. Jiang, and K. T. Cheng, "Pattern generation for delay testing and dynamic timing analysis considering power-supply noise effects," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 20, no. 3, pp. 416–425, Mar. 2001.

[22] J. Jiang, K. Cheng, and A. Krstic, "Estimation of maximum power and instantaneous current using a genetic algorithm," in *Proc. Custom Integr. Circuits Conf.*, 1997, pp. 135–138.

[23] S. Zhao and K. Roy, "Estimation of switching noise on power supply lines in deep sub-micro CMOS circuits," in *Proc. 13th Int. Conf. VLSI Des.*, 2000, pp. 168–173.

[24] Z. Zhang, S. M. Reddy, and I. Pomeranz, "On generating pseudo-functional delay fault tests for scan designs," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Syst.*, Oct. 2005, pp. 398–405.
[25] X. Liu and M. S. Hsiao, "On identifying functionally untestable transition faults," in *Proc. IEEE Int. High-Level Des. Validation Test Workshop*, Nov. 2004, pp. 121–126.
[26] C. Visweswariah, K. Ravindran, K. Kalafala, S. G. Walker, S. Narayan, D. K. Beece, J. Piaget, N. Venkateswaran, and J. G. Hemmett, "First-order incremental block-based statistical timing analysis," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 25, no. 10, pp. 2170–2180, Oct. 2006.
[27] S. Wang and S. K. Gupta, "ATPG for heat dissipation minimization during test application," *IEEE Trans. Comput.*, vol. 47, no. 2, pp. 256–262, Feb. 1998.
[28] S. Remersaro, X. Lin, Z. Zhang, S. M. Reddy, I. Pomeranz, and J. Rajski, "Preferred fill: A scalable method to reduce capture power for scan based designs," in *Proc. IEEE Int. Test Conf.*, Oct. 2006, pp. 1–10.
[29] W. Li, S. M. Reddy, and I. Pomeranz, "On reducing peak current and power during test," in *Proc. IEEE Symp. VLSI*, May 2005, pp. 156–161.
[30] Cadence, Inc. (2005). *0.18 um Standard Cell GSCLib Library Version 2.0* [Online]. Available: http://crete.cadence.com
[31] *PrimeTime SI User Guide, Version D-2010.06*, Synopsys, Inc., Mountain View, CA, Jun. 2010.

Junxia Ma (S'08–M'11) received the B.Sc. degree in electrical engineering from Beihang University, Beijing, China, in 2003, the M.E. degree in electrical engineering from the University of Macau, Macao, in 2006, focusing on high-speed analog-to-digital converter design, and the Ph.D. degree in electrical and computer engineering from the University of Connecticut, Storrs, in 2010, researching layout-aware delay fault testing techniques considering signal and power integrity issues.

She was a Member of the Technical Staff with Cadence Design Systems, Inc., Beijing, from 2006 to 2007. She is currently with LSI Corporation, Milpitas, CA, as a Senior Methodology Design Engineer in the Test Methodology Group. Her current research interests include delay test, low-power test, power, and signal integrity issues in nanometer very large scale integration testing.

Dr. Ma was selected as a finalist for the Connecticut Women of Innovation Award in 2011 by the Connecticut Technology Council. Her presentation on "layout-aware test pattern generation to maximize power supply noise" received the Best in Session Award in the 2009 Semiconductor Research Corporation TECHCON, Austin, TX.

Mohammad Tehranipoor (S'02–M'04–SM'07) is currently an Associate Professor of Electrical and Computer Engineering with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs. He has published over 140 journal articles and refereed conference papers, two books, and seven book chapters. His current research interests include computer-aided design and test, reliability analysis, and hardware security and trust.

He was a recipient of the Best Paper Award at the 2005 VLSI Test Symposium (VTS), the Best Paper Award at the 2008 North Atlantic Test Workshop (NATW), the Best Paper Award at NATW-2009, Honorable Mention for the Best Paper Award at NATW-2008, the Best Paper Candidate at the 2006 Design Automation Conference (DAC), and the Best Panel Award at VTS-2006. He was also a recipient of the 2008 IEEE Computer Society Meritorious Service Award, the 2009 NSF CAREER Award, and the 2009 UConn ECE Research Excellence Award. He serves on the program committees of several leading conferences and workshops. He served as the Program Chair of the 2007 IEEE Defect-Based Testing Workshop, the Program Chair of the 2008 IEEE Defect and Data Driven Testing (D3T), the Co-Program Chair of the 2008 International Defect and Fault Tolerance in VLSI Systems, and the General Chair for D3T-2009 and DFT-2009. He co-founded a new workshop called the IEEE International Workshop on Hardware-Oriented Security and Trust (HOST) and served as the HOST-2008 and HOST-2009 General Chair and Chair of the Steering Committee. He is currently serving as an Associate Editor for the *Journal of Electronic Testing: Theory and Applications*, the IEEE DESIGN AND TEST OF COMPUTERS, and the *Journal of Low Power Electronics*. He is a member of ACM and ACM SIGDA.