Built-In Self-Test of High-Density and Realistic ILV Layouts in Monolithic 3-D ICs

Arjun Chaudhuri, *Member, IEEE*, Sanmitra Banerjee, Jinwoo Kim, Sung Kyu Lim, *Fellow, IEEE*, and Krishnendu Chakrabarty

***Abstract*— Nanoscale interlayer vias (ILVs) in monolithic 3-D (M3D) ICs have enabled high-density vertical integration of logic and memory tiers. However, the sequential assembly of M3D tiers via wafer bonding is prone to variability in the immature fabrication process and manufacturing defects. The yield degradation due to ILV faults can be mitigated via dedicated test and diagnosis of ILVs using built-in self-test (BIST). Prior work has carried out fault localization for a regular 1-D placement of ILVs in the M3D layout where shorts are assumed to arise only between unidirectional ILVs. However, to minimize wirelength in M3D routing, ILVs may be irregularly placed by a place-and-route tool, and shorts can also occur between an up-going ILV and a down-going ILV. To test and localize faults in realistic ILV layouts, we present a new BIST framework that is optimized for test time and PPA overhead. We also present a graph-theoretic approach for representing potential fault sites in the ILVs and carry out inductive fault analysis to drop noncritical sites. We describe a procedure for optimally assigning ILVs to the BIST pins and determining the BIST configuration for test-cost minimization. Evaluation results for M3D benchmarks demonstrate the effectiveness of the proposed framework.**

***Index Terms*— Built-in self-test (BIST), fault modeling, graph theory, heterogeneous, high-density integration, inter-layer vias (ILVs), monolithic 3-D (M3D).**

I. Introduction

THE emergence of sequential or monolithic 3-D (M3D) integration has enabled high-density vertical integration of heterogeneous technologies and advanced Moore's law by accommodating more transistors in the same die footprint compared to 2-D ICs [1]. Adjacent tiers in an M3D IC are separated by a thin interlayer dielectric (ILD), and the transistors are formed in the epitaxial silicon layers [2]. The source, drain, and gate terminals of the transistors in different tiers are connected via vertical interconnects, referred to as *interlayer vias* (ILVs), which penetrate the active silicon layer. The diameter and pitch of the ILVs are significantly smaller than that of the through-silicon vias in stacked 3-D integration. Consequently, the ILVs contribute lower capacitive load and reduce the overall wirelength and power consumption of the M3D circuit compared to 2-D and stacked 3-D designs [3]. This provides designers with increased flexibility for optimizing the placement of logic gates in the different tiers of the M3D design in order to maximize power–performance–area (PPA) benefits [4].

However, aggressive scaling of the ILD thickness makes ILVs especially prone to defects [5], [6]. The aggressive scaling of ILD is carried out to reduce the height of an ILV. This step leads to lower resistive–capacitive (RC) parasitics and wirelength-induced delay in the intertier connections passing through ILVs. Such a scaling effort is required for the commercialization of M3D ICs and motivates the need for comprehensive ILV-test mechanisms. In [7], challenges and solutions related to the fabrication of top-tier transistors are discussed. High-density placement of ILVs increases the likelihood of shorts between adjacent ILVs, especially when the design rules for minimum ILV pitch are not mature. Voids formed at the interface of ILD and the active layer can propagate to nearby ILVs via grain boundaries in the ILD and silicon. Fault effects due to ILV defects can propagate to active devices in the proximity via crystallographic imperfections in the resistive silicon epitaxial layer and adversely affect circuit performance [8]. As a result, targeted ILV testing is needed to ensure effective defect screening and quality assurance. While ILVs can be tested together with the M3D logic/memory tiers, defect isolation and yield learning require a solution that can test the ILVs in a dedicated manner.

The ILV-built-in self-test (BIST) method proposed in [9] and [10] assumes a 1-D arrangement of ILVs in a bus, where an ILV can be shorted to at most two ILVs, i.e., the adjacent left and right ILVs. However, during automatic algorithm-driven place-and-route by a commercial tool, the ILVs are not necessarily arranged in a 1-D array, as shown in Fig. 1. Here, the ILVs are highlighted in a two-tier M3D Rocketcore chip; the Nangate 45-nm open-source library is used for the chip design where the ILVs connect the metal layer-6 in the bottom tier's back-end-of-line (BEOL) to the metal layer-1 in top tier's BEOL. Such an irregular ILV arrangement implies that the number of potential ILV-to-ILV shorts can become significantly greater than that in a regular 1-D arrangement of ILVs. Therefore, while the ILV-BIST capture engine with N input pins in [10] can test all potential shorts in a single iteration for the 1-D ILV placement, many more iterations are needed to cover shorts in the case of irregular ILV placement. This necessitates optimization of the on-chip ILV-BIST engines

Manuscript received 12 August 2022; revised 9 November 2022; accepted 20 November 2022. Date of publication 17 January 2023; date of current version 24 February 2023. This work was supported in part by the DARPA ERI 3DSOC Program under Award HR001118C0096 and in part by the National Science Foundation under Grant CCF-1908045. *(Corresponding author: Arjun Chaudhuri.)*

Arjun Chaudhuri, Sanmitra Banerjee, and Krishnendu Chakrabarty are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708 USA (e-mail: ac499@duke.edu).

Jinwoo Kim and Sung Kyu Lim are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TVLSI.2022.3228850.

Digital Object Identifier 10.1109/TVLSI.2022.3228850

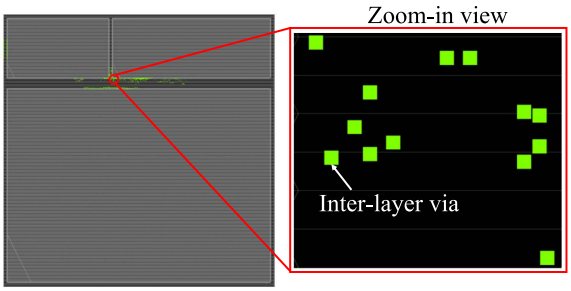

Fig. 1. Irregular ILV placement after place-and-route.

for reducing test time and hardware overhead. In addition, the prior BIST architecture cannot localize shorts between up-going and down-going ILVs because of the assumption that up-going and down-going ILV buses are physically placed far from each other.

In this article, we present a new BIST framework to detect and localize opens, stuck-at faults (SAFs), and bridging faults (shorts) in irregularly placed ILVs in the minimum possible number of test iterations. The key contributions are given as follows.

1) We present a reconfigurable BIST architecture that uses three test patterns for detecting and localizing SAFs, shorts, and opens—single and multiple—in irregularly placed ILVs (both up-going and down-going).
2) We present a graph-theoretic approach, which uses the *ILV defect graph*, for representing potential ILV shorts based on the physical ILV locations in the M3D layout.
3) We present a novel defect-level-aware ILV-testing procedure that prunes a defect graph based on the likelihood of short occurrence and driven by inductive fault analysis (IFA) [11].
4) We assign ILVs to the pins of the BIST capture engine for minimizing area overhead and the number of iterations required to test all potential ILV shorts and opens.
5) We present design-space exploration (DSE) of the optimized BIST architecture for selecting an effective design configuration.
6) We evaluate the BIST overhead for M3D benchmarks.

The remainder of this article is organized as follows. Section II presents an overview of M3D technology, placement of ILVs, and shortcomings of prior BIST and test solutions for ILV testing. Section III describes the proposed BIST framework for detecting and localizing faults in up-going and down-going ILVs that are placed in a dense and irregular arrangement. Section IV describes the generation and pruning of the ILV defect graph. Section V presents the methodology for determining optimal ILV-to-BIST assignment and BIST engine count for minimizing test time and area overhead. Section VI presents the evaluation results. Finally, Section VII concludes this article.

II. Background

A. M3D Fabrication Process

The first step in M3D fabrication involves a standard high-temperature process to integrate the transistors and associated interconnects in the bottom tier. A thin ILD is then created over the bottom tier's BEOL metal stack, and the low-temperature molecular bonding of the silicon-on-insulator (SOI) substrate is used to obtain the top tier's transistors [12], [13]. The ILVs are then fabricated to connect the BEOL metal stacks of the top and bottom tiers. The above steps are repeated for the fabrication of additional tiers.

B. M3D Routing and ILV Placement

A 1-D placement of TSVs is considered in [14] for online detection and mitigation of short and open defects in TSVs. Likewise, a 1-D array of ILVs has been assumed in prior BIST schemes for ILV faults where an ILV can be shorted to at most two ILVs in the vicinity [9], [10]. However, optimized M3D routing leads to ILV placements based on wirelength minimization and for obtaining timing closure. Pseudo-3-D flows for the physical design of M3D ICs extend the capabilities of commercial 2-D place-and-route tools for placement and routing of M3D designs [15]. In these flows, partitioning-first (partitioning-last) strategies are adopted where a 2-D netlist is partitioned before (after) commercial 2-D tool-driven placement; the partitioning-last approach leverages the 2-D placement information for partitioning the design into multiple tiers. The tier-partitioning algorithm determines the ILV count; the number of cuts made to divide a netlist graph into two partitions equals the number of ILVs. For example, a min-cut algorithm is typically used to limit the number of ILVs such that ILV faults are less likely to cause yield loss. The physical placement of ILVs takes place as part of the global 3-D routing procedure after the logic cells are partitioned and placed in different tiers. The ILVs are typically placed closer to their driving logic gates to reduce the timing delay of the paths passing through them [16].

The resulting ILV locations in the layout are not necessarily 1-D, or even regular as in a 2-D array, as they are determined by the pin locations of the standard cells in the different tiers, the cut locations during tier-partitioning, and the 3-D routing procedure and associated objective of PPA optimization. Consequently, the number of potential fault sites (especially shorts) increases drastically, which, in turn, can lead to higher area overhead for on-chip test and diagnosis methods. Therefore, there is a need for a low-cost BIST framework that can detect and localize faults for realistic ILV placements.

C. Dedicated Testing and Fault Localization Needs for ILVs

The M3D fabrication process involves the low-temperature deposition of a thin epitaxial film of silicon after the BEOL of the bottom tier has been fabricated. The deposition of silicon via wafer bonding, followed by grinding and etch-back [17], can form voids and delamination defects in the ILD that adversely affects ILVs [16]. In contrast, BEOL vias are surrounded by dielectric layers that do not contain any bonding interface. As a result, traditional BEOL vias are less susceptible to faults compared to ILVs.

The ILVs penetrate the thin-film silicon layer, which is both resistive and capacitive in nature [18]. Consequently, a void or defect in the ILV is likely to grow over time and propagate

to nearby active devices, i.e., top-tier transistors, resulting in timing degradation of the circuit. Therefore, there is a need to minimize the number of test escapes for ILV faults by employing a targeted BIST mechanism that can detect and localize faults in the ILVs. Such dedicated test and diagnosis schemes enable yield learning and provide feedback to the foundry for process rectification and revision of design rules.

During postbond or preassembly testing, the proposed BIST macro can screen faulty M3D dies with low test escape. Consequently, resources will not be spent on packaging a known bad die, thereby reducing the unit per hour (UPH) cost, assembly cycle time, and the assembly material cost. Targeted ILV BIST will also accelerate physical failure analysis once faults are localized during production testing (or postassembly testing). Assuming that spare ILVs are available, BIST will also enable in-field repair and recovery.

The ILV faults can become yield limiters in the early days of M3D fabrication. Hence, the targeted test for ILVs is needed for technology bring-up and yield learning. Sharing of logic or memory BIST for ILV test will lead to loss of diagnostic resolution as potential fault candidates (upon fault detection) will include logic gates along with ILVs leading to the increased difficulty of accurate root-cause analysis. The larger pool of candidate fault sites will also lead to an increase in the physical failure analysis effort needed for yield ramp-up.

D. ILV Fault Models

The typical fault models for an ILV are shorts, opens, and SAFs [6], [19]. Faults can be classified into hard and resistive categories based on the type and size of the underlying defects (root causes). Hard shorts can occur due to imperfect design rules followed during circuit layout and particle contamination during fabrication [20]. Resistive shorts can occur when the ILV metal diffuses through the ILD to make a partial contact with another nearby ILV [21] or due to defects at the bonding interface between two tiers [6]. A hard open occurs when a gap exists between the bottom end of an ILV and its landing pad. A resistive open typically occurs due to bonding defects [6], hairline cracks, and pinhole defects [22].

E. Prior Work on ILV Test and Diagnosis

Due to the high ILV integration density, retrofitting of conventional interconnect BIST approaches can introduce significant overhead. Methods such as [23] and [24] use dedicated scan elements (test points) for test access. However, these solutions require large test application time since the number of test patterns required for high fault coverage can become prohibitively large for high ILV density [25]. Moreover, the number of required test points is directly proportional to the ILV count. ATPG-based interconnect test methods, such as [26], are likely to be less effective for ILV testing because I/O pins are available only on one layer in an M3D IC; either test data or test responses—or both in the case of ILVs that do not land on the bottom tier—must be propagated through multiple tiers and the associated ILVs. This requirement adds significantly to the propagation constraints for ATPG. Even if tests can be found by an ATPG tool, additional ILV faults on test paths, which is a likely scenario due to high ILV density, will impede testability. Commercial ATPG tools tend to target single faults for a test-pattern generation. However, multiple faults are likely for dense ILV layouts; hence, test escapes might occur if tests are generated under the single-fault assumption. The proposed BIST alleviates these problems by using a compact set of test patterns that exhaustively test for single or multiple ILV fault scenarios with test-output compaction and negligible fault-masking probability.

While postbond TSV testing techniques can be extended to postassembly M3D testing, recently proposed methods, such as [27], need a die-wrapper register cell on both ends of the ILV for controllability and observability. The drawbacks of applying the IEEE 1838 3-D test standard to M3D ICs, which contain many more vertical connections than 3-D-stacked ICs, include: 1) the current test standard does not provide on-chip ILV-fault localization capabilities; 2) dedicated die-wrapper registers on every ILV will have high area overhead; and 3) inland wrapping (to avoid adding dedicated wrapper cells) will test both ILV and tier logic as part of "EXTEST," leading to a loss in diagnostic accuracy. High diagnostic accuracy is imperative for yield ramp-up for new processes, such as ILV fabrication. In [28], a design flow is proposed to test the full M3D stack, including ILVs, but without fault localization. Postbond TSV testing methods proposed in [29] and [30] use response compaction for the detection of resistive defects; however, on-chip fault localization is not supported.

In [5] and [31], an ILV BIST solution is presented using interface scan cells and a twisted ring counter. However, it mandates a dedicated test layer, which adds to the number of fabrication steps and area overhead. This technique also assumes that the number of upward-facing ("up") ILVs is equal to the number of downward-facing ("down") ILVs between the two tiers. However, in real designs, this assumption is unlikely to hold, and dummy ILVs are added to equalize the ILV counts. The BIST architecture in [10] cannot localize shorts between up-going and down-going ILVs, and assumes a 1-D ILV arrangement for testing. Such an assumption severely restricts the applicability of the BIST to realistic ILV layouts where ILVs are placed in a 2-D array or an irregular arrangement after the PPA-optimized place-and-route of the M3D design.

III. Shared-BIST Architecture for Fault Detection and Localization

A. Shared-BIST Architecture

We present a BIST framework to detect and localize faults in irregularly placed ILVs in dense M3D layouts. Fig. 2 illustrates the BIST solution for on-chip fault detection and localization. This architecture has four main hardware and software components: 1) on-chip BIST Launch engine; 2) on-chip BIST Capture engine; 3) on-chip switch-box layer; and 4) off-chip BIST-optimization engine, which is implemented as a software framework.

The BIST Launch engine is responsible for generating deterministic test patterns to test for SAFs, opens, shorts, and resistive defects (delay faults) in the ILVs. A reconfigurable delay bank is built inside the BIST Launch engine for tuning

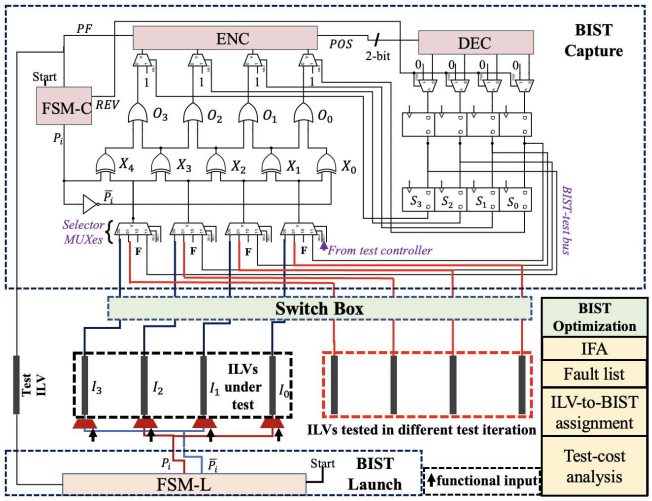

Fig. 2. Block diagram of the ILV-BIST framework.

the delay of test paths through ILVs in the BIST mode for the detection of small-delay defects (SDDs). Wear-out (aging-induced) faults typically manifest as SDDs and, thus, can be detected in the field using the proposed ILV-BIST. Even latent (early life) defects that otherwise do not pose a threat to design functionality can be detected by BIST through overtesting. This is achieved by adjusting the delay stages inside the delay bank such that the test-path slack through an ILV is smaller than that of the longest functional path through the ILV.

The BIST Capture engine uses response compaction to generate the pass/fail result for the ILV test along with the potential location(s) of the failing ILV(s). The switch-box layer facilitates the test and localization of shorts (both hard and resistive) between up-going and down-going ILVs; details are discussed in Section III-C. In large designs with irregular ILV placement, the number of potential ILV-to-ILV short candidates becomes prohibitively large leading to large BIST area and test-time overheads, thereby rendering BIST-insertion infeasible. We have developed an optimization algorithm that uses IFA and layout information to prepare a fault list containing selected ILV pairs that are susceptible to shorts, together with all ILV opens and SAFs (see Section IV for details). Based on the fault list, the BIST-optimization framework generates the BIST configuration for insertion in the gate-level M3D netlist; details are provided in Section V. The BIST configuration consists of the number of BIST engines to be inserted and an optimal assignment of ILVs to the BIST pins to minimize test cost comprising area and timing overheads.

The BIST engine tests up to n ILVs concurrently where all ILVs propagate signals in the same direction. Here, n is referred to as the *width* of the BIST engine and is typically a power of 2, i.e., $n = 2^k$, where k is a nonnegative integer. In other words, n is the number of input pins of the BIST Capture engine. Without loss of generality, we consider the BIST Launch and BIST Capture to be placed in the bottom and top tiers, respectively. We use a sequence TS of three 1-bit test patterns, $TS = \{P_0 = 1, P_1 = 0, P_2 = 1\}$, to test for transition faults, SAFs, and shorts in the ILVs-under-test, i.e., the ILV connected to the capture engine. The pattern P_0 is followed by P_1, which, in turn, is followed by P_2.

Each 1-bit pattern P_i enforces the launch of alternating 0's and 1's into the capture engine's inputs for detecting shorts between ILVs assigned to the adjacent input pins. This is achieved by broadcasting P_i and the inversion of P_i, i.e., $\bar{P}_i$, to the odd-numbered and even-numbered pins of the BIST Capture, respectively. The BIST Launch engine leverages a finite-state machine (FSM) called FSM-L for generating the test sequence TS. The FSM-L macro produces two outputs, P_i and $\bar{P}_i$, in any given clock cycle. The test pattern is then fed to the test-mode input of a 2:1 multiplexer (MUX) whose output is connected to an ILV-under-test. The select line of this MUX enables the switch from functional to BIST mode of operation.

The BIST Capture engine is comprised of six key components: 1) an FSM-C; 2) an XOR-OR network; 3) a priority encoder (ENC); 4) a priority decoder (DEC); 5) a first-in–first-out (FIFO) macro of depth 2; and 6) selector MUXes. Due to the large ILV count in M3D designs, having dedicated scan flops to capture the ILV responses in BIST mode will lead to a large area overhead. A dedicated scan chain for the ILV test can help in accurate fault localization by backtracing the failing flop to the failing ILV. However, it takes several clock cycles to shift out the scan response in order to determine the failing flop(s). As a result, response compaction is necessary to reduce the area overhead of the added BIST circuitry and the test time for M3D designs with many ILVs. Our XOR-OR response compactor generates a single bit corresponding to the pass/fail information for ILVs-under-test. The on-chip ENC–DEC macro uses only a few bits to provide information on the fault location that can be used as feedback for in-field self-repair. Thus, both fault detection and localization benefit from on-chip response compaction and enable low-cost characterization of ILV faults.

To minimize area overhead, the Shared-BIST architecture allows multiple ILVs to share one BIST engine via time-multiplexing. The BIST Capture has n *selector* MUXes, corresponding to the n input pins of the capture engine, which are shared by multiple ILVs across different test iterations. The width w of a selector MUX is determined by the total number of test iterations, t, required to test for all possible ILV faults: $w = 2^{\lceil \log_2 t \rceil}$. For example, two test iterations ($t = 2$) imply that two ILVs are assigned to a particular pin of the BIST Capture across two different iterations because all ILV faults cannot be tested in a single iteration by the n input pins. In Fig. 2, the intention is to test for shorts between the ILV pairs: (I_3, I_2), (I_2, I_1), and (I_1, I_0). Now, to test for an additional short between ILVs I_1 and I_3, they must be assigned to adjacent pins of the BIST-Capture engine, and hence, an additional test iteration would be required. Fig. 2 illustrates a second test iteration (shown with a red dotted rectangle) where two of the four ILVs being tested can be I_1 and I_3. By using the selector MUXes, we can switch between the two test iterations—shorts (I_3, I_2), (I_2, I_1), and (I_1, I_0) are tested in the first iteration followed by short (I_1, I_3) in the second iteration. No potential ILV-to-ILV short goes undetected in our proposed methodology, and 100% coverage

Truth Table for ENC

O_3	O_2	O_1	O_0	$POS[1]$	$POS[0]$	PF
1	1	1	1	X	X	0
1	1	1	0	0	0	1
1	1	0	X	0	1	1
1	0	X	X	1	0	1
0	X	X	X	1	1	1

Truth Table for DEC

$POS[1]$	$POS[0]$	PF	S_3	S_2	S_1	S_0
X	X	0	0	0	0	0
0	0	1	1	1	1	1
0	1	1	1	1	1	0
1	0	1	1	1	0	0
1	1	1	1	0	0	0

Fig. 3. Truth tables of ENC and DEC blocks.

of all potential shorts is achieved through careful planning and assignment of the ILVs to the BIST pins across different test iterations.

The outputs of the selector MUXes feed an XOR-OR network. The outputs of adjacent selector MUXes are tied to the inputs of a two-input XOR gate. For n MUXes, there are $n + 1$ XOR gates. The floating pins of the leftmost and rightmost XOR gates are connected to the outputs of an FSM block called FSM-C. The FSM-C macro generates the same test sequence S synchronously with FSM-L. The $n + 1$ XOR gate outputs feed n two-input OR gates where the outputs of adjacent XOR gates are tied to the inputs of an OR gate. The outputs of $n = 2^k$ OR gates feed a priority ENC, referred to as ENC, via 2:1 MUXes; the ENC macro has 2^k inputs and generates $k + 1$ outputs.

The least significant bit (LSB) of the ENC output bus, PF, indicates if the input bus of ENC contains at least one "0"-carrying bit. If the input bus contains a "0," PF $= 1$. The PF bit indicates the pass/fail status of the ILVs under test; PF $= 1$ indicates that at least one fault is present in the ILVs and PF $= 0$ otherwise. The remaining k output bits, collectively called POS, indicate the position (in binary format) of a "0" bit in the 2^k-bit input bus, considering the output of the leftmost OR gate as the most significant bit (MSB). Fig. 3 shows the Boolean functionality of ENC in the form of a truth table. For example, for $k = 3$, the input bus to the ENC has eight bits. If the LSB of the input bus is "0," the ENC produces "001" as the output: PF is "1" and POS is "00."

The POS output bus feeds a priority decoder called DEC. The DEC macro has k inputs and 2^k outputs. Fig. 3 presents the truth table for DEC. The outputs are tied to a 2^k-bit wide FIFO macro of depth 2. The outputs of DEC are connected to the FIFO's first stage via 2^k 2:1 MUXes. The select lines of these MUXes are driven by a signal REV generated by FSM-C. The 2^k-bit output bus of FIFO's second stage feeds the select inputs of the 2^k 2:1 MUXes at the input of ENC. A "0" select-input passes the OR outputs into the ENC, and a "1" select-input bypasses the OR outputs to send "1" to the ENC. The PF output of ENC acts as a feedback to FSM-L (connecting via a test ILV) and FSM-C to determine the next test pattern to be generated.

The selector MUX switches between test iterations (test mode), the BIST-test mode, and the functional mode. In the functional mode, the MUX output is tied to a constant binary value F to prevent unnecessary switching in the BIST Capture logic. In test (functional) mode, BIST pattern generation is activated (frozen) by asserting (deasserting) the "Start" inputs to FSM-L and FSM-C. The select lines of the selector MUXes, together with the "Start" signal, are driven by an on-chip test controller.

B. Detection and Localization of Single and Multiple Faults

The Shared-BIST tests for all possible shorts and transition faults in the ILVs across several test iterations. The total number of test iterations depends on the ILV count, the number of potential ILV-to-ILV short locations, the number of Shared-BIST engines, and the engine width n. In a given test iteration, the BIST Launch engine applies the test sequence TS to the ILVs under test that is assigned to the corresponding BIST Capture engine. A test iteration concludes when the entire test sequence of three patterns has been applied, and all faults are targeted in the ILVs under test.

1) Fault-Free ILVs: During a test iteration, the ILVs receive alternating 0's and 1's under every pattern $P_i (i \in \{0, 1, 2\})$. If the ILVs are fault-free, they propagate the alternating 0's and 1's to the XOR inputs. Consequently, every XOR gate outputs "1." The XOR outputs feed the OR gates, which, in turn, output "1." As a result, the ENC logic returns PF $= 0$, indicating that the ILVs are fault-free.

2) Single and Multiple Faults in ILVs: The Shared-BIST detects a short between two ILVs assigned to adjacent BIST Capture pins. Consider n ILVs connected to the capture engine in a given test iteration: $I_{n-1}, I_{n-2}, \ldots, I_0$. Here, $I_{n-1}(I_0)$ denotes the ILV assigned to the leftmost (rightmost) pin. Let X_i denote the XOR gate connected between I_i and I_{i-1}. If a short is present between ILVs I_i and I_{i-1}, and I_{i-1} drives I_i, the outputs of adjacent XOR gates X_{i+1} and X_i become "0." Alternatively, if I_i drives I_{i-1}, the outputs of adjacent XOR gates X_i and X_{i-1} become "0." As adjacent XOR outputs feed an OR gate, the corresponding OR gate's output becomes "0." In essence, the position of the "0"-carrying bit in the OR layer's output bus is indicative of the short location and can be traced back to a set of at most three candidate ILVs containing the short. The output bus Y of the XOR-OR network feeds the ENC macro that returns PF $= 1$ along with the position of the "0"-bit in Y. For example, in Fig. 2, if the output of the OR gate O_2 is "0," both its inputs must be "0." Therefore, both XOR gates X_3 and X_2 must be receiving identical inputs. This implies that at least one short is present among the ILVs I_1, I_2, and I_3.

The pattern transition $P_0 \rightarrow P_1$ ($P_1 \rightarrow P_2$) tests for a faulty $1 \rightarrow 0$ ($0 \rightarrow 1$) transition on an ILV occurring due to a delay defect, such as an open, process variations, or a stuck-at-1(0) fault. Similar to the prior analysis for shorts, it can be shown that a single transition fault in the ILVs under test can be detected and localized to a candidate set of at most three ILVs upon application of the pattern transitions. If multiple ILVs are faulty, the ENC-DEC logic pair forces the FSM to continue generating the same pattern transition until all faults activated by that particular transition are detected. A detected fault on an ILV is bypassed using the DEC during subsequent application of the same pattern transition, thereby allowing a faulty ILV in a less significant bit position (with respect to ENC's input pins) to be localized by the ENC.

Example: In Fig. 2, consider the case when the leftmost ILV (I_3) contains a stuck-at-1 fault and the rightmost ILV (I_0)

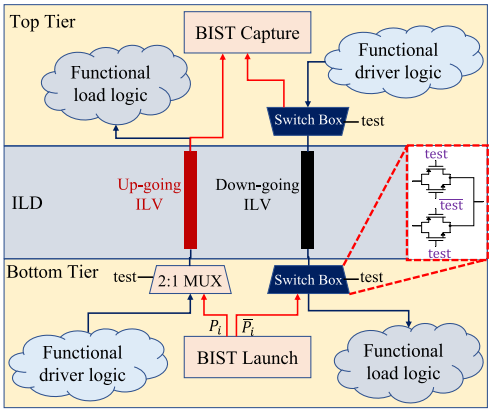

Fig. 4. Transmission gate-based switches for enforcing unidirectional current flow in ILVs under test.

contains a stuck-at-0 fault. When pattern $P_i = 1$ is applied, I_3 receives the pattern-complement 0 but propagates 1 due to the fault. On the right of I_3, the fault-free ILV I_2 receives and propagates $P_i = 1$. Consequently, the corresponding XOR gates (X_4 and X_3) produce 0's, which, in turn, leads to the corresponding OR gate (O_3) producing a 0. In the same pattern cycle, the OR gate O_0 also produces 0 due to the fault in ILV I_0. Thus, the input to the ENC is $[O_3, O_2, O_1, O_0] = [0, 1, 1, 0]$. Based on ENC's truth table, input vector 0110 produces an output POS = 11 and PF = 1. These outputs indicate that a fault is present in at least one of the four ILVs, and the leftmost ILV (I_3; denoted by binary-to-decimal conversion of POS = 11) contains the fault. At the same time, the DEC uses POS to mask the output of the OR gate O_3 (and causes it to send 1 to the ENC) so that we can identify other failing ILVs with the same pattern P_i. Next, we apply the same pattern P_i with the output of O_3 masked by DEC. Now, the input vector to the ENC is "1110" that results in POS = 00 and PF = 1. Thus, we uncover the identity of the rightmost failing ILV (I_0) using ENC by continuing to apply the same pattern P_i while masking the previously detected failing ILVs' outputs using DEC. The test program responsible for controlling pattern generation based on the ENC output is implemented inside both FSM-C and FSM-L. Therefore, in Fig. 2, we have a feedback path carrying PF from the ENC to the FSM-C and FSM-L macros.

C. Localizing Shorts Between Up-Going and Down-Going ILVs

Fig. 4 illustrates the transmission gate-based switch boxes that ensure that the test pattern, launched by BIST Launch, is propagated in the same direction by both up-going and down-going ILVs in the test mode. This enables Shared-BIST to localize shorts between up-going and down-going ILVs by assigning them to adjacent pins of the BIST Capture engine. Tristate buffers can also be used instead of transmission gates.

Without loss of generality, if the BIST Capture is in the top tier, the current flow through down-going ILVs is reversed using the MUX-DEMUX pairs in the BIST mode. However, the physical structure and dimensions of the ILVs remain unaffected by this reversal of signal flow. After the reversal of the logical connection, a single MUX gate in BIST-Launch becomes the only driver gate for the down-going ILV, and a single DEMUX gate in the switch box of BIST-Capture becomes the only load. Due to the presence of a single load gate, the driver gate does not need to draw an exceedingly high current from the power supply for sending through the ILV for charging/discharging the load capacitance. Thus, for the same ILV cross-sectional area, the current density remains unaffected, and the impact of electromigration on the ILV metal due to the current-flow reversal is negligible.

IV. Defect-Graph Generation for ILV Faults

A. Identifying Hotspots for ILV Shorts

The top view of an ILV in the M3D layout is similar to that of a conventional BEOL via. It is typically rectangular in shape with the ILV's location coordinates specified in the design exchange format (DEF) file of the layout [32]. The DEF file is generated after the automatic place-and-route of the M3D design is completed; see Section VI-A for an overview of the M3D physical design flow adopted in this work. The ILV location is typically defined as the coordinates (x_u, y_u) of the centroid of the rectangle that represents the ILV u in the layout's top view.

The ILVs, together with the potential shorts between them, can be represented using a weighted graph, referred to as a *defect graph* (G). In G, the vertices denote the ILVs, and the edges denote potential shorts between the corresponding ILVs. The weight $w_{u,v}$ of an edge (u, v) is the Euclidean distance between ILVs u and v in the layout: $w_{u,v} = ((x_u - x_v)^2 + (y_u - y_v)^2)^{1/2}$. For N ILVs, the maximum number of possible shorts that can occur (i.e., edge count in G) is $(N \cdot (N-1)/2)$. The overhead of the BIST hardware required to test for all those shorts can become prohibitively large for large values of N. However, not all shorts are likely to occur. The likelihood of a short occurrence depends on the physical distance between two ILVs; for example, a short is more likely to occur between two closely placed ILVs than between ILVs that are far apart.

The Euclidean distance between two ILVs' centroids can be used to evaluate the likelihood of a short. For example, if a defect of size exceeding d (in arbitrary distance units) is unlikely to occur during the fabrication process, a short is not likely to occur between two ILVs that are apart by a distance greater than d. Consequently, the defect graph G can be pruned by removing those edges whose weight $w_{u,v} > d$. A probabilistic estimate of the short likelihood between two ILVs in G is obtained via IFA, as discussed in Section IV-B. Such an estimate guides the pruning of G for low-cost BIST insertion. The edges remaining after pruning denote the candidate ILV pairs, or hotspots, where shorts are more likely to occur.

Certain arrangements of ILVs in the layout enable the dropping of selected shorts (edges) in G from testing. This is because the defect that is responsible for causing a short between ILVs u and v is also guaranteed to short at least one other ILV pair in the given ILV placement. As a result, we can drop the short between u and v from testing.

For example, consider three ILVs u, v, and w in a proximal collinear arrangement (i.e., 1-D array). There is a need to test for the shorts (u, v) and (v, w). However, the short (u, w) can be dropped because the defect causing this short is also likely to cause at least one of the other shorts already tested for, i.e., $\{(u, v), (v, w)\}$. In other words, the shorts $\{(u, v), (v, w)\}$ are implied by the short (u, w), and hence, (u, w) can be safely removed from G. Such an implication-based geometric pruning of G, coupled with IFA, is covered in Section IV-B.

B. Geometric Pruning of ILV Defect Graph: Inductive Fault Analysis

IFA is a procedure for identifying faults that are likely to occur [11]. We leverage IFA to determine the candidate ILV pairs for shorts and pruning the defect graph G. To the best of our knowledge, IFA has not been used before ILV testing in M3D integration.

Our IFA procedure takes as input the probability distribution of defect size r, $p^{\text{def}}(r)$, which has been observed for new M3D technology from a foundry.[1] The probability of a short $(P_{\text{sh}}^{(u,v)})$ occurring between two ILVs u and v equals the probability that the size of the defect causing the short (u, v) exceeds $w_{u,v}$. Therefore, the likelihood of occurrence of the short (u, v) is given by $P_{\text{sh}}^{(u,v)} = \int_{r=w_{u,v}}^{\infty} p^{\text{def}}(r)dr$. We compute the short likelihood $P_{\text{sh}}^{(u,v)}$ for all $(N \cdot (N-1)/2)$ edges in G. The edges (or shorts) with likelihood exceeding a user-defined threshold are retained for testing; others are pruned.

1) Probabilistic Geometric Pruning: For topology-based and IFA-driven pruning, we consider the distance between ILVs to determine the likelihood that a defect can cause a short; shorts between far-off ILVs are unlikely and can, therefore, be neglected (and the corresponding edges pruned from the defect graph). However, we do not need to necessarily test all of the remaining shorts. This is because a defect is likely to cause multiple shorts in a cluster of closely placed ILVs; in such cases, testing one of these shorts ensures that other proximal shorts, if present, will also be detected. Therefore, based on the relative position of the ILVs in the layout and the geometry of the possible defects, we can further prune the defect graph. Note that, while defects can be randomly shaped, yield prediction and IFA approaches usually assume that lithographic defects have the shape of circular disks [33] or squares [34]. In addition, methods to convert randomly shaped defects to an equivalent circular defect that can lead to a similar probability of fault occurrence have been proposed in prior work and verified using experimental observations [35]. Therefore, we use the circular-shaped defect model to calculate the probability that a defect can remain untested if an edge is pruned from the defect graph.

2) Pruning Based on Circular Defect Model: Consider the top view of a layout with three ILVs, namely, A, B, and C [see Fig. 5(a)]. The lengths of the sides $\overline{BC}$, $\overline{CA}$, and $\overline{AB}$ of the ΔABC are a, b, and c, respectively. Without loss of generality, suppose that $c \geq a$ and $c \geq b$. The following

[1]Foundry and process details are being withheld due to confidentiality reasons.

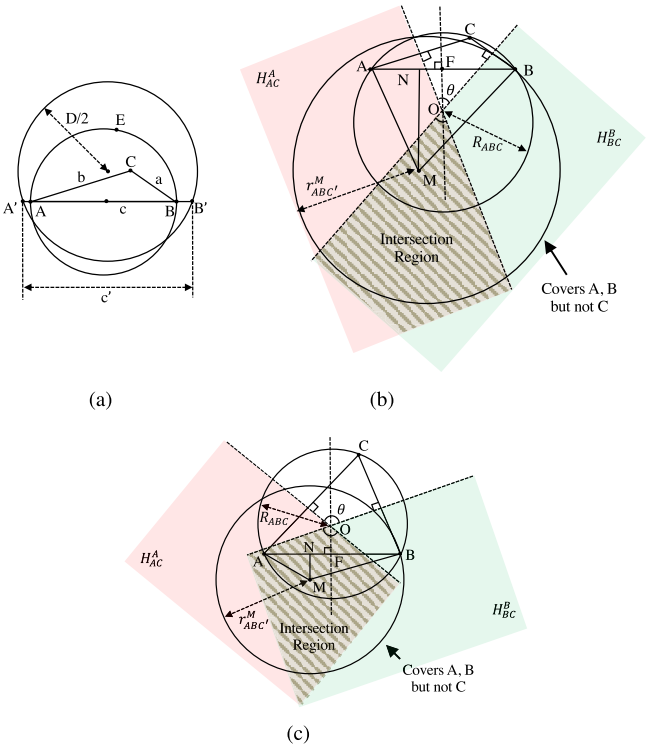

Fig. 5. (a) Smallest circular defect that shorts ILVs A and B has diameter equal to $\overline{AB}$ and is centered at its midpoint. (b) For a triangle ABC, where the circumcenter O lies outside the triangle, any circular defect that covers A and B but not C, must have a radius greater than the circumradius of ΔABC and must be centered in the shaded intersection region. (c) Conversely, if the circumcenter O lies inside ABC, any circular defect that covers A and B but not C, must have a radius greater than the $\overline{AB}/2$ and must be centered in the shaded intersection region. (b) Circumcenter O outside ABC. (c) Circumcenter O inside ABC.

theorem provides a geometric characterization of the smallest circular defect that can short two ILVs.

Theorem 1: The smallest circular defect that can short two ILVs is centered at the midpoint of the line segment connecting the ILVs and has a diameter equal to the inter-ILV distance.

Proof: We prove this theorem using contradiction. Considering Fig. 5(a), suppose that there exists a circular defect with diameter $D < c$ that shorts the ILVs A and B. We extend the line segment $\overline{AB}$ on both sides to A' and B' such that $\overline{A'B'}$ is a chord of the defect circle. Suppose that the length of $\overline{A'B'}$ is c' with $c' \geq c$. Given that the diameter is the largest chord in a circle, $D \geq c'$. However, by definition, $D < c \leq c'$. This leads to a contradiction, and therefore, the diameter of a defect that shorts A and C is greater than or equal to c. Now, consider a circular defect with diameter c centered at the midpoint of $\overline{AB}$. It is clear that this defect shorts A and B, and is the only possible circular defect of diameter c that shorts A and B. This is because, for any other defect with diameter c, at least one of A and B will remain outside of the defect circle. This completes the proof. □

The circle ABE in Fig. 5(a) represents the smallest circular defect that shorts ILVs A and B and also shorts ILV C with A and B. While this holds for this particular defect size, it is possible that other (larger) defects can short A and B

without affecting C [see Fig. 5(b)]. Such defects can, therefore, remain untested if the edge $\overline{AB}$ is pruned from the defect graph. In triangle ABC, consider the perpendicular bisectors of $\overline{AB}$, $\overline{BC}$, and $\overline{CA}$ intersecting at the circumcenter O. By definition, O is equidistant from A, B, and C; this distance is given by the circumradius $R_{ABC} = abc/(4 \cdot S_{ABC})$. Here, S_{ABC} denotes the area of triangle ABC. Let $p^{\text{circ}}(r)$ denote the probability of occurrence of a circular defect of radius r. The following theorem, in particular, (1), provides an upper bound on the probability of defect escape if $\overline{AB}$ is pruned from G.

Theorem 2: In a defect graph with vertices (ILVs) A, B, and C, if $P^{\text{circ}}_{ABC'}$ denotes the probability that a circular defect covering A and B does not cover C, then

$$P^{\text{circ}}_{ABC'} \leq \frac{\pi - \angle BCA}{2\pi} \int_{R^*}^{\infty} p^{\text{circ}}(r)\, dr \tag{1}$$

where $R^* = R_{ABC}$ if O lies outside ΔABC and $R^* = \overline{AB}/2$ otherwise.

Proof: The perpendicular bisector of any line segment divides a plane into two half-planes; all points in a half-plane are closer to the endpoint of the line segment that lies in the half-plane. Suppose that the half-plane with the point A formed by the perpendicular bisector of $\overline{AC}$ is denoted by H^A_{AC}, and the half-plane with the point B formed by the perpendicular bisector of $\overline{BC}$ is denoted by H^B_{BC}. From Fig. 5(b), observe that H^A_{AC} and H^B_{BC} intersect at O, and all points in the shaded region between the two perpendicular bisectors lie in both the half-planes. Given that all points in H^A_{AC} (H^B_{BC}) are closer to A (B) than to C, all points in the intersection region are closer to both A and B compared to C. Consequently, for all points in the intersection region, there exists a circular defect centered at the point, which can cover A and B (thereby shorting them), without covering C. All such defects can remain undetected if $\overline{AB}$ is pruned.

The converse is also true; for any point, X, outside the intersection region, $(\overline{XC}) \leq \max\{(\overline{XA}), (\overline{XB})\}$. Therefore, if circular defects centered outside the intersection region cover both A and B, it must cover C; all such defects will still be detected if $\overline{AB}$ is pruned. This establishes that all defects that can remain undetected if $\overline{AB}$ is pruned must be centered at the intersection region between H^A_{AC} and H^B_{BC}. Let E_{shaded} be the event that a circular defect assuming that circular defects are uniformly distributed on the plane, the probability that a circular will be centered in the intersection region; its probability is then given by $P(E_{\text{shaded}}) = \theta/2\pi$, where $\theta = \pi - \angle BCA$ is the angle between the perpendicular bisectors, as shown in Fig. 5(b).

Suppose that the perpendicular bisector of $\overline{AB}$ intersects it at F; $\overline{AF} = \overline{FB}$. Consider a point, M in the intersection region between H^A_{AC} and H^B_{BC} with $\overline{MN} \perp \overline{AB}$. The radius of the smallest circular defect centered at M that can short A and B is $r^M_{ABC'} = \max(\overline{AM}, \overline{BM})$. Note that two cases might arise here based on the location of the circumcenter O wrt ΔABC. If O lies outside ΔABC [see Fig. 5(b)], $\overline{MN} \geq \overline{OF}$. Observe also that $\max(\overline{AN}, \overline{NB}) \geq \overline{FB}$, with equality holding when M lies on $\overline{OE}$. Therefore, we have

$$\begin{aligned} r^M_{ABC'} &= \max(\overline{AM}, \overline{BM}) = \sqrt{\overline{MN}^2 + \max(\overline{AN}, \overline{NB})^2} \\ &\geq \sqrt{\overline{OF}^2 + \overline{FB}^2} = R_{ABC} \end{aligned} \tag{2}$$

with equality holding when M coincides with the circumcenter O. On the other hand, if the circumcenter O lies inside ΔABC [see Fig. 5(c)], $\overline{AM} + \overline{BM} \geq \overline{AB}$. Therefore, in this case, $r^M_{ABC'} = \max(\overline{AM}, \overline{BM}) \geq \overline{AB}/2$, with equality holding if M is the midpoint of $\overline{AB}$. In summary, $r^M_{ABC'} \geq R_{ABC}$ if O lies outside ΔABC and $r^M_{ABC'} \geq \overline{AB}/2$ otherwise.

Suppose that $E^M_{ABC'}$ is the event that a circular defect centered at M covers A and B while not covering C. With $p^{\text{circ}}(r)$ being the probability of occurrence of a circular defect of radius r and O lying outside ΔABC, we then have

$$P(E^M_{ABC'}) = \int_{r^M_{ABC'}}^{\infty} p^{\text{circ}}(r)\, dr \leq \int_{R_{ABC}}^{\infty} p^{\text{circ}}(r)\, dr. \tag{3}$$

This is because $p^{\text{circ}}(r)$ decreases monotonically with increasing r and $r^M_{ABC'} \geq R_{ABC}$. Similarly, for the case where O lies inside ΔABC, we have

$$P(E^M_{ABC'}) \leq \int_{\overline{AB}/2}^{\infty} p^{\text{circ}}(r)\, dr. \tag{4}$$

The probability that a random circular defect will cover A and B, but not C, is then given by $P^{\text{circ}}_{ABC'} = P(E_{\text{shaded}} \cap E^M_{ABC'})$, where E_{shaded} is the event that a circular defect is centered in the shaded intersection region and $E^M_{ABC'}$ is the event that such a circular defect covers A, B, but not C. These events are mutually independent; therefore, if O lies outside ΔABC, we have

$$\begin{aligned} P^{\text{circ}}_{ABC'} &= P(E_{\text{shaded}} \cap E^M_{ABC'}) \\ &= P(E_{\text{shaded}}) \cdot P(E^M_{ABC'}) \\ &= \frac{\pi - \angle BCA}{2\pi} \int_{r^M_{ABC'}}^{\infty} p^{\text{circ}}(r)\, dr \\ &\leq \frac{\pi - \angle BCA}{2\pi} \int_{R_{ABC}}^{\infty} p^{\text{circ}}(r)\, dr. \end{aligned} \tag{5}$$

In the case where O lies inside ΔABC, we similarly have

$$P^{\text{circ}}_{ABC'} \leq \frac{\pi - \angle BCA}{2\pi} \int_{\overline{AB}/2}^{\infty} p^{\text{circ}}(r)\, dr. \tag{6}$$

□

During pruning, a maximum acceptable probability of defect escape is set according to the target defect level, and all edges for which the defect-escape probability lies below this threshold can then be safely pruned.

3) Defect Level-Aware Threshold Probability for Pruning: The number of shorts pruned should not lead to significant defect escape. In other words, the defect-escape probability resulting from the pruned edges must not exceed the target defect level (DL) for the chip. Accordingly, geometric pruning is carried out in adherence with a predetermined DL set by the user. In the case of circular defects, Fig. 6 shows the number of shorts or edges that can be pruned for a certain DL and $p^{\text{circ}}(r)$. The defect-size distribution function $p^{\text{circ}}(r)$ is given by: $p^{\text{circ}}(r) = a \cdot e^{-b \cdot r}$. As the size of the

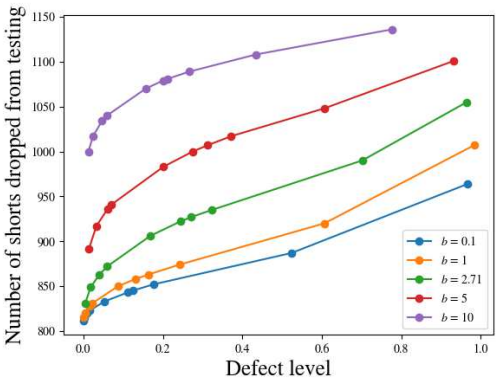

Fig. 6. Defect level-aware pruning of ILV defect graph for various defect-size distributions: $p^{\text{circ}}(r) = (b)/(1 - e^{-\sqrt{2}b}) \cdot e^{-b \cdot r}$.

largest possible circular defect (r_{lim}) is limited by the die area ($W \times L$, where W and L are the width and length of the die footprint), we can safely assume that the probability of the defect size exceeding $r_{\text{lim}} = (W^2 + L^2)^{1/2}$ is nearly zero. Therefore, $p^{\text{circ}}(r) = 0$ for $r > r_{\text{lim}}$. Also, note that the area under $p^{\text{circ}}(r)$ between $r = 0$ and $r = r_{\text{lim}}$ must equal 1. Therefore, $\int_{r=0}^{r_{\text{lim}}} a \cdot e^{-b \cdot r} dr = 1 \implies a = (b/1 - e^{-r_{\text{lim}} \cdot b})$. For demonstrating pruning in Fig. 6, a defect graph G with N ILVs is synthetically generated by randomly sampling N (x_i, y_i) coordinate pairs between 0 and 1, i.e., $0 \leq x_i, y_i \leq 1$. Hence, $r_{\text{lim}} = \sqrt{2}$. The maturity of the M3D technology is indicated by the magnitude of b in the expression for $p^{\text{circ}}(r)$. Larger b implies a more mature fabrication flow where the likelihood of large-sized defects is extremely low. For the same DL, we see that more shorts can be pruned, or dropped from testing, for higher values of b. The defect-size distribution corresponding to $b = 2.71$ is extracted from a foundry's measured data.

V. Optimization of Shared-BIST Architecture

A. Problem Formulation for Assigning ILVs to BIST Pins

To test for a short between two ILVs (i.e., an edge in the defect graph G), the ILVs must be assigned to adjacent pins of the BIST-Capture engine. The odd- and even-numbered pins of the BIST-Capture receive complimentary pattern bits (P_i and $\bar{P}_i$, respectively) from the BIST-Launch. Consequently, a short can be detected between ILVs assigned to adjacent pins of the BIST-Capture. The BIST-Capture engine has a limited number of input pins, which is determined by the overhead budget of the test infrastructure. Therefore, all shorts in a densely connected defect graph may not be covered in a single test iteration. Moreover, multiple test iterations are needed when the number of ILVs present in the M3D design exceeds the number of BIST-Capture pins available for assignment. Consider m BIST-Capture engines with c pins per engine. The *test capacity* (T_C) of the Shared-BIST is given by the maximum number of ILVs that can be tested in a single test iteration: $T_C = m \times c$.

In the case of multiple BIST-Capture engines, the odd-numbered pins in every BIST engine receive the same pattern bit (P_i) in a given test iteration. Similarly, the even-numbered pins in every BIST engine receive the same pattern bit ($\bar{P}_i$) in a given iteration. Note that two ILVs (I_i and I_j) must be assigned to adjacent pins of the same BIST-Capture engine in a given iteration for enabling localization of a short between them. If I_i and I_j are assigned to odd- and even-numbered pins of different BIST-Capture engines, a short can be detected due to the application of complementary pattern bits; however, the short cannot be localized to I_i and I_j. This is because there can be many such ILV pairs receiving complementary patterns in the same iteration, and the detected short can be attributed to any one (or more) of them. We have established earlier (in Section III-B) that the output POS of the priority ENC, carrying information about the fault location, can be traced back to at most three candidate ILVs assigned to adjacent pins of the same BIST-Capture of which the ENC is a part. Given that I_i and I_j are assigned to adjacent pins of the same BIST-Capture engine and the corresponding POS output is traced back to three ILVs containing I_i and I_j, the root cause of the fault can be attributed to open(s) in either or both of I_i and I_j or a short between I_i and I_j. Therefore, the candidate set of faulty ILVs can be significantly pruned when ILVs are assigned to pins of the same BIST-Capture engine.

When ILVs are assigned to the BIST pins for short detection in a given test iteration, they are automatically tested for opens, SAFs, and delay faults by virtue of the three test patterns applied consecutively in the same iteration. As a result, ILVs already assigned for detecting shorts need not be reassigned in another iteration to test for opens separately. First, we assign ILVs to the BIST pins for covering all shorts in G. Then, the remaining ILVs (which are not part of any short) are assigned for detecting opens and delay faults. The required number of test iterations is maximum (worst case scenario) when only one short is tested per test iteration. For testing N ILVs (vertices) with S shorts (edges) in G, the total number of test iterations, t_{max}, required in the worst case scenario is $t_{\text{max}} = \lceil (N_{ns}/T_C) \rceil + S$, where $N_{ns} (\leq N)$ is the number of ILVs not involved as a candidate in any short.

For densely connected defect graphs with many edges (candidate shorts), multiple test iterations may be needed to test for all possible shorts. Similarly, large-sized defect graphs with many vertices (ILVs) may need multiple iterations with a limited number of BIST-Capture engines and limited test capacity. For N ILVs placed in a 1-D array, a single iteration is sufficient to test for all possible faults using a BIST engine with N pins. If the same N ILVs are placed irregularly, more potential shorts may arise, requiring more test iterations (see Fig. 7).

If the number of required test iterations is t, a $t:1$ selector MUX is needed at every input pin of the BIST-Capture to switch between the t iterations. A large value of t increases the area overhead of the Shared-BIST insertion. Moreover, if t increases for large and dense defect graphs, the likelihood of an ILV being assigned in multiple test iterations increases. This implies that the same ILV is connected to multiple input pins of the selector MUX, leading to higher fan-out (FO) and increased capacitive load on the ILV. The increased wire load increases the delay of the path through that ILV and

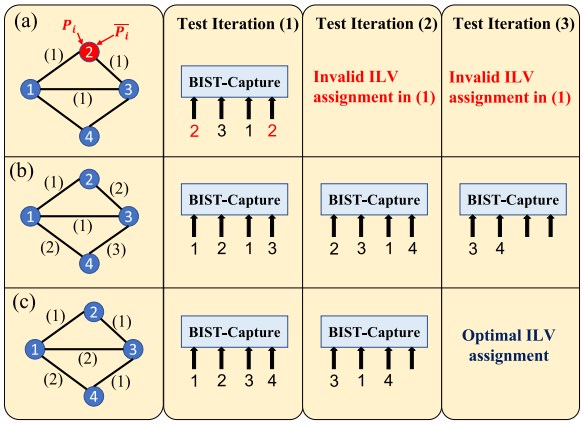

Fig. 7. Examples of ILV-to-BIST assignment. (a) Invalid ILV assignment. (b) Sub-optimal ILV assignment. (c) Optimal ILV assignment.

degrades circuit timing. Therefore, minimizing the number of test iterations t minimizes the area and timing overhead of the inserted BIST hardware and reduces the overall test time for ILV-fault detection and localization.

The sequence in which ILVs are assigned to the BIST pins has a significant impact on t. In Fig. 7, four ILVs are shown in a pruned defect graph. For a single BIST engine with $c = 4$ pins, the ILV assignment [see Fig. 7(a)] leads to an invalid assignment as the same ILV cannot be simultaneously assigned to odd and even-numbered pins. A valid assignment [see Fig. 7(b)] leads to three test iterations, which is one more than the minimum possible for the given defect graph. The assignment [see Fig. 7(c)] illustrates the optimal ILV-to-BIST assignment. These examples highlight the importance of ILV ordering during their assignment to the BIST pins in order to reduce BIST overhead and test time.

A 1-D defect graph of N ILVs with $N-1$ shorts can be viewed as a graph with all $N-1$ edges belonging to a single Hamiltonian path. A *Hamiltonian path* is a graph path that visits every vertex exactly once [36]; note that a defect graph for irregularly placed ILVs does not necessarily contain a Hamiltonian path. A 1-D defect graph G with all of its $c-1$ edges in a Hamiltonian path can be tested in a single iteration with a BIST engine having c pins. If a new edge (short) is added to G, an additional test iteration is needed to cover that edge. Thus, for a given value of c, the test-iteration count is minimum when the edges that are part of the longest simple path in G are tested first. The covered edges (and associated ILVs) are dropped from G, and the remaining edges in the reduced G are assigned in subsequent test iterations following the longest-path approach. Thus, to minimize t, we must determine the longest simple path in G at the beginning of a test iteration and assign the ILVs to the BIST pins in the same order as they appear in the longest path. However, finding the longest path in an undirected graph is NP-Complete [37]. We next develop an integer linear programming (ILP) model for optimal ILV assignment to BIST pins.

B. ILP Model for Optimal ILV Assignment

We present below an ILP model for minimizing the test-iteration count t for a fixed BIST-engine count m, where the number of pins per engine is $c = 2^b$ (b is a positive integer). Suppose that N ILVs need to be tested, where the ILVs are numbered 1 through N. Let the pins of the BIST-Capture engines be numbered from 1 to $T_C = m \cdot c$. We define two binary decision variables, $x_{p,i,j} \in \{0,1\}$ and $y_j \in \{0,1\}$, for the ILP model. The decision variable $x_{p,i,j} = 1$ if ILV i ($1 \le i \le N$) is assigned to pin p ($1 \le p \le T_C$) in test iteration j ($1 \le j \le t_{\max}$); otherwise, $x_{p,i,j} = 0$. The set of decision variables $x_{p,i,j}$ is denoted by X_R. The decision variable $y_j = 1$ if at least one ILV is assigned to one of the BIST pins in test iteration j; otherwise, $y_j = 0$. The ILP model for minimizing t is given by

$$\min_{X_R} \; t = \sum_{j=1}^{t_{\max}} y_j \tag{7a}$$

$$\text{s.t.} \sum_{j=1}^{t_{\max}} \sum_{p=1}^{T_C} x_{p,i,j} \ge 1 \quad \forall 1 \le i \le N \tag{7b}$$

$$\sum_{j=1}^{t_{\max}} \sum_{k=0}^{m-1} \sum_{p=k\cdot c}^{k\cdot c+c-1} x_{p,u,j} \cdot x_{p+1,v,j} + x_{p,v,j} \cdot x_{p+1,u,j} \ge 1 \quad \forall (u,v) \in G \tag{7c}$$

$$\left(\sum_{p\in E} x_{p,i,j}\right) \cdot \left(\sum_{p\in O} x_{p,i,j}\right) = 0 \quad \forall 1 \le i \le N, \; 1 \le j \le t_{\max} \tag{7d}$$

$$\sum_{i=1}^{N} \sum_{p=1}^{T_C} x_{p,i,j} \le T_C \cdot y_j \quad \forall 1 \le j \le t_{\max} \tag{7e}$$

$$\sum_{i=1}^{N} x_{p_i,j} \le 1 \quad \forall 1 \le p \le T_C, \; 1 \le j \le t_{\max}. \tag{7f}$$

The minimization of the objective function (22a) ensures the minimization of the test-iteration count t. Constraint (22b) guarantees that all ILVs are assigned to the BIST pins across t test iterations to cover all possible faults. Constraint (22c) ensures that every short or edge (u,v) ($1 \le u \le N, 1 \le v \le N, u \ne v$) in G is tested at least once across all iterations by assigning ILVs u and v to adjacent pins of the same BIST-Capture engine in the same test iteration. Constraint (22d) enforces that the same ILV cannot be assigned to both odd-numbered (set E of pins) and even-numbered pins (set O of pins) in the same test iteration; such an assignment is deemed invalid as the same ILV cannot be simultaneously driven by P_i and $\bar{P}_i$. Constraint (22e) ensures that the number of ILVs assigned to the BIST pins in a given iteration does not exceed the test capacity. Finally, constraint (22f) allows at most one ILV to be assigned to the same pin in a given iteration; multiple drivers for the same pin or net lead to a high-Z condition and disable fault localization.

The number of variables in the above ILP model is $t_{\max} \cdot (N \cdot T_C + 1) = \mathcal{O}(N^3)$. The total number of linear and nonlinear constraints enforced in the model is $2N + S + 3t_{\max} + T_C = \mathcal{O}(N) + \mathcal{O}(N^2) + \mathcal{O}(N) + T_C = \mathcal{O}(N^2)$. The cubic complexity of the variable count can make the runtime of the ILP model prohibitively large for defect graphs (ILV layouts) with many ILVs, i.e., a large value of N. Therefore, we design a heuristic

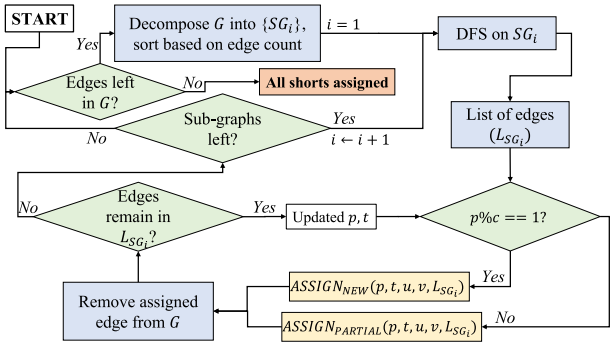

Fig. 8. Greedy heuristic for ILV-to-BIST assignment.

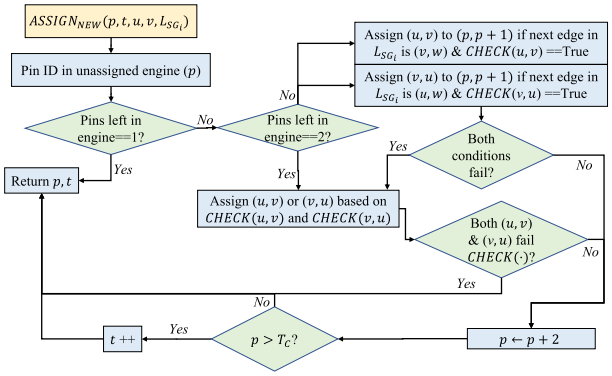

Fig. 9. Flowchart for assigning an ILV pair (u, v) to the pins of an unassigned BIST engine in a given test iteration.

algorithm for greedy ILV-to-BIST assignment and compare its performance with that of the ILP model on medium-sized defect graphs for which the model does not time out. Note that the ILP model serves an important purpose: it can be used to assess the quality of heuristic solutions for medium-sized problem instances.

C. Greedy Procedure for ILV Assignment

Suppose that m BIST engines, with c pins per engine, are available. We design a greedy algorithm to assign ILVs to the BIST pins such that the total number of pins used is minimized. Minimizing the number of pins (p) implies minimization of test-iteration count t, where $t = \lceil (p/T_C) \rceil$. Fig. 8 shows the flowchart for greedily assigning the edges or shorts in the defect graph G to the BIST-Capture pins. Figs. 9 and 10 present the procedures for assigning a given edge or ILV pair (u, v) to adjacent pins of the BIST-Capture. The procedure $CHECK(u)$ validates the assignment of an ILV u to a pin with pin ID p based on the parity of p. The ILV u cannot be assigned to both odd and even-numbered pins in the same test iteration as the same ILV cannot be simultaneously driven by the patterns P_i and $\bar{P}_i$. Therefore, if u is already assigned to an odd (even) numbered pin, $CHECK(u)$ returns **False** when the algorithm attempts to assign u to an even (odd) numbered pin in the same iteration. Similarly, $CHECK(u, v)$ legalizes the assignment of an ILV pair (u, v) to adjacent pins of the BIST-Capture for a given test iteration. For N ILVs, the worst case computational complexity of the greedy heuristic is $\mathbf{O}(N^2)$.

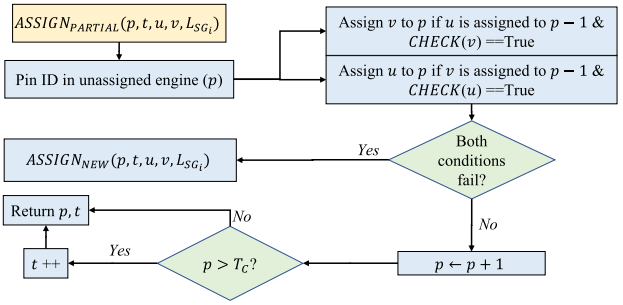

Fig. 10. Flowchart for assigning an ILV pair (u, v) to the pins of a partially assigned BIST engine in a given test iteration.

D. BIST Engine Count for Overhead Minimization

Additional FO branches of an ILV are created if the same ILV is connected to different pins of the BIST-Capture engine in the same test iteration or to different pins of the selector MUX for getting tested in different iterations. A higher ILV FO presents a proportionately larger capacitive load for the functional path through the ILV, resulting in an increased path delay. If an ILV u is a part of n_{sh} shorts in the defect graph G, the minimum number of FO branches needed to test the ILV for associated shorts is $\lceil (n_{\text{sh}}/2) \rceil$. This is because for any two shorts involving u—(u, v_1) and (u, v_2)—the ILV-to-BIST assignment can be done in a way such that both shorts are tested with a single connection between u and the BIST pin; such an assignment is $\{p-1 : v_1, p : u, p+1 : v_2\}$. To test for a third short (u, v_3), a second FO branch must be added to u for connecting it to a second BIST pin. A lower test-iteration count implies a smaller width of the selector MUXes, which implies that the same ILV is likely to be connected to fewer pins of a selector MUX. As a result, minimizing test-iteration count t with an appropriate choice of the BIST-engine count m also reduces the impact of BIST on the ILV-path timing.

As BIST is inserted in the gate-level netlist, it is not possible to compute the wirelength of the newly added nets as part of the ILV-to-BIST assignment. This is because the placement of the BIST engines relative to the ILV locations (and conditional upon the surrounding logic congestion and available space) will determine the wirelength of the resultant routing, which, in turn, will have a certain timing overhead due to the added capacitive load on the functional paths through the ILVs. Including FO information in the ILV-assignment algorithm will degrade the optimality of the assignment leading to a large value of t, and associated area and test-time overheads. Therefore, we consider FO information during a weighted test-cost analysis (see Section VI-B) prior to the selection of area-optimal BIST configuration and BIST insertion.

VI. Experimental Results

A. Evaluation of Proposed Heuristic for ILV Assignment

Table I compares the test-iteration counts and runtimes for ILV assignment obtained using the ILP model and the proposed greedy heuristic. The comparison is demonstrated on a synthetic defect graph G containing seven nodes (ILVs) and randomly sampled edges (candidate shorts to test). The number of edges is determined by the probability of a short (P_{sh})

TABLE I
EVALUATION OF ILV-TO-BIST PIN-ASSIGNMENT METHODS

N	m	P_{sh}	Test iterations			Runtime (s)		
			ILP	First-Fit	Greedy	ILP	First-Fit	Greedy
5	2	0.1	1	1	1	0.1	0.0002	0.0003
	2	0.3	1	2	1	0.26	0.0001	0.0006
	2	0.5	2	2	2	2.14	0.0002	0.0008
	2	0.7	2	4	3	37.5	0.0006	0.0014
7	1	0.1	1	2	1	0.04	0.0001	0.0004
	1	0.3	2	2	2	2.76	0.0001	0.0008
	1	0.5	2	3	2	13.3	0.0008	0.002
10	2	0.1	2	2	2	42	0.0002	0.001
	1	0.3	2	3	2	0.6	0.0002	0.001

TABLE II
COMPARISON OF FFA AND PROPOSED GREEDY HEURISTIC ON MEDIUM-TO-LARGE DEFECT GRAPHS

N	m	c	P_{sh}	Test iterations		Runtime (s)	
				First-Fit	Greedy	First-Fit	Greedy
25	2	8	0.3	13	10	0.006	0.008
50	3	8	0.4	45	31	0.14	0.04
75	4	8	0.6	104	75	1.37	0.24
100	5	16	0.7	99	59	6.65	0.77
200	4	16	0.8	508	319	132.96	9.2
500	5	16	0.1	317	199	80.9	4.83
750	6	16	0.2	1161	795	401.13	22.6

occurring between any two nodes in G. A short occurs between two ILVs in G with the probability P_{sh}. In other words, an edge connects a pair of nodes in G if a random number p_{sh} is uniformly sampled from the range (0, 1) and $p_{\text{sh}} \leq P_{\text{sh}}$. The greedy heuristic returns near-optimal solutions for the test-iteration count, with the highest deviation from the (ILP-determined) optimal solution being only one iteration. In addition, the heuristic algorithm provides orders-of-magnitude speedup compared to the ILP model. Tables I and II also compare the greedy algorithm with that of the first-fit algorithm (FFA) used for online bin-packing [38]. The objective of FFA is to assign items (i.e., ILVs) to bins (i.e., BIST pins in a given test iteration) by using minimum possible bins (i.e., test iterations). FFA picks an edge (ILV pair) from the list of edges arranged in an arbitrary sequence and assigns the edge to the first available set of adjacent pins in the BIST-Capture engines for which the assignment is legalized via a call to the $CHECK(\cdot)$ procedure. The greedy heuristic outperforms FFA in both performance and runtime for a wide range of defect-graph sizes.

In Tables I and II, the reported runtime in seconds indicates the CPU runtime required by the ILV-to-BIST assignment algorithm (FFA, greedy, or ILP) to generate the ILV-to-BIST pin assignment. This is a one-time off-chip software-based runtime overhead that is required to determine the optimal ILV-to-BIST pin assignment prior to BIST insertion, circuit resynthesis, and layout design.

B. Evaluation of BIST Overhead

The two-tier M3D benchmarks used for evaluating BIST overhead are AES-128 (f_m = 483.09 MHz), Nova (f_m = 144.1 MHz), Rocketcore (I) (f_m = 130.7 MHz), and Rocketcore (II) (f_m = 110.7 MHz), containing 269, 322, 1073, and 1062 ILVs, respectively; f_m is the max. operating frequency. The ILV defect graphs are extracted from the DEF files and pruned to yield defect levels of 10^{-14}, 10^{-7}, and 10^{-4} for AES-128, Nova, and Rocketcore (I/II), respectively. Applying the greedy heuristic algorithm on the pruned graphs, we determine the optimum test-iteration count t for a given BIST-engine count m and for c = 16. From t, we estimate the area overhead of the inserted BIST by accounting for the selector MUXes and switch boxes. The additional FO load and timing impact due to BIST insertion are determined by the assignment of ILVs to the BIST pins; the FO load is calculated as the total number of FO branches connecting the ILVs to the BIST pins across all test iterations. DSE is then carried out to obtain m that minimizes the *test cost*. The test cost is evaluated as a weighted sum of normalized area overhead, normalized FO load, and normalized test-iteration count; the allotted weights are w_a, w_{fo}, and $(1-w_a-w_{fo})$, respectively. We evaluate the test cost for m ranging from 1 to 64 and select the m for which the test cost is minimum. Fig. 11 presents the DSE results for Nova and Rocketcore (I). Lower weightage to the area overhead increases the engine count.

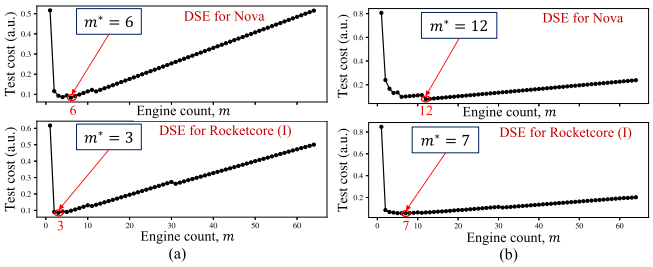

Fig. 11. Optimum BIST-engine counts (m^*) in Nova and Rocketcore. (a) w_a = 0.5 and w_{fo} = 0.3. (b) w_a = 0.2 and w_{fo} = 0.3.

TABLE III
IMPACT OF BIST INSERTION ON PPA OF M3D DESIGNS

Design	(m_α, t_α)	(m_β, t_β)	Metric	Non-BISTed	Δ_α (%)	Δ_β (%)
AES-128	(5,6)	(19,2)	Gate count	120,352	1.68	2.18
			Cell area (μm^2)	167,662.73	2.01	2.56
			Wirelength (m)	1.94	4.79	5.23
			Critical delay (ns)	2.07	0.76	3.21
			Total power (mW)	48.25	5.31	6.15
Nova	(6,4)	(12,3)	Gate count	148,410	1.06	2.10
			Cell area ((μm^2))	273,982.13	0.95	1.87
			Wirelength (m)	3.98	0.36	0.61
			Critical delay (ns)	6.94	0.00	0.00
			Total power (mW)	24.30	1.37	2.61
Rocket--core (I)	(3,103)	(7,52)	Gate count	330,385	1.73	2.25
			Cell area ((μm^2))	732,287.36	1.09	1.44
			Wirelength (m)	7.63	3.25	3.45
			Critical delay (ns)	7.65	0.00	0.00
			Total power (mW)	78.14	0.74	1.12
Rocket--core (II)	(3,92)	(7,48)	Gate count	300,227	1.90	2.47
			Cell area ((μm^2))	702,213.93	1.14	-23.73
			Wirelength (m)	7.29	3.52	3.41
			Critical delay (ns)	9.03	2.38	4.34
			Total power (mW)	64.59	0.75	1.11

(m_α, m_β): optimum BIST-engine counts for w_a = 0.5 and w_a = 0.2, respectively; (t_α, t_β): test-iteration counts corresponding to m_α and m_β, respectively; $(\Delta_\alpha, \Delta_\beta)$: BIST overhead for m_α and m_β, respectively.

Following BIST insertion in the gate-level partitioned design, the design is placed-and-routed by constraining the ILV locations to be the same as that before BIST insertion. Table III presents the PPA overhead of the inserted BIST configurations. The impact of BIST on chip area, timing, and

TABLE IV
SUMMARY OF POSTBOND 3-D IC TEST FRAMEWORKS ENABLING DETECTION AND LOCALIZATION OF FAULTS IN ILVS

Criteria	ILV-BIST	[29]	[30]	[27]
Fault detection	Yes	Yes	Yes	Yes
On-chip fault localization	Yes	Yes	Yes	No
On-chip localization granularity (ILV count)	3	N_{ch}	N_{ch}	-
Localization (on/off-chip) time (cycles)	$3k$	N_{ch}	N_{ch}	SC
Fault detection time (cycles)	$3t$	$\frac{N}{N_{ch}}$	$\frac{N}{N_{ch}}$	SC
Diagnosis of defect type	Yes	Yes	Yes	No
Number of BIST engines	m	1	1	-
Process variation-aware test	No	No	Yes	No

$t = \lceil \frac{N_{ns}}{mc} \rceil + t_S$; $SC = \lceil \log_2 N \rceil \cdot (2N + 1) + 2N$.

power consumption is low in all cases. We also evaluate the delay of the longest test path through an ILV that begins and ends at flops in the BIST logic. We observe that the longest test-path delay is smaller than the functional clock period for all four benchmarks. This implies that the test path does not violate circuit timing and will not cause timing failure in the BIST mode even if the ILVs are defect-free.

Insertion of multiple BIST engines leads to a higher area overhead than a single BIST engine. However, the wirelength overhead due to multiple BIST engines is expected to be comparable to that of a single engine. This is because, while the average physical distance between an ILV and a BIST engine is going to be smaller for multiple engines (leading to wirelength reduction), routing more BIST logic can lead to an increase in the wirelength. The final overhead numbers also depend on the extent of optimization carried out by the routing tool.

We compare the fault detection and localization features of our proposed framework with those supported by prior postbond test frameworks designed for testing TSVs [29], [30]. Both the prior test frameworks target resistive opens, resistive shorts, and SAFs in the TSVs. In other words, the target fault models in TSVs are the same as those used for the ILV test. The PPA overhead comprising gate count, wirelength, and path timing is difficult to analyze and compare as the final overhead numbers largely depend on the circuit size, analog or digital nature of the test macros, synthesis or design style of individual components of the test framework, and optimization during place-and-route.

Table IV summarizes the key features of ILV-BIST, postbond test frameworks proposed for TSVs in [29] and [30], and the IEEE 1838 Standard on die-wrapper based 3-D IC testing [27]. In [29], the ILVs are clustered into $N_{\text{ch}}(\approx 10)$ partitions based on their spatial proximity in the circuit layout. Assuming equal ILV counts in every cluster, (N/N_{ch}) is the number of concurrent test groups formed where N_{ch} ILVs in one concurrent group are tested simultaneously to obtain the pass/fail status of the given concurrent test group. As a result, the on-chip localization granularity is N_{ch}.

For the proposed ILV-BIST framework, m (≥ 2) is the Pareto-optimum number of BIST engines, c (≥ 16) is the number of ILVs tested concurrently by a single ILV-BIST engine, and t_S is the number of test iterations required to test for all potential ILV shorts. The number of faulty ILVs among the ILVs in a concurrent test group is denoted by k. Note that the localization time indicates the number of cycles required to identify all failing ILV(s) in a concurrent test group once a fault is detected. For [29] and [30], the localization granularity equals 1 for off-chip localization because the failing ILV(s) can be identified once the individual test responses of the N_{ch} ILVs in one concurrent test group are shifted out.

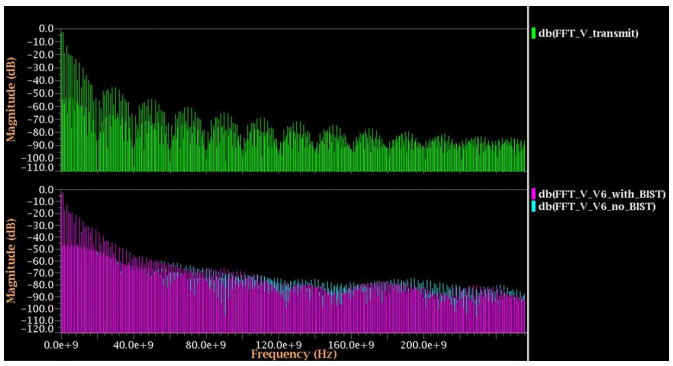

Fig. 12. Frequency-domain representation of signals transmitted and received through ILVs with and without BIST load.

C. Impact of BIST Insertion on Signal Distortion

Additional BIST circuitry results in an additional RC load being added to an ILV. The RC load due to the FO connections added between an ILV and the pins of the BIST-Capture engine results in additional signal-propagation delay through the ILV. The delayed signal can be viewed as a distorted form of the original signal transmitted into the ILV. We study the impact of increased wirelength (RC load) due to BIST insertion on signal distortion in the ILV at high frequencies.

For evaluating the effects on signal distortion, we carry out SPICE simulation of a lumped RC model of an ILV at 2 GHz [10]. Based on our PPA evaluation of benchmark circuits, the increase in wirelength per BIST engine is approximately 1.2%. First, we compute the RC values associated with the increased wirelength by considering wire resistance of 0.24 Ω per unit length and wire capacitance of 0.35 fF per unit length. We add this RC load as the BIST load to the ILV.

Next, we derive the weights of the frequency components of the transmitted signal and the received signal (both in the presence and the absence of BIST load) using fast Fourier transform (FFT). The transmitted signal is a pulse with the rise and fall delays equaling 0.05 ns. Fig. 12 shows the transmitted and received signals—with BIST and no BIST—in the frequency domain. We compute the percentage change in the root mean square (rms) of the frequency composition of the received signal with BIST load with respect to the received signal without BIST load. The change in rms frequency composition is an indicator of the harmonic distortion in the received signal due to the added BIST load. We find that the percentage change in the rms for the received signal with BIST load is only 0.8% with respect to the received signal without BIST load. Thus, even at high frequencies (2 GHz), the added BIST circuitry has a negligible effect on the harmonic distortion of the signal received at the ILV's output.

VII. CONCLUSION

We have presented a low-cost BIST architecture that requires only three test patterns to detect opens, SAFs, and

shorts in high-density ILV layouts. The BIST engine can also detect and localize single and multiple faults in the ILVs. Evaluation of PPA overhead for two-tier M3D benchmarks demonstrates the effectiveness of the proposed approach.

REFERENCES

[1] M. D. Bishop et al., "Monolithic 3-D integration," *IEEE Micro*, vol. 39, no. 6, pp. 16–27, Nov./Dec. 2019.

[2] P. Batude et al., "3D monolithic integration," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2011, pp. 2233–2236.

[3] Y. Li et al., "Monolithic 3D integration of logic, memory and computing-in-memory for one-shot learning," in *IEDM Tech. Dig.*, Dec. 2021, pp. 21.5.1–21.5.4.

[4] M. M. Shulaker, T. F. Wu, M. M. Sabry, H. Wei, H.-S. P. Wong, and S. Mitra, "Monolithic 3D integration: A path from concept to reality," in *Proc. DATE*, 2015, pp. 1197–1202.

[5] A. Koneru, S. Kannan, and K. Chakrabarty, "A design-for-test solution based on dedicated test layers and test scheduling for monolithic 3-D integrated circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 38, no. 10, pp. 1942–1955, Oct. 2019.

[6] A. Koneru, S. Kannan, and K. Chakrabarty, "Impact of electrostatic coupling and wafer-bonding defects on delay testing of monolithic 3D integrated circuits," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 13, no. 4, pp. 1–23, Oct. 2017.

[7] L. Brunet et al., "Breakthroughs in 3D sequential technology," in *IEDM Tech. Dig.*, Dec. 2018, pp. 7.2.1–7.2.4.

[8] G. Booker and R. Stickler, "Crystallographic imperfections in epitaxially grown silicon," *J. Appl. Phys.*, vol. 33, no. 11, p. 3281, 1962.

[9] A. Chaudhuri, S. Banerjee, H. Park, B. W. Ku, K. Chakrabarty, and S.-K. Lim, "Built-in self-test for inter-layer vias in monolithic 3D ICs," in *Proc. ETS*, May 2019, pp. 1–6.

[10] A. Chaudhuri et al., "Built-in self-test and fault localization for inter-layer vias in monolithic 3D ICs," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 18, no. 1, pp. 1–37, Jan. 2022.

[11] J. P. Shen et al., "Inductive fault analysis of MOS integrated circuits," *IEEE Design Test Comput.*, vol. 2, no. 6, pp. 13–26, Dec. 1985.

[12] T. Yonehara, "Epitaxial layer transfer technology and application," in *Proc. SOI-3D*, Oct. 2015, pp. 1–3.

[13] P. Batude et al., "3DVLSI with CoolCube process: An alternative path to scaling," in *Proc. Symp. VLSI Technol.*, Jun. 2015, pp. T48–T49.

[14] K. Dang, A. B. Ahmed, A. B. Abdallah, and X.-T. Tran, "TSV-IaS: Analytic analysis and low-cost non-preemptive on-line detection and correction method for TSV defects," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI (ISVLSI)*, Jul. 2019, pp. 501–506.

[15] H. Park, B. W. Ku, K. Chang, D. E. Shim, and S. K. Lim, "Pseudo-3D approaches for commercial-grade RTL-to-GDS tool flow targeting monolithic 3D ICs," in *Proc. Int. Symp. Phys. Design*, Mar. 2020, pp. 47–54.

[16] K. Chang et al., "Design automation and testing of monolithic 3D ICs: Opportunities, challenges, and solutions," in *Proc. ICCAD*, 2017, pp. 805–810.

[17] L. Brunet et al., "Direct bonding: A key enabler for 3D monolithic integration," *ECS Trans.*, vol. 64, no. 5, p. 381, 2014.

[18] C. C. Mee, M. K. M. Arshad, U. Hashim, and M. F. M. Fathil, "Impact of silicon epitaxial thickness layer in high power diode devices," *AIP Conf. Proc.*, vol. 1733, no. 1, 2016, Art. no. 020072.

[19] A. Koneru, S. Kannan, and K. Chakrabarty, "Impact of wafer-bonding defects on monolithic 3D integrated circuits," in *Proc. EPEPS*, Oct. 2016, pp. 91–94.

[20] N. Campregher et al., "Analysis of yield loss due to random photolithographic defects in the interconnect structure of FPGAs," in *Proc. ACM/SIGDA FPGA*, 2005, pp. 138–148.

[21] R. M. Geffken et al., "Method of forming a self-aligned copper diffusion barrier in vias," U.S. Patent 5 985 762, Nov. 16, 1999.

[22] M. Tsai, A. Klooz, A. Leonard, J. Appel, and P. Franzon, "Through silicon via (TSV) defect/pinhole self test circuit for 3D-IC," in *Proc. IEEE Int. Conf. 3D Syst. Integr.*, Sep. 2009, pp. 1–8.

[23] R. Pendurkar et al., "Switching activity generation with automated BIST synthesis for performance testing of interconnects," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 20, no. 9, pp. 1143–1158, Sep. 2001.

[24] J. Rajski and J. Tyszer, "Fault diagnosis of TSV-based interconnects in 3-D stacked designs," in *Proc. ITC*, Sep. 2013, pp. 1–9.

[25] A. Jutman, "Shift register based TPG for at-speed interconnect BIST," in *Proc. 24th Int. Conf. Microelectron.*, 2004, pp. 751–754.

[26] D. Erb, K. Scheibler, M. Sauer, S. M. Reddy, and B. Becker, "Multi-cycle circuit parameter independent ATPG for interconnect open defects," in *Proc. VTS*, Apr. 2015, pp. 1–6.

[27] A. Cron and E. J. Marinissen, "IEEE standard 1838 is on the move," *Computer*, vol. 54, no. 11, pp. 88–94, Nov. 2021.

[28] S. Thuries et al., "M3D-ADTCO: Monolithic 3D architecture, design and technology co-optimization for high energy efficient 3D IC," in *Proc. DATE*, Mar. 2020, pp. 1740–1745.

[29] Y.-W. Lee, H. Lim, S. Seo, K. Cho, and S. Kang, "A low-cost concurrent TSV test architecture with lossless test output compression scheme," *PLoS ONE*, vol. 14, no. 8, Aug. 2019, Art. no. e0221043.

[30] J. Mok, H. Lim, and S. Kang, "Enhanced postbond test architecture for bridge defects between the TSVs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 29, no. 6, pp. 1164–1177, Jun. 2021.

[31] S. Kannan et al., "Testing monolithic three dimensional integrated circuits," U.S. Patent 10 775 429, Sep. 15, 2020.

[32] *DEF Syntax*. Accessed: Aug. 11, 2022. [Online]. Available: http://coriolis.lip6.fr/doc/lefdef/lefdefref/DEFSyntax.html#Vias

[33] T. L. Michalka, R. C. Varshney, and J. D. Meindl, "A discussion of yield modeling with defect clustering, circuit repair, and circuit redundancy," *IEEE Trans. Semicond. Manuf.*, vol. 3, no. 3, pp. 116–127, Aug. 1990.

[34] J. P. de Gyvez and C. Di, "IC defect sensitivity for footprint-type spot defects," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 11, no. 5, pp. 638–658, May 1992.

[35] X. Jiang, Y. Hao, and G. Xu, "Equivalent circular defect model of real defect outlines in the IC manufacturing process," *IEEE Trans. Semicond. Manuf.*, vol. 11, no. 3, pp. 432–441, Aug. 1998.

[36] M. DeLeon, "A study of sufficient conditions for Hamiltonian cycles," *Rose-Hulman Undergraduate Math. J.*, vol. 1, no. 1, p. 6, 2000.

[37] A. Schrijver et al., *Combinatorial Optimization: Polyhedra and Efficiency*, vol. 24. Berlin, Germany: Springer, 2003.

[38] A. C.-C. Yao, "New algorithms for bin packing," *J. ACM*, vol. 27, no. 2, pp. 207–227, Apr. 1980.