Test-Architecture Optimization and Test Scheduling for TSV-Based 3-D Stacked ICs

Brandon Noia, Krishnendu Chakrabarty, *Fellow, IEEE,* Sandeep Kumar Goel, *Senior Member, IEEE,* Erik Jan Marinissen, *Fellow, IEEE,* and Jouke Verbree

***Abstract*—Through-silicon via (TSV)-based 3-D stacked ICs (SICs) are becoming increasingly important in the semiconductor industry. In this paper, we address test architecture optimization for 3-D stacked ICs implemented using TSVs. We consider two cases, namely 3-D SICs with die-level test architectures that are either fixed or still need to be designed. We next present mathematical programming techniques to derive optimal solutions for the architecture optimization problem for both cases. Experimental results for three handcrafted 3-D SICs comprising of various systems-on-a-chip (SoCs) from the ITC'02 SoC test benchmarks show that compared to the baseline method of sequentially testing all dies, the proposed solutions can achieve significant reduction in test length. This is achieved through optimal test schedules enabled by the test architecture. We also show that increasing the number of test pins typically provides a greater reduction in test length compared to an increase in the number of test TSVs. Furthermore, we show that shorter test lengths are generally achieved with the larger, more complex dies lower in the stack. This is because test data must pass through every die lower in a stack in order to reach its target die, and with the larger dies lower in the stack, more test bandwidth may be provided to these dies using fewer routing resources.**

***Index Terms*—3-D SIC, DFT, ILP, optimization.**

I. Introduction

THE semiconductor industry is pushing relentlessly for high-performance and low-power chips. Recent advances in semiconductor manufacturing technology have enabled the creation of complete systems with direct stacking and bonding of die-on-die. These system chips are commonly referred to as 3-D stacked ICs (SICs). Although a number of 3-D integration methods have been proposed in the literature, in this paper we focus on through-silicon via (TSV) vertical interconnects, as it offers the promise of the highest vertical interconnect density among the proposed technologies. Using TSV technology, 3-D ICs are created by placing multiple device layers together through wafer or die stacking, and these are then connected using vertical TSVs [1].

The promise of 3-D IC technology lies in the numerous benefits it can potentially provide over traditional 2-D ICs [2]. Due to the steady increase in chip complexity, continued technology scaling, and reduced voltage levels, interconnects have become longer in 2-D ICs, leading to increased circuit delay and power consumption. 3-D ICs will lead to a reduction in the average interconnect length and help obviate the problems caused by long global interconnects [1], [3], [4]. Since dies can be stacked in a 3-D environment, on-chip data bandwidth can be increased as well. Furthermore, since 3-D ICs allow chips to grow in the vertical dimension instead of requiring larger die area, higher packing density and smaller footprint can be achieved.

The manufacturing of functional 3-D SICs has recently been demonstrated. Memories are easier to stack compared to logic due to high yields after repair and simplified testing and design [5], and as such 3-D memory stacks have already been manufactured [5]. Stacks that include memory stacked on logic [6] or multiple logic dies [7] are likely to be seen in the near future. Although 3-D design-and-test automation is not yet fully mature for commercial exploitation, it is well on its way [8]. These tools need to be able to exploit the benefits of 3-D technologies, while taking into account the various design-related tradeoffs. For example, in a TSV-based 3-D SIC, the number of TSVs for test access is limited because of their associated chip area costs. Most TSVs are likely to be dedicated for functional access, power/ground, and clock routing.

Testing core-based dies in 3-D SICs brings forward new challenges [9], [10]. In order to test the dies and associated cores, a test access mechanism (TAM) must be included on the dies to transport test data to the cores, and a 3-D TAM is needed to transfer test data to the dies from the stack input/output pins. TAM design in 3-D SICs involves additional challenges compared to TAM design for 2-D SoCs. In a 3-D SIC, a test architecture must be able to support testing of individual dies as well as testing of partial and complete stacks [10]. Furthermore, test architecture optimization must not only minimize the test length, but it also needs to minimize

Manuscript received March 22, 2011; accepted May 26, 2011. Date of current version October 19, 2011. This work was supported in part by the National Science Foundation, under Grant CCF-1017391, and by a contract from Semiconductor Research Corporation (SRC). The work of B. Noia was also supported in part by a Masters Scholarship from SRC. A preliminary version of this paper was presented at the IEEE European Test Symposium, 2010. This paper was recommended by Associate Editor C.-W. Wu.

B. Noia and K. Chakrabarty are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708 USA (e-mail: brn2@ee.duke.edu; krish@ee.duke.edu).

S. K. Goel was with Duke University, Durham, NC 27708 USA. He is now with TSMC Technology, Inc., San Jose, CA 95134 USA (e-mail: skgoel@tsmc.com).

E. J. Marinissen is with the 3-D Integration Program, IMEC, Leuven B-3001, Belgium (e-mail: erik.jan.marinissen@imec.be).

J. Verbree was with IMEC, Leuven B-3001, Belgium. He is now with the Department of Computer Engineering, Delft University of Technology, Delft 2628CD, The Netherlands (e-mail: j.verbree@student.tudelft.nl).

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.2011.2160177

the number of TSVs used to route the 3-D TAM, as each TSV has area costs associated with it and is a potential source of defects in an 3-D SIC. The test length is therefore dependent on the test architecture and test schedule and constrained by a limit on the available test resources.

In this paper, we address the problem of test-architecture optimization of 3-D SICs with: 1) *hard dies*, in which a test architecture already exists; 2) *soft dies*, for which we also design the test architecture within each die; and 3) *firm dies*, in which a test architecture already exists but serial/parallel conversion hardware may be added to the die in order to reduce test-pin and TSV use and achieve better test resource allocation for stack testing. For the sake of simplicity and ease of implementation, we assume session-based test scheduling [33], i.e., all tests which are executed simultaneously need to be completed before a next session with tests is started. We further present methods for minimizing the number of TSVs or test pins used for target test lengths. While it is theoretically possible to have multiple dies on a given layer in a stack, we only consider one die per layer in a stack. Also, a core is considered to be part of a single die only, i.e., we do not consider "3-D cores." In addition to minimizing the test length for each soft die, we minimize the test length for the complete stack in all three problem instances.

The key contributions of this paper are as follows.

1) We present integer linear programming solutions for three different, but realistic, 3-D test-architecture optimization problems. These solutions produce optimal TAM designs and test schedules under test-access constraints for three design scenarios, referred to as hard, soft, and firm dies. The optimization models are rigorously derived and their optimality is formally established. We show that these methods result in significantly decreased test time over heuristic methods for 3-D TAM design.
2) We extensively study the effects of a number of parameters on the optimization results; the results provide insights to designers on how to approach design for testability for dies prior to 3-D integration, how to design the 3-D stack itself, and how to best allocate the limited test resources for the 3-D SIC. The parameters of interest include the placement of die in a stack, the availability of test TSVs and test pins.
3) We show that, given test bandwidth and TSV constraints, Pareto-optimality exists in the solution space. This finding is especially significant for the hard die case, although it is also present for the case of firm dies. These results imply that, using our models, designers can explore many test solutions for the hard and firm die scenarios in order to prevent sub-optimal allocation of test resources.
4) Firm and soft die are shown to provide lower test times with an increase in test resources. Since both scenarios result in lower test times than the case of hard dies, designers can consider 2-D and 3-D TAM designs as related to optimization problems.

The rest of this paper is organized as follows. Section II provides an overview of the related prior work. Section III uses a simple example to motivate this paper and formally describes the three problems addressed in this paper. Section IV presents integer linear programming (ILP) models to solve the test-architecture optimization problems described in Section III. Section V presents experimental results for various 3-D SICs constructed using SoCs from the ITC'02 SoC test benchmarks [11]. Finally, Section VI concludes this paper.

II. Related Prior Work

Testing of 2-D SoCs and the optimization of related test-access architectures have been well studied [12]–[15]. Optimization methods have included ILP [12], rectangle packing [12], [16], iterative refinement [14], and other heuristics [15], [17]. However, all these methods were originally developed for 2-D SoCs, and the added test complexities related to 3-D technology were not considered.

Recently, some early work has been reported on testing of 3-D SICs. Heuristic methods for designing core wrappers in 3-D SICs were developed in [18]. These methods do not address the problem of 3-D TAM design. ILP models for test architecture design for each die in a stack are presented in [20]. While these ILP models take into account some of the constraints related to 3-D SIC testing such as a TSV limit, this approach does not consider the reuse of die-level TAMs. A TAM wire-length minimization technique based on simulated annealing is presented in [21]. A drawback of this approach is that it implies a 3-D test architecture that is not feasible in practice. Heuristic methods for reducing weighted test cost while taking into account the constraints on test pin widths in pre-bond and post-bond tests are described in [25]. An unrealistic assumption made in [25] is that TAMs can start and terminate in any layer.

In most prior work on 3-D SIC testing, TAM optimization is performed at die-level only, which leads to inefficient TAMs and non-optimal test schedules for partial-stack and complete-stack tests. Furthermore, all previous methods assume that the designer can create TAM architectures on each die during optimization, which may not be possible in all cases. In [26], a die-level wrapper and associated 3-D architecture is presented to allow for all pre-bond and post-bond tests. This approach proposes die-level wrappers and it leverages current standards, IEEE 1149.1 and IEEE 1500. In addition to functional and test modes, die-level wrappers allow bypass of test data to and from higher die in the stack and reduced test bandwidth during pre-bond tests. This is a realistic and practical look at test architectures in 3-D SICs, but it offers no insight into optimization and test scheduling. The optimization methods presented in our paper are compatible with the test architecture of [26], and they do not make any unrealistic assumptions on die wrappers or the 3-D TAM. Our paper considers test-architecture optimization for the entire stack and considers 3-D stacks with hard, soft, or firm dies. We also explore the effect of available test pins, test TSVs used, and die ordering on TAM design and test scheduling.

We do not consider pre-bond testing in this paper. The reader is referred to [22], [24], and [27] for emerging techniques for pre-bond testing. In other related work, [23] described a test architecture that reuses pre-bond test hardware

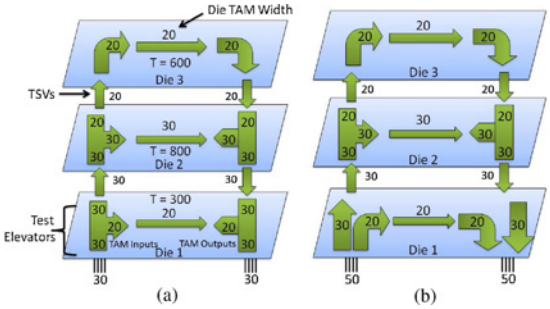

Fig. 1. Example 3-D SIC with three hard dies. (a) Serial test architecture. (b) Parallel test architecture.

for post-bond test. In [28], an architecture is proposed for pre-bond testing. The authors of [29] proposed a pre-bond-testable clock tree design, and in [30] TSV test challenges are examined for a stacked DRAM.

III. Problem Definition

In a 3-D SIC, which currently consists of anywhere from two to eight dies [31], the lowest die is usually directly connected to chip I/O pins, therefore it can be tested using test pins. To test the non-bottom dies in the stack, test data must enter through the test pins on the lowest die. Therefore, to test other dies in the stack, the TAM must be extended to all dies in the stack through the test pins at the lowest die. To transport test data up and down the stack, "TestElevators" [26] need to be included on each die except for the highest die in the stack [10]. The number of test pins and TestElevators as well as the number of TSVs used affect the total test length for the stack.

Consider an example 3-D SIC with three dies with given test access architectures as shown in Fig. 1. Suppose the test lengths for Die 1, Die 2, and Die 3 are 300, 800, and 600 clock cycles, respectively. The total number of available test pins at the bottom die is 100. Die 1 requires 40 test pins (TAM width of 20), and Dies 2 and 3 require 60 TestElevators and 40 TestElevators, respectively. The test length for each die is determined by its test architecture.

Fig. 1(a) shows the TestElevator widths and the number of TSVs used if all dies are tested serially. In this case, a total of 100 TSVs are used, and 100 test pins are available, from which only 60 are utilized. The total test length for the stack is the sum of the test lengths of the individual dies, i.e., 1700 cycles. Fig. 1(b) shows the test architecture required if Die 1 and Die 2 are tested in parallel. In this case, the number of TSVs used is the same as in Fig. 1(a). However, all 100 test pins are required to test Die 1 and Die 2 in parallel. Also, 60 TestElevators must pass between Die 1 and Die 2 in order to pass a separate 30-bit wide TAM to Die 2 for parallel testing. For this case, the total test length for the stack is $\max\{300, 800\} + 600 = 1400$ cycles. This example clearly shows that there is a tradeoff between test length and the number of test pins and TSVs used. Therefore, a test-architecture optimization algorithm for 3-D SICs has to minimize the test length while taking into account upper limits on the number of test pins and TSVs used.

Test-architecture optimization for 3-D SICs with hard dies is illustrated in Fig. 2. For a hard die, the 2-D test architecture on the die is fixed. The only structures over which the designer has control are the 3-D TAM. Hard dies offer less flexibility for optimization in the sense that each die must have exactly the pre-defined number of input and output TAM wires appropriated to it in the design of the 3-D TAM. Therefore, the only decisions that can be made in designing the 3-D TAM are which (if any) die can be tested in parallel with one another given the limitations on test pins and test TSVs. Hard dies may be present in TAM design problems if vendors sell fabricated dies to a 3-D integrator.

Fig. 2(a) illustrates the variables that arise for the hard-die problem. As can be seen, a fixed 2-D TAM width is given along with the known test time for each die. The given constraints are the number of test pins $W_{\max}$ and the number of test TSVs $TSV_{\max}$ available. A solution, therefore, can be given as in Fig. 2(b). Here, each die receives the required and pre-defined test bandwidth, but Die 1 and Die 2 are tested in parallel through the 3-D TAM.

We denote the test-architecture optimization problem for hard dies as PSHD, where "PS" stands for "problem statement" and "HD" stands for "hard dies." The problem can be defined as follows.

Problem 1: 3-D SIC with Hard Dies (PSHD).

Given a stack with a set M of dies, total number of test pins $W_{\max}$ available for test, and a maximum number of TSVs ($TSV_{\max}$) that can be used globally (throughout the entire stack) for TAM design. For each die $m \in M$, the die's number corresponds to its tier in the stack (Die 1 is the bottom die, Die 2 is next, and so forth), and we are given the number of test pins on each die w_m ($w_m \leq W_{\max}$) required to test the die, and the associated test length t_m (since the test architecture per die is given, t_m is also given). Determine an optimal TAM design and corresponding test schedule for the stack such that the total test length T for the stack is minimized and the number of TSVs used does not exceed $TSV_{\max}$.

Two dual problems, Pshdt (the "T" stands for TSV minimization) and Pshdw (the "W" stands for test-pin-count minimization), respectively, can be stated as follows. For Pshdt, we determine an optimal TAM design and corresponding test schedule for the stack such that the total number of TSVs used for the stack is minimized and the upper limits on test length $T_{\max}$ and test pin count $W_{\max}$ are not exceeded. For Pshdw, we determine an optimal TAM design and test schedule for the stack such that the total number of test pins used for the stack is minimized and the upper limits on test length $T_{\max}$ and total number of TSVs ($TSV_{\max}$) are not exceeded.

The hard-die model is based on prior work on SoC testing [32], with additional constraints, the firm and soft die models are considerably different and more complex. Besides simply adding 3-D design constraints, each die must be considered across a range of many different possible TAM widths and we have to consider many variations in which dies are tested in parallel as we move along a 3-D stack. These considerations require the addition of many more variables and constraints. Overall, these additions make the firm-die and soft-die models significantly more complex than the hard-die model, potentially limiting the number of die that can be included in the model before run time becomes prohibitively high.

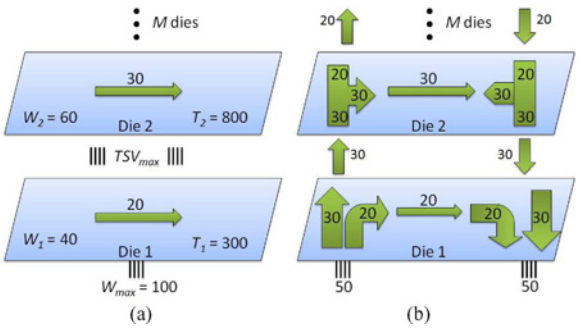

Fig. 2. Illustration of PSHD. (a) Problem instance. (b) Optimized architecture.

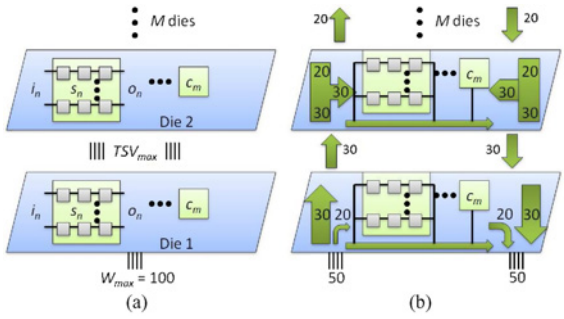

Fig. 3. Illustration of PSSD. (a) Problem instance. (b) Optimized architecture.

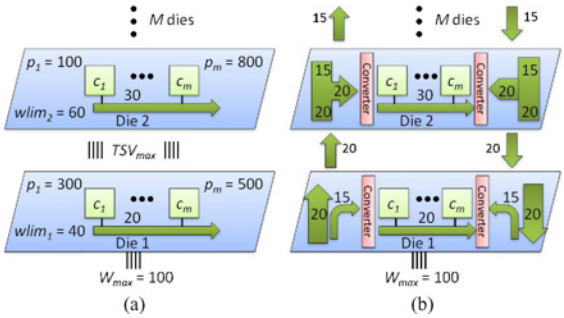

Fig. 4. Illustration of PSFD. (a) Problem instance. (b) Optimized architecture.

The above problem statement is different for a 3-D SIC with soft dies. In the case of soft dies, the test architecture for each die is not pre-defined, but it is determined during the test-architecture design for the stack. In this case, both the 2-D and 3-D TAMs are co-designed. Scan chains for each test module are given, but the test wrappers for each module and the TAM are designed during 3-D TAM design. This allows the designer to develop the most efficient 2-D/3-D TAM designs given TSV-count and test-pin-count constraints. Soft dies model the additional flexibility available for optimization when dies are fabricated in-house for 3-D integration.

Test-architecture optimization for 3-D SICs with soft dies is illustrated in Fig. 3. Fig. 3(a) shows the known quantities associated with the soft die model, namely a number of modules per die, with pre-defined scan-chains per die, and $W_{\max}$ and $TSV_{\max}$. Fig. 3(b) shows the result of optimization, including wrapper, 2-D TAM, and 3-D TAM designs.

The test-architecture optimization problem for soft dies can be formally defined as follows.

Problem 2: 3-D SIC with Soft Dies (PSSD).

Given a stack with a set M of dies, the total number of test pins $W_{\max}$ available for test at the lowest die, and a maximum number of TSVs ($TSV_{\max}$) that can be used for TAM design. For each die $m \in M$, we are given the total number of cores c_m. Furthermore, for each core c, the number of inputs i_c, outputs o_c, total number of test patterns p_c, total number of scan chains s_c, and for each scan chain v, the lengths of the scan chain in flip flops $l_{c,v}$ are given. Determine an optimal TAM design and test schedule for the stack, as well as for each die, such that the total test length T for the stack is minimized and the number of TSVs used does not exceed $TSV_{\max}$.

Two dual problems, Pssdt and Pssdw, respectively, can again be stated as follows. For Pssdt, we determine an optimal TAM design and test schedule for the stack and for each die such that the total number of TSVs used for the stack is minimized and the upper limits on test length $T_{\max}$ and test pin count $W_{\max}$ are not exceeded. For Pssdw, we determine an optimal TAM design and test schedule for the stack as well as for each die such that the total number of test pins used for the stack is minimized and the upper limits on test length $T_{\max}$ and total number of TSVs ($TSV_{\max}$) are not exceeded.

Finally, we present the problem statement for a 3-D SIC with firm die. In the case of firm dies, the test architecture for each die is pre-defined as for a hard die, but additional serial/parallel conversion hardware may be added to the die to allow for fewer test elevators (or test pins in the case of the lowest die) to be used than the fixed 2-D TAM width for the die. The conversion hardware is added before the inputs and after the outputs of the die wrapper. The input hardware multiplexes a smaller number of TAM wires to a larger number of die wrapper wires. Demultiplexers at the output of the die wrapper transfer test responses from a larger number of die wrapper wires to a smaller number of TAM wires. Compared to the scenario involving hard dies, this scenario allows the use of fewer test pins at the expense of higher test lengths, but it also allows additional flexibility in test scheduling and test-time optimization.

The problem of test-architecture optimization for 3-D SICs with firm dies is shown in Fig. 4. Fig. 4(a) shows the known quantities for the firm die problem; these are similar to the hard die problem except that test times are given for certain serial/parallel conversion bandwidths for each die. During optimization, one of these converters (or no converter) can be used, as seen in Fig. 4(b). For Die 1, for example, a 3-D TAM width of 15 bits is used, though the 2-D TAM was designed for a width of 20 bits.

The test-architecture optimization for firm die is formally defined as follows.

Problem 3: 3-D SIC with Firm Dies (PSFD).

Given a stack with a set M of dies, the total number of test pins $W_{\max}$ available for test at the lowest die, and a maximum number of TSVs ($TSV_{\max}$) that can be used for TAM design. For each die $m \in M$, we are given a fixed 2-D TAM architecture with the total number of cores c_m, which TAM partitions they utilize, and their TAM widths. Furthermore, for each core n, the total number of test patterns p_n is given and we are given the number of test pins $wlim_m$ required to test the

die. Determine an optimal TAM design and test schedule for the stack, as well as possible serial/parallel conversion widths for each die, such that the total test length T for the stack is minimized and the number of TSVs used does not exceed $TSV_{\max}$.

Problems 1–3 are all $\mathcal{NP}$-hard ("proof by restriction"), as they can be reduced using standard techniques to the rectangle packing problem, which is known to be $\mathcal{NP}$-hard [34]. For example, for PSHD, if we remove the constraints related to maximum number of TSVs, each die can be represented as a rectangle with width equal to its test length and height equal to the number of required test pins. Now we need to pack all these rectangles (dies) into a bin with width equal to the total number of test pins and height equal to the total test length for the stack, which needs to be minimized. Similarly, for PSSD, a rectangle must also be selected for each die from a set of rectangles with different widths and heights, but a special case of the scenario is identical to PSHD. Despite the $\mathcal{NP}$-hard nature of these problems, they can be solved optimally since the number of layers in a 3-D SIC is expected to be limited, e.g., up to four layers have been predicted for logic stacks [35].

The above problems are more general than the combinatorial problem of rectangle packing. The added 3-D design constraints and the greater design freedom available, especially for firm and soft dies, drastically increase the solution space. Rectangle packing is only a special and a considerably more simple case of our problem statements.

IV. Test-Architecture Optimization

In this paper, we use ILP to solve the problems defined in the previous section. Although ILP methods do not scale well with problem instance size, the problem instance sizes for P_{SHD} and P_{SSD} are relatively small for realistic stacks, therefore ILP methods are good candidates for solving them.

A. ILP Formulation for PSHD

To create an ILP model for this problem, we need to define the set of variables and constraints. We first define a binary variable x_{ij}, which is equal to 1 if die i is tested in parallel with die j, and 0 otherwise. Constraints on variable x_{ij} can be defined as follows:

$$x_{ii} = 1 \qquad \forall i \tag{1}$$

$$x_{ij} = x_{ji} \qquad \forall i, j \tag{2}$$

$$1 - x_{ij} \geq x_{ik} - x_{jk} \geq x_{ij} - 1 \qquad \forall i \neq j \neq k. \tag{3}$$

The first constraint indicates that every die is always considered to be tested with itself. The second constraint states that if die i is tested in parallel with die j, then die j is also tested in parallel with die i. The last constraint ensures that if die i is tested in parallel with die j, then it must also be tested in parallel with all other dies that are tested in parallel with die j.

Next, we define a second binary variable y_i, which is equal to 0 if die i is tested in parallel with die j on a lower layer ($l_i > l_j$), and 1 otherwise. The total test length T for the stack is the sum of test lengths of all dies that are tested in series plus the maximum of the test lengths for each of the sets of parallel tested dies. Using variables x_{ij} and y_i, the total test length T for a stack with set of dies M can be defined as follows:

$$T = \sum_{i=1}^{|M|} y_i \cdot \left(\max_{j=i..|M|} \{x_{ij} \cdot t_j\} \right). \tag{4}$$

An inductive proof of correctness of (4) is presented in the appendix. It should be noted that (4) has two non-linear elements, the *max* function, and the product of variable y_i and the *max* function. We linearize this by introducing two new variables. The variable c_i takes the value of the *max* function for each die i and the variable u_i represents the product $y_i \cdot c_i$. The variables u_i and c_i are defined using standard linearization techniques as shown in Fig. 5. The linearized function for total test length can be written as follows:

$$T = \sum_{i=1}^{|M|} u_i. \tag{5}$$

As the number of test pins used for parallel testing of dies should not exceed the given test pins $W_{\max}$, a constraint on the total number of pins used to test all dies in a parallel set can be defined as follows. In the inequalities, w_j refers to the TAM width for die j

$$\sum_{j=1}^{|M|} x_{ij} \cdot w_j \leq W_{\max} \qquad \forall i. \tag{6}$$

Similarly, the total number of used TSVs should not exceed the given TSV limit $TSV_{\max}$. The number of TSVs used to connect layer i to layer $i-1$ is the maximum of the number of pins required by the layer at or above layer i that takes the most test pin connections, and the sum of parallel tested die at or above layer i in the same parallel tested set. Based on this, we can define the constraint on the total number of TSVs used in a test architecture as follows:

$$\sum_{i=2}^{|M|} \{\max_{k=i}^{|M|} \{w_k, \sum_{j=k}^{|M|} w_j \cdot x_{kj}\}\} \leq TSV_{\max}. \tag{7}$$

We can linearize the above set of constraints by representing the *max* function by a variable d_i. Finally, to complete the ILP model for PSHD, we must define constraints on binary variable y_i and the relationship between binary variable y_i and x_{ij}. For this purpose, we first define a constant C that approaches but is less than 1. We then define y_i as follows:

$$y_1 = 1 \tag{8}$$

$$y_i \geq \frac{1}{1-i} \sum_{j=1}^{i-1} \left(x_{ij} - 1\right) - C \qquad \forall i > 1. \tag{9}$$

Equation (8) forces y_1 to 1, since the lowest layer cannot be tested in parallel with any layer lower than itself. Constraint (9) defines y_i for the other layers. To understand this constraint, we first make the observation that the objective function [as

Objective:

Minimize $\sum_{i=1}^{|M|} u_i$

Subject to:

$t_{max} = \max_{i=1}^{|M|} t_i$
$c_i \geq x_{ij} \cdot t_j \quad \forall i, j = i..|M|$
$u_i \geq 0 \quad \forall i$
$u_i - t_{max} \cdot y_i \leq 0 \quad \forall i$
$u_i - c_i \leq 0 \quad \forall i$
$c_i - u_i + t_{max} \cdot y_i \leq t_{max} \quad \forall i$
$\sum_{j=1}^{|M|} x_{ij} \cdot w_j \leq W_{max} \quad \forall i$
$x_{ii} = 1 \quad \forall i$
$x_{ij} = x_{ji} \quad \forall i, j$
$1 - x_{ij} \geq x_{ik} - x_{jk} \geq x_{ij} - 1 \quad \forall i \neq j \neq k$
$\sum_{i=2}^{|M|} d_i \leq TSV_{max} \quad \forall i$
$d_i \geq \sum_{j=k}^{|M|} w_j \cdot x_{kj} \quad \forall i, k = i..|M|$
$d_i \geq w_j \quad \forall i, j = i..|M|$
$y_1 = 1$
$y_i \geq \frac{1}{1-i} \sum_{j=1}^{i-1} (x_{ij} - 1) - C \quad \forall i > 1$

Fig. 5. ILP model for 3-D TAM optimization PSHD.

shown in (4)] would be minimized if each y_i is zero. This would make the objective function value equal to 0, which is an absolute minimum test length. Thus, we only need to restrict y_i to 1 where it is absolutely necessary, and then we can rely on the objective function to assign a value 0 to all unrestricted y_i variables. This equation considers the range of values that the sum of x_{ij} can take. The fraction in the equation normalizes the sum to a value between 0 and 1 inclusive, while the summation considers all possible cases for a die being tested in parallel with a die below it. A formal proof of correctness of this constraint is given in the appendix. The complete ILP model is shown in Fig. 5.

A benefit of using ILP is that the dual problems Pshdt and Pshdw can be easily tackled by appropriately modifying the model of Fig. 5. For both Pshdt and Pshdw, we introduce a maximum test length constraint $T_{\max}$ and add the following inequality to the model:

$$\sum_{i=1}^{|M|} u_i \leq T_{\max}.$$

As can be easily seen, the previous objective function is now transformed into a constraint. For Pshdt, we remove the constraint on TSVs used, which is the inequality involving $TSV_{\max}$, and use the following objective function:

$$\text{Minimize} \sum_{i=2}^{|M|} d_i.$$

For Pshdw, we remove the constraint on the number of test pins used, which is the inequality involving $W_{\max}$, and introduce the variable P to represent the number of test pins used by the stack. The following inequalities define P:

$$P \geq \sum_{j=1}^{|M|} x_{ij} \cdot w_j \qquad \forall i. \tag{10}$$

Our objective for Pshdw is therefore to minimize P.

B. ILP Formulation for PSSD

The ILP formulation for 3-D SICs with soft cores is derived in a similar manner as that for 3-D SICs with hard cores. In this

Objective:

Minimize $\sum_{i=1}^{N} u_i$

Subject to:

$t_{max} = \max_{i=1}^{|M|} t_i$
$c_i \geq v_{ij} \quad \forall i, j = i..|M|$
$v_{ij} \geq 0 \quad \forall i, j$
$v_{ij} - t_{max} \cdot x_{ij} \leq 0 \quad \forall i, j = 1..|M|$
$-\sum_{n=1}^{k_i} (g_{jn} \cdot t_j(n)) + v_{ij} \leq 0 \quad \forall i, j$
$\sum_{n=1}^{k_i} (g_{jn} \cdot t_j(n)) - v_{ij} + t_{max} \cdot x_{ij} \leq t_{max} \quad \forall i, j$
$u_i \geq 0 \quad \forall i$
$u_i - t_{max} \cdot y_i \leq 0 \quad \forall i$
$u_i - c_i \leq 0 \quad \forall i$
$c_i - u_i + t_{max} \cdot y_i \leq t_{max} \quad \forall i$
$z_{ijk} \geq 0 \quad \forall i, j, k$
$z_{ijk} - t_{max} \cdot x_{jk} \leq 0 \quad \forall i, j, k$
$-wi + z_{ijk} \leq 0 \quad \forall i, j, k$
$w_i - z_{ijk} + t_{max} \cdot x_{jk} \leq t_{max} \quad \forall i, j, k$
$\sum_{i=2}^{|M|} d_i \leq TSV_{max}$
$d_i \geq \sum_{j=k}^{|M|} z_{jkj} \quad \forall i, k = i..|M|$
$d_i \geq w_j \quad \forall i, j = i..|M|$
$\sum_{j=1}^{|M|} z_{jij} \leq W_{max} \quad \forall i$
$x_{ii} = 1 \quad \forall i$
$x_{ij} = x_{ji} \quad \forall i, j$
$1 - x_{ij} \geq x_{ik} - x_{jk} \geq x_{ij} - 1 \quad \forall i \neq j \neq k$
$y_1 = 1$
$y_i \geq \frac{1}{1-i} \sum_{j=1}^{i-1} (x_{ij} - 1) - M \quad \forall i > 1$

Fig. 6. ILP model for 3-D TAM optimization PSSD.

case, the test length t_i for die i is a function of the TAM width w_i assigned to it. Using the variables x_{ij} and y_i as defined in Section IV-A, the total test length T for the stack with the set of soft dies M can be defined as follows:

$$T = \sum_{i=1}^{|M|} y_i \cdot \max_{j=i..|M|} \{x_{ij} \cdot t_j(w_j)\}. \tag{11}$$

It should be noted that (11) has several non-linear elements. To linearize this equation, first we must define the test length function. For this purpose, we introduce a binary variable $g_{in} = 1$ if $w_i = n$, and 0 otherwise. We then linearize this expression using the variable v_{ij} for $x_{ij} \cdot \sum_{n=1}^{k_i} (g_{jn} \cdot t_j(n))$. Similar to (5), the variable c_i takes the value of the max function for each die i and the variable u_i represents the product $y_i \cdot c_i$. Since w_j is now a decision variable, we linearize $x_{ij} \cdot w_j$ using a new variable z_{ijk} defined for all i, j, k. We represent the max function by the variable d_i as before. By using the variable z_{ijk}, the TAM width that can be given to each die can be constrained by an upper limit, which is the number of available test pins. We represent this with the following set of inequalities. The complete ILP model for PSSD is shown in Fig. 6

$$\sum_{j=1}^{|M|} z_{jij} \leq W_{\max} \qquad \forall i. \tag{12}$$

As before, we make alterations to the ILP model to solve the dual problems Pssdt and Pssdw. For both Pssdt and Pssdw, as with the hard die dual problems, we introduce a maximum test length constraint $T_{\max}$ and add the following constraint to the problem:

$$\sum_{i=1}^{|M|} u_i \leq T_{\max}. \tag{13}$$

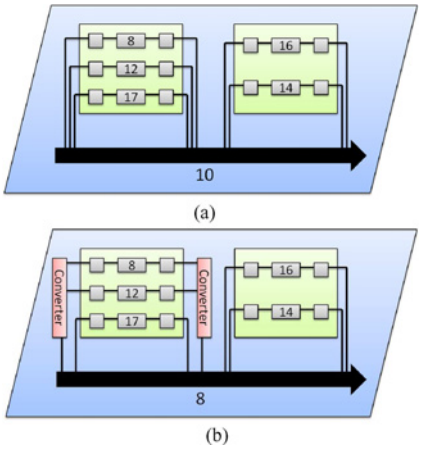

Fig. 7. Illustration of TAM width reduction using serial/parallel conversion. (a) Without conversion. (b) With conversion.

For Pssdt, we remove the constraint on TSVs used and use the following objective function:

$$\text{Minimize} \sum_{i=2}^{|M|} d_i.$$

For Pshdt, we remove the constraint on the number of test pins used, which is the inequality involving $W_{\max}$, and introduce the variable P once again. The following inequality defines P:

$$P \geq \sum_{j=1}^{|M|} z_{jij} \qquad \forall i. \tag{14}$$

Our objective for PSHDT is therefore to minimize P.

C. ILP Formulation for PSFD

The ILP formulation for 3-D SICs with firm dies is an extension to the model for soft dies. We add a constraint to indicate that the number of test pins used for a die cannot exceed the number of test pins required by the fixed 2-D TAM for that die. This constraint is expressed as follows:

$$w_i \leq wlim_i \qquad \forall i \tag{15}$$

where $wlim_i$ is the number of test pins required by the 2-D TAM on each die i prior to any serial/parallel conversion. In order to accurately determine test lengths for the dies using serial/parallel conversion, we modify the control-aware TAM design method of [36] to allow the architecture to be fixed in terms of assignment of modules to TAM partitions for a die. We then iteratively reduce the effective widths of the TAM partitions and reoptimize, thereby determining the optimal serial/parallel conversion to use depending on the bandwidth given to that die, as shown in Fig. 7. Fig. 7(a) shows a die prior to TAM width reduction, where ten pins are required to test the die. There are two cores, one with three wrapper chains consisting of the given number of scan flops, and another with two wrapper chains. The amount of time needed to test each core is dependent on the length of the longest wrapper chain and the number of test patterns required by the core. In this example, we assume that both cores require the same number of test patterns. Therefore, when we reduce the TAM width by two, it is best to combine the wrapper chains of length 8 and 12 in the first core, resulting in a longest wrapper chain of 20 as seen in Fig. 7(b).

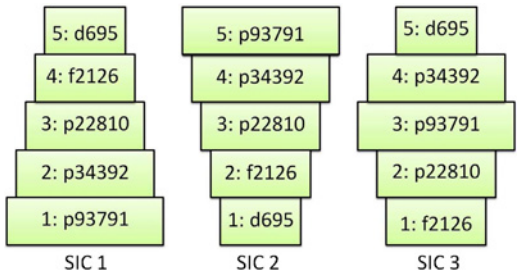

Fig. 8. Three 3-D SIC benchmarks.

TABLE I
TEST LENGTHS AND NUMBER OF TEST PINS FOR DIES AS REQUIRED IN PSHD

Die	d695	f2126	p22810	p34392	p93791
Test length	96 297	669 329	651 281	1 384 949	1 947 063
Test pins	15	20	25	25	30

V. EXPERIMENTAL RESULTS

In this section, we present experimental results for the ILP models presented in the previous section. As benchmarks, we have handcrafted three 3-D SICs (as shown in Fig. 8) out of several SoCs from the ITC'02 SoC Test Benchmarks as dies inside the 3-D SICs. The SoCs used are d695, f2126, p22810, p34292, and p93791. In SIC 1, the die are ordered such that the lowest die is the most complex (p93791), with dies increasing in complexity as one moves higher in the stack. The order is reversed in SIC 2, while for SIC 3, the most complex die is placed in the middle of the stack, with dies decreasing in complexity moving out from that die. For equal test bitwidths, the dies lowest in the stack in SIC 1 would have the highest test times. In Table I, f2126 has a slightly higher test time than p22810 because it has a smaller test bitwidth. P22810, however, is still the more complex die from a test perspective. Since SIC 1 and SIC 2 are two extreme cases, they better illustrate the results that we generated. SIC 3 is included to demonstrate test times for an intermediate case of 3-D stacking, as opposed to simply the opposite extremes.

To determine the test architecture and test length for a given die (SoC) with a given TAM width, we have used the control-aware TAM design method in [36]. Control-aware TAM design takes into account the number of scan enable signals required for independent testing of TAMs in the architecture. For PSHD (3-D SIC with hard dies), the test lengths (cycles) and TAM widths for different dies are listed in Table I. Note that test pins were assigned to dies based on their sizes in order to avoid very large test lengths for any individual die.

The minimal achievable test length for the hard die stack can be seen to be 1 947 063 cycles, which occurs when all dies are tested in parallel with one another. To investigate the effect of achieving this test length for a 3-D stack, we examine SIC 1 and SIC 2. For both SICs, this architecture requires 115 test pins on the bottom die. For SIC 1, this requires 195 test TSVs. For SIC 2, it requires 265 test TSVs.

Table II compares optimal results produced using ILP with those produced using a greedy algorithm for PSHD SIC 1.

TABLE II
COMPARISON OF OPTIMIZATION RESULTS BETWEEN PSHD AND A GREEDY ALGORITHM FOR SIC 1

TSV_{max}	W_{pin}	PSHD (ILP)		PSHD (Greedy)		
		Test Length (cycles)	Test Schedule	Test Length (cycles)	Test Schedule	Percentage Difference in Test Length
160	30	4 748 920	1, 2, 3, 4, 5	4 748 920	1, 2, 3, 4, 5	0.0
160	35	4 652 620	1, 2, 3, 4‖5	4 652 620	1, 2, 3, 4‖5	0.0
160	40	4 652 620	1, 2, 3, 4‖5	4 652 620	1, 2, 3, 4‖5	0.0
160	45	3 983 290	1‖5, 2‖4, 3	4 001 340	1, 2‖5, 3‖4	0.5
160	50	3 428 310	1‖4, 2‖3, 5	3 428 310	1‖4, 2‖3, 5	0.0
160	55	2 712 690	1‖2, 3‖4, 5	2 712 690	1‖2, 3‖4, 5	0.0
160	60	2 616 390	1‖2, 3‖4‖5	2 712 690	1‖2, 3‖4, 5	3.7
160	65	2 616 390	1‖2, 3‖4‖5	2 712 690	1‖2, 3‖4, 5	3.7
160	70	2 616 390	1‖2‖5, 3‖4	2 616 390	1‖2‖5, 3‖4	0.0
160	75	2 598 340	1‖2‖4, 3‖5	2 616 390	1‖2‖5, 3‖4	0.7
160	80	2 598 340	1‖2‖4, 3‖5	2 616 390	1‖2‖5, 3‖4	0.7
160	85	2 598 340	1‖2‖4, 3‖5	2 616 390	1‖2‖5, 3‖4	0.7
160	90	2 598 340	1‖2‖4, 3‖5	2 616 390	1‖2‖5, 3‖4	0.7
160	95	2 043 360	1‖2‖3‖4, 5	2 616 390	1‖2‖5, 3‖4	28.0
160	100	2 043 360	1‖2‖3‖4, 5	2 043 360	1‖2‖3‖4, 5	0.0

TABLE III
COMPARISON OF OPTIMIZATION RESULTS BETWEEN PSHD AND A GREEDY ALGORITHM FOR SIC 1

TSV_{max}	W_{pin}	PSSD (ILP)		PSSD (Greedy)		
		Test Length (cycles)	Test Schedule	Test Length (cycles)	Test Schedule	Percentage Difference in Test Length
140	30	4 795 930	1‖2‖3‖4, 5	7 842 000	1‖2,3,4‖5	63.5
140	35	4 237 100	1‖2‖3‖4, 5	7 633 580	1‖3,2‖4, 5	80.1
140	40	3 841 360	1‖2‖3‖4, 5	6 846 400	1‖3,2‖4, 5	78.2
140	45	3 591 550	1‖2‖3‖4, 5	6 379 510	1‖2‖3, 4‖5	77.6
140	50	3 090 720	1‖2‖3‖4, 5	6 041 270	1‖2‖3, 4‖5	95.5
140	55	2 991 860	1‖2‖3‖4‖5	5 873 430	1‖2‖3‖4,5	96.3
140	60	2 873 290	1‖2‖3‖4, 5	5 821 900	1‖2‖3‖4, 5	102.6
140	65	2 784 050	1‖2‖3‖4‖5	5 705 410	1‖2‖3‖4, 5	104.9
140	70	2 743 320	1‖2‖3‖4‖5	5 638 140	1‖2‖3‖4, 5	105.5
140	75	2 629 500	1‖2‖3‖4‖5	5 638 140	1‖2‖3‖4, 5	114.4
140	80	2 439 380	1‖2‖3‖4‖5	5 496 200	1‖2‖3‖4, 5	125.3
140	85	2 402 330	1‖2‖3‖4‖5	5 447 190	1‖2‖3‖4, 5	126.7
140	90	2 395 760	1‖2‖3‖4‖5	5 447 190	1‖2‖3‖4, 5	127.4
140	95	2 383 400	1‖2‖3‖4‖5	5 447 190	1‖2‖3‖4, 5	128.5
140	100	2 369 680	1‖2‖3‖4‖5	5 351 480	1‖2‖3‖4, 5	125.8

The greedy algorithm attempts to combine dies in parallel tested sets, starting with those that would lead to the greatest reduction in test time. If more than one combination results in the same reduction, it prioritizes those combinations that result in the smallest test resource use. Compared to PSSD and PSFD, PSHD is a less complex problem. Thus, the greedy algorithm is capable of producing results that are sometimes optimal, though it often does not result in the minimum test time.

Table III shows the information in Table II for problem PSSD. The soft die problem is more difficult to solve with a greedy heuristic. The heuristic algorithm for PSSD uses the greedy algorithm from PSHD as a subroutine in test-architecture optimization. It begins with an assignment of equal number of test pins to each of the dies and optimizes the 2-D and 3-D TAM under these constraints. It then randomly adds and removes random numbers of test pins from each die, each time balancing the result to use the maximum number of test pins, and optimizes again. It checks for reductions in test time, returning to the best solution so far if no test time reduction is produced or constraints are violated. It terminates after 10 000 iterations of no improvement. As can be seen, the optimal ILP solution tends to be much better in terms of test length than the heuristic solution, simply because the solution space is so large.

It is useful to discuss briefly how the test architecture for a soft die is built using the information produced from the ILP optimization. Prior to using the ILP model, 2-D TAM architectures that minimize test time are produced for each die in the stack assuming a wide range of TAM widths available to that die. In this sense, the 2-D architecture is already completed and we need to choose which architecture to use. The ILP formulation provides data with regards to which TAM width to use for each die and the test schedule for testing the die, which lets the designer know which die need to be tested in parallel. With this information, the design of the 3-D TAM is elementary. The integrator simply provides the appropriate TAM width to each die, assuring that the number of test elevators between each die is sufficient for the bandwidth needed for any parallel tests.

Take, for example, the information provided for the ILP optimization of soft die in Table III when W_{pin} is 60. Though not shown in the table, the ILP optimization provides the designer with the following widths for each die (these values show the number of test pins used by that die, so the TAM width is half the given value): $W_1 = 30$, $W_2 = 20$, $W_3 = 6$,

TABLE IV
EXPERIMENTAL RESULTS FOR PSHD

TSV_{max}	W_{pin}	SIC 1			SIC 2			SIC 3		
		Test Length (cycles)	Test Schedule	Reduction (%)	Test Length (cycles)	Test Schedule	Reduction (%)	Test Length (cycles)	Test Schedule	Reduction (%)
160	30	4 748 920	1, 2, 3, 4, 5	0.00	4 748 920	1, 2, 3, 4, 5	0.00	4 748 920	1, 2, 3, 4, 5	0.00
160	35	4 652 620	1, 2, 3, 4‖5	2.03	4 652 620	1‖2, 3, 4, 5	2.03	4 652 620	1‖5, 2, 3, 4	2.03
160	40	4 652 620	1, 2, 3, 4‖5	2.03	4 652 620	1‖3, 2, 4, 5	2.03	4 652 620	1‖5, 2, 3, 4	2.03
160	45	3 983 290	1‖5, 2‖4,3	16.12	3 983 290	1‖3, 2‖4, 5	16.12	3 983 290	1‖4, 2‖5, 3	16.12
160	50	3 428 310	1‖4, 2‖3,5	27.81	3 428 310	1, 2‖5, 3‖4	27.81	3 428 310	1, 2‖4, 3‖5	27.81
160	55	2 712 690	1‖2, 3‖4,5	42.88	2 712 690	1, 2‖3, 4‖5	42.88	2 712 690	1, 2‖3, 4‖5	42.88
160	60	2 616 390	1‖2, 3‖4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖4‖5, 2‖3	44.91
160	65	2 616 390	1‖2, 3‖4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖4‖5, 2‖3	44.91
160	70	2 616 390	1‖2‖5, 3‖4	44.91	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	75	2 598 340	1‖2‖4, 3‖5	45.29	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	80	2 598 340	1‖2‖4, 3‖5	45.29	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	85	2 598 340	1‖2‖4, 3‖5	45.29	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	90	2 598 340	1‖2‖4, 3‖5	45.29	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	95	2 598 340	1‖2‖4, 3‖5	45.29	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	100	2 043 360	1‖2‖3‖4, 5	56.97	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91
160	105	2 043 360	1‖2‖3‖4, 5	56.97	2 616 390	1‖2‖3, 4‖5	44.91	2 616 390	1‖2‖3, 4‖5	44.91

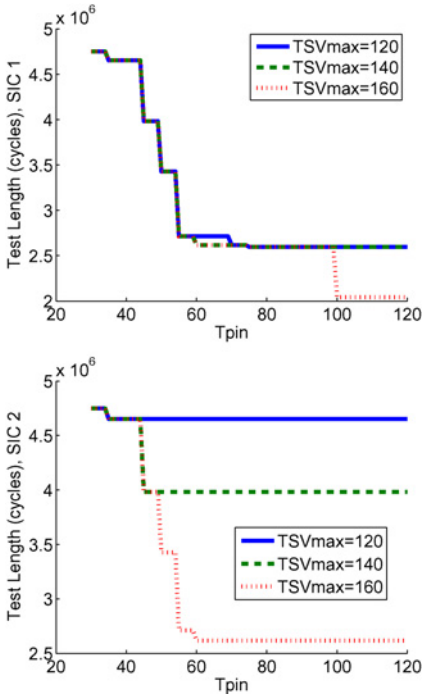

Fig. 9. Test length with respect to TSV_{max} for SIC 1 and SIC 2 with hard dies.

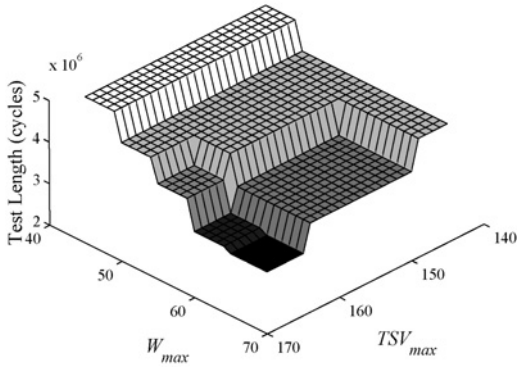

Fig. 10. Variation in test length with W_{max} and TSV_{max} for SIC 2 with hard dies.

$W_4 = 4$, $W_5 = 30$. The designer sees that Dies 1-4 are tested in parallel, followed by Die 5. Since the width of the top die dominates this stack, TSV routing simply requires 30 test elevators between each die. Die 1, 2, 3, and 4 each utilize a different test pin of the 60 test pins available, and Die 5 can utilize any 30 of the test pins. Which pin is routed to which die is up to the best judgment of the designer, as is wire routing and the like.

For a fixed TSV_{max} and range of W_{max}, Table IV presents representative results for PSHD for the three benchmark 3-D SICs using hybrid TestRail architectures [14]. Additional values for TSV_{max} could be considered, but they do not provide any new insights. For PSHD and its comparison to PSHD, optimizations were done using hybrid TestBus [14] architecture for variety. The ILP models were solved using the XPRESS-MP tool [37]. In this table, Column 1 shows the maximum number of TSVs allowed (TSV_{max}), while Column 2 represents the number of available test pins W_{max}. Columns 3, 6, and 9 represent the total test length (cycles) for the stack for 3-D SIC 1, 2, and 3, respectively. Columns 4, 7, and 10 show the resulting test schedule for the 3-D SICs, where the symbol "‖" indicates parallel testing of dies, and a "," represents serial testing. Finally, Columns 5, 8, and 11 show the percent decrease in test length over the serial testing case for the three 3-D SICs. From Table IV we can see that compared to serial testing of all dies (first row in the table), the proposed method obtains up to 57% reduction in test length. Note that although identical test lengths were obtained for SIC 2 and SIC 3 for $TSV_{max} = 160$, different TAM architectures and test schedules are obtained from the optimization algorithm (see Columns 4, 10).

For a different number of TSVs (TSV_{max}), Fig. 9(a) and (b) shows the variation in test length T with an increase in number of test pins W_{max} for SIC 1 and SIC 2. From the figures, we can see that both TSV_{max} and W_{max} determine which dies should be tested in parallel, and thus the total test length for the stack. For a given value of TSV_{max}, increasing W_{max} does not always decrease the test length, showing the presence of Pareto-optimal points. These have an important impact on test

TABLE V
COMPARISONS BETWEEN PSHD AND PSFD

Optimization Problem	W_{max}	Test Length (cycles)	Reduction (%)	Test Schedule	No. of Test Pins Used per Die
PSHD	35	4 678 670	0	1, 2, 3, 4‖5	30, 24, 24, 20, 14
	44	4 009 340	0	1‖4, 2, 3‖5	30, 24, 24, 20, 14
	50	3 381 720	0	1‖3, 2‖5, 4,	30, 24, 24, 20, 14
	60	2 658 750	0	1‖5, 2‖3, 4,	30, 24, 24, 20, 14
	80	2 658 750	0	1‖5, 2‖3, 4,	30, 24, 24, 20, 14
PSFD	35	3 828 490	18.17	1‖4,2‖3, 5	28, 24, 10, 7, 14
	44	2 875 900	28.27	1‖2, 3‖4, 5	28, 16, 24, 18, 14
	50	2 641 060	21.90	1‖2, 3‖4‖5	30, 18, 24, 18, 4
	60	2 335 780	12.15	1‖2‖3‖4, 5	28, 16, 10, 6, 14
	80	1 971 400	25.85	1‖2‖3‖4‖5	30, 24, 10, 8, 8

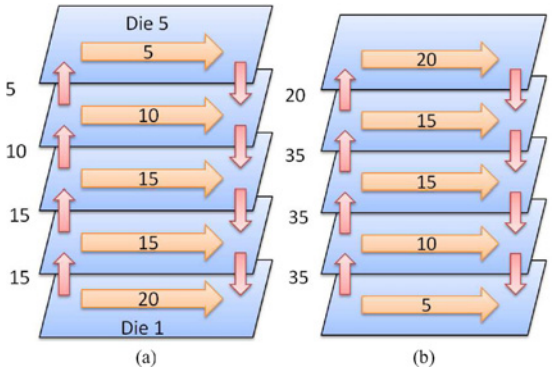

Fig. 11. Example of optimization for (a) SIC 1 versus (b) SIC 2.

resource allocation, since test resources for a target test length should be provided only to the extent that they align with the first point in a Pareto-optimal plateau.

Fig. 10 shows the variation in test length for SIC 2 when both TSV_{max} and W_{max} are varied. From the figure, we can see that a small increase in the number of test pins W_{max} for a given TSV_{max} reduces the test length significantly, while to achieve the same reduction in test length with a fixed number of test pins W_{max}, a large increase in TSV_{max} is required.

Fig. 11 demonstrates the differences between SIC 1 and SIC 2 during optimization. The two 3-D stacks with five dies each are shown, with TAM widths displayed on each die and the number of TSVs used between each die shown to the left of the stack. Fig. 11(a) shows the number of TSVs needed to test Die 1 and Die 2 in parallel followed by Dies 3, 4, and 5 for SIC 1. It is desirable to test Die 1 and Die 2 in parallel since they are the dies with the longest test lengths. This requires 90 TSVs. For SIC 2, this requires 250 TSVs as shown in Fig. 11(b). This demonstrates why optimization produces better results for SIC 1 than for SIC 2.

Table V compares results for PSHD and PSFD for TestBus architectures. We see from Table V that by adding serial/parallel conversion of TAM inputs to hard dies, we can obtain as much as 28% reduction in test length. This is because the conversion allows for an increase in the test length of individual die in order to minimize the overall SIC test length during test schedule optimization. We also note that the test schedules and the number of test pins utilized for each die differ considerably between the hard-die and firm-die problem instances. Compared to a hard die, a firm die requires a small amount of extra hardware to convert a narrow TAM at the die input

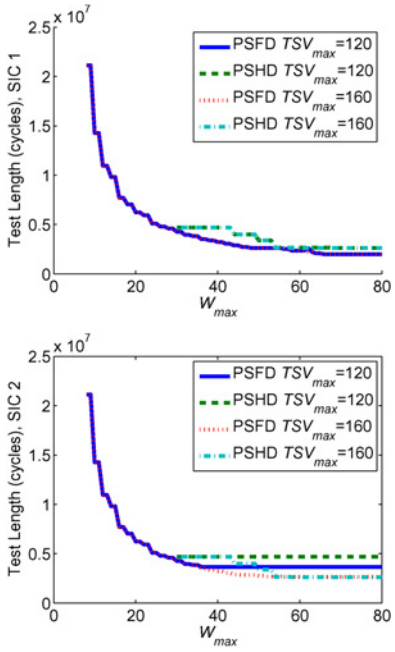

Fig. 12. Comparison of variation in test length with W_{max} for SIC 1 and SIC 2 between firm dies and hard dies.

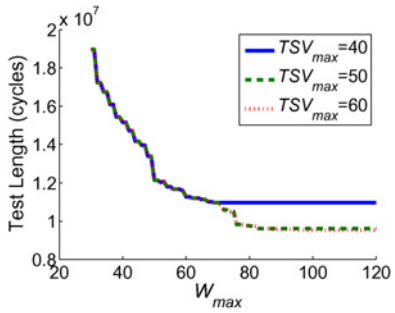

Fig. 13. Variation in test length with W_{max} for SIC 1 with soft dies.

to a wider die-internal TAM, and vice versa. The area cost of this additional hardware is negligible compared to the die area and the hardware required for the core and die wrappers.

Fig. 12 shows comparative test lengths between PSHD and PSFD when W_{max} is varied, for two values of TSV_{max} and for two SICs. It is impossible to test the hard dies in these cases using fewer than 30 test pins without using serial/parallel conversion. As fewer test pins are used, the test lengths for individual die greatly increase, resulting in a sharp increase in overall test length below certain values of W_{max}. It is important to note that the test length for a SIC with hard dies can never be shorter than the test length for the same SIC with firm dies; at best it can be equal. This is because, in the worst case with respect to test length, the optimization for firm dies is equivalent to the optimization for hard dies, i.e., no serial/parallel conversion is carried out. It can be seen that the use of serial/parallel conversion hardware can result in less use of test resources and shorter test time compared to hard die without conversion. This observation is particularly valid in SIC 2, where the position of dies in the stack is more of a limiter to test time reduction.

Fig. 14 shows Pshdt (3-D SIC with hard dies and TSV-count optimization) results for SIC 1. Under tight test length constraints, a solution to the optimization is unattainable for

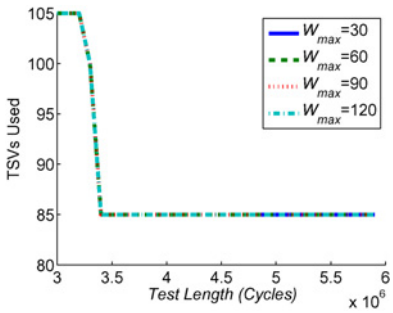

Fig. 14. Variation in TSVs used with $T_{\max}$ for SIC 1 with hard dies.

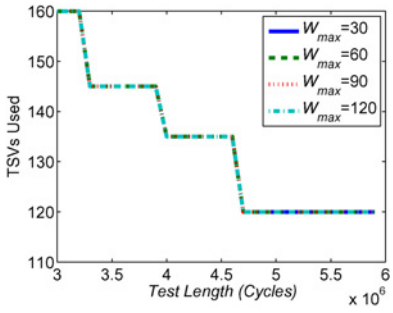

Fig. 15. Variation in TSVs used with $T_{\max}$ for SIC 2 with hard dies.

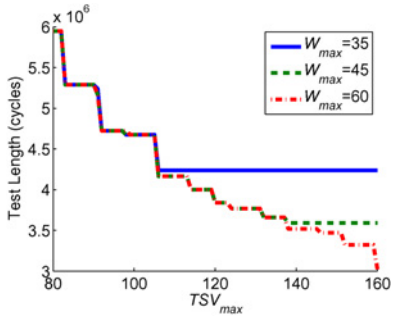

Fig. 16. Variation in test length with $TSV_{\max}$ for SIC 2 with soft dies.

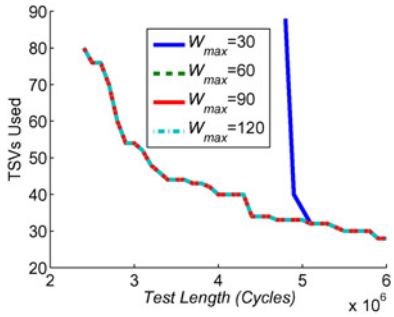

Fig. 17. Variation in TSVs used with $T_{\max}$ for SIC 1 with soft dies.

smaller values of $W_{\max}$. In Fig. 14, for example, a $W_{\max}$ value of 30 will not produce a feasible test architecture until the test length is above 470 000 cycles. Once an optimized architecture is achievable, the minimum number of TSVs used is generally the same regardless of the value of $W_{\max}$. There are two reasons for this. The first is that there are only a few configurations of the 3-D TAM for the hard-die stack, and multiple configurations will give the same minimal TSV value. This is only a partial explanation, however, as equal minimal-TSV values for various $W_{\max}$ values are seen for soft die as well. The primary reason for the results of Fig. 14 is that in order to minimize the number of TSVs used by the stack, the ILP solver will tend toward testing all die in series with each other. If this cannot be done, then it will attempt to test only those die with the smallest TAM width in parallel. This test configuration—tending toward serial testing—also happens to be the configuration that results in the fewest number of test pins used. This is why the TSV-count values in Fig. 14 overlap even for tight test pin constraints—minimizing the TSVs used also tends to minimize the number of test pins used. This is seen for both SIC 1 and SIC 2 (Fig. 15), though the number of TSVs needed for testing in the less optimized SIC 2 stack is higher. Results for Pshdw (3-D SIC with hard dies and test-pin-use optimization) are also as expected—if minimizing TSV use also tends to minimize test-pin use, then minimizing test-pin use should also tend to minimize TSV use. As such, we once again observed overlapping minimal test-pin use for both tight and loose $TSV_{\max}$ constraints. Note that optimizing for minimum TSV or test pin use tends toward serial testing. Therefore, these optimizations result in very different test architectures than optimizing for test time. In contrast, solutions that minimize test time tend to result in parallel testing of many die, as can be seen for PSSD in Table III.

For PSSD (3-D SIC with soft dies), Pareto-optimality is almost non-existent when $W_{\max}$ is varied (see Fig. 13). This is due to the fact that as dies in the stack are soft, it is always possible to find one die for which adding an extra test pin reduces the overall test length. Some Pareto-optimal points can be identified for SIC 2. This is because the most complex dies in a stack tend to be the bottleneck in reducing test length. Since these dies are stacked toward the top of the stack in SIC 2, TSV constraints are more restrictive; the addition of test pins to these dies requires more TSVs and TestElevators throughout the stack. However, for PSSD, although varying $W_{\max}$ does not create Pareto-optimal points, varying $TSV_{\max}$ results in various Pareto-optimal points as shown in Fig. 16. Note that this effect is more pronounced in SIC 2 than in the other 3-D SICs. This is because the addition of test pins to the bottleneck die (at the highest layer) introduces a larger TSV overhead than in the other 3-D SICs. Furthermore, as long as $W_{\max}$ is sufficient, $TSV_{\max}$ is the limiter on test length. For PSHD, PSSD, and PSFD, the stack configuration (SIC 1) with the largest die at the lowest layer and the smallest die at the highest layer is the best for reducing test length while using the minimum number of TSVs. Fig. 19 shows a comparison of optimized test lengths for soft die between SIC 2 and SIC 3. As shown, SIC 3 leads to test lengths lower than or equivalent to SIC 2 at higher values for $W_{\max}$. However, under tight test pin constraints SIC 2 results in better test lengths.

Fig. 17 shows Pssdt (3-D SIC with soft dies and TSV-count optimization) results for SIC 1. Compared to the optimizations for hard dies, we see less Pareto-optimality as expected, since more leeway exists in the soft model. For the reasons described earlier, we still see similar results as in Figs. 1–14. Similar observations are made for test-pin optimization, as seen in Fig. 18, which shows Pssdw (3-D SIC with soft dies and test-pin-use optimization) results for SIC 2.

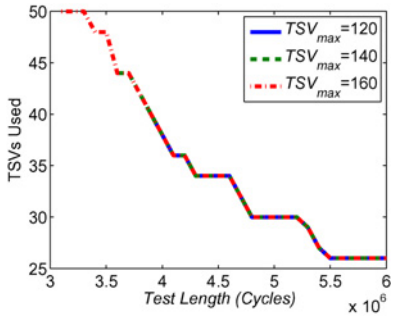

Fig. 18. Variation in test pins used with $T_{\max}$ for SIC 2 with soft dies.

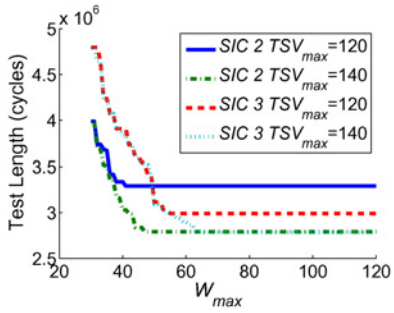

Fig. 19. Comparison of test lengths for PSSD for SIC 2 and SIC 3.

We next study the optimization of SIC 2 versus SIC 3 for problem PSSD (see Fig. 19). For hard die, as shown in Table IV, similar test times with different architectures were produced. This is because the hard die model is too limited with 3-D constraints to lead to different test time for SIC 2 and SIC 3. This is not the case for soft die, where the additional degree of freedom leads to different architectures and better test times in SIC 3 when compared to SIC 2, as expected. From a design perspective, this means that a stack layout with lower test time can be achieved if one can generally keep the most complex die in lower layers in the stack.

Finally, we examine the scalability of the ILP-based optimization method by determining the stack size (number of layers) for which the ILP solver takes more than one day of CPU time. For PSHD, we obtain $M = 16$, while for PSSD, we see that $M = 10$. Since these values of M are unlikely for realistic stacks, we conclude that the proposed method is scalable and practical.

VI. Conclusion

We introduced the problem of test-architecture optimization for 3-D stacked ICs with hard, soft, and firm dies. In the case of hard dies, the test architecture for each die is fixed and given, for firm dies serial/parallel conversion is allowed, while for soft dies the test architecture has to be determined while designing the test architecture for the entire stack. We used ILP techniques to solve the above problems. We considered constraints on the number of available test pins and the number of TSVs used. Results for three different stack configuration made up of five SoCs taken from the ITC'02 SoC Test Benchmarks show that Pareto-optimal points are present for stacks with hard dies, while for stacks with soft dies and firm dies, test length is, in most cases, monotonically reduced with an increase in the number of test pins. Moreover, increasing the number of test pins typically provides a greater reduction in test length compared to an increase in the number of TSVs. Finally, stacks with large and complex dies at the lowest layers lead to lower test lengths compared to stacks with complex dies at the highest layers.

Appendix

We present proofs of correctness of (4) and (9).

A. Inductive Proof Correctness for (4)

1) *Base Case:* For the base case, we consider two layers, and there are two possible outcomes. Either both die are tested in series, or both die are tested in parallel. In the case of series testing, then $y_1 = 1, x_{11} = 1, x_{12} = 0$ and $y_2 = 1, x_{21} = 0$, $x_{22} = 1$. Using this information and (4), we determine the test length as $y_1 \max\{x_{11} \cdot t_1\} + y_2 \max\{x_{22} \cdot t_2\} = \max\{t_1, 0\} + \max\{t_2\} = t_1 + t_2$. For parallel testing, the variables become $y_1 = 1, x_{11} = 1, x_{12} = 1$ and $y_2 = 0, x_{21} = 1, x_{22} = 1$. The equation becomes $1 \cdot \max\{t_1, t_2\} + 0 \cdot \max\{t_2\} = \max\{t_1, t_2\}$. These can both be seen to be correct.

2) *Induction Hypothesis:* We assume that (4) holds for M die.

3) *Recursive Step:* We wish to prove that the test length for die $M+1$ is properly considered in the overall test length. Either die $M+1$ is tested in serial with regards to the die in the stack, or it is tested in parallel with some die on a lower layer of the stack. When die $M+1$ is tested in series, $y_{M+1} = 1, x_{M+1,M+1} = 1$, and $x_{n,M+1}$ and $x_{M+1,n}$ are zero for all $n \neq M+1$. The test length becomes

$$\begin{aligned} &y_1 \cdot \max\{x_{11}t_1, x_{12}t_2, \cdots, x_{1M}t_M, x_{1,M+1}t_{M+1}\} + \\ &y_2 \cdot \max\{x_{22}t_2, x_{23}t_3, \cdots, x_{2M}t_M, x_{2,M+1}t_{M+1}\} + \\ &\qquad \cdots + y_M \max\{x_{MM}t_M\}. \end{aligned} \tag{16}$$

In this equation, $x_{n,M+1}$ is 0 for all $n \neq M+1$, so the test length of die $M+1$ is only added to the total test length once, for

$$y_{M+1} \max\{x_{M+1,M+1}t_{M+1} = t_{M+1}\}. \tag{17}$$

For the parallel case, die $M+1$ is tested in parallel with one or more die below it. Let die k be the die lowest in layer that is tested in parallel with die $M+1$. Then $y_k = 1, x_{k,M+1} = 1, y_{M+1} = 0$, and $x_{M+1,k} = 1$. Then

$$y_{M+1} \max\{x_{M+1,M+1}t_{M+1}\} \tag{18}$$

goes to zero, and

$$y_k \max\{x_{kk}t_k, x_{k,k+1}t_{k+1}, ..., x_{k,M+1}t_{M+1}\} \tag{19}$$

takes into account the test length of dies k, $M+1$, and any other die tested in this parallel tested set.

B. Inductive Proof of Correctness for (9)

1) *Base Case:* We consider for the base case a stack of 2 dies. The variable y_1 is always equal to one, $x_{11} = 1$, and $x_{22} = 1$. There are two possible configurations for testing the two die. The first configuration is that both die are tested

serially. If this is true, then the variables take the values of $x_{12} = 0$ and $x_{21} = 0$. The equation for y_2 then becomes

$$y_2 \geq \frac{1}{1-2}(x_{21} - 1) - M \tag{20}$$

$$y_2 \geq 1 - M. \tag{21}$$

Since $M < 1$, $1 - M$ is some small fraction greater than zero. Thus, y_2 must be greater than zero, and since it is binary it must take the value of one. The second possibility is that both die are tested in parallel, such that $x_{12} = 1$ and $x_{21} = 1$. This defines y_2 as follows:

$$y_2 \geq \frac{1}{1-2}(x_{21} - 1) - M \tag{22}$$

$$y_2 \geq -M. \tag{23}$$

This leaves y_2 unrestricted, as it can only take the value zero or one and is thus always greater than a negative number. Due to the objective, y_2 will become zero as we desire, since it is tested in parallel with a die lower in the stack than itself.

2) *Induction Hypothesis:* We assume (9) holds for the case of m die.

3) *Case of $m + 1$ Die:* For die $m + 1$, (9) becomes

$$y_{m+1} \geq \frac{1}{1-(m+1)}([x_{(m+1)1} - 1] + [x_{(m+1)2} - 1] + \ldots + [x_{(m+1)m} - 1]) - M. \tag{24}$$

If $m + 1$ is tested serially, then the summation adds the quantity -1 a total of m times. This results in

$$y_m \geq \frac{1}{-m}(-m) - M \tag{25}$$

$$y_m \geq 1 - M. \tag{26}$$

This forces y_{m+1} to one. We now consider the range of values that can be taken by the right-hand side of (24) for parallel test cases of die $m+1$ by considering the extremes. If die $m+1$ is tested in parallel with only one die below it, then one of the terms of the summation becomes zero and we get

$$y_m \geq \frac{1}{-m}(-(m-1)) - M \tag{27}$$

$$y_m \geq \frac{1-m}{-m} - M. \tag{28}$$

The fraction clearly results in a positive number less than one, and subtracting M makes this a negative value, leaving y_{m+1} unrestricted. In the case of die $m + 1$ being tested in parallel with every die below it, then every term of the summation is zero and we get

$$y_m \geq \frac{1}{-m}(0) - M \tag{29}$$

$$y_m \geq -M. \tag{30}$$

Thus, for all cases of die $m + 1$ tested in parallel, the right-hand side is in the range $[\frac{1-m}{-m} - M, -M]$, which are all negative values, thereby leaving y_{m+1} unrestricted.

REFERENCES

[1] Y. Xie, G. H. Loh, and K. Bernstein, "Design space exploration for 3-D architectures," *J. Emerg. Technol. Comput. Syst.*, vol. 2, no. 2, pp. 65–103, 2006.

[2] R. Weerasekera, L.-R. Zheng, D. Pamunuwa, and H. Tenhunen, "Extending systems-on-chip to the third dimension: Performance, cost and technological tradeoffs," in *Proc. Int. Conf. Comput.-Aided Des.*, Nov. 2007, pp. 212–219.

[3] G. Loh, Y. Xie, and B. Black, "Processor design in 3-D die stacking technologies," *IEEE Micro*, vol. 27, no. 3, pp. 31–48, May–Jun. 2007.

[4] W. R. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. M. Sule, M. Steer, and P. D. Franzon, "Demystifying 3-D ICs: The pros and cons of going vertical," *IEEE Des. Test Comput.*, vol. 22, no. 6, pp. 498–510, Nov.–Dec. 2005.

[5] U. Kang, H.-J. Chung, S. Heo, S.-H. Ahn, H. Lee, S.-H. Cha, J. Ahn, D.-M. Kwon, J. H. Kim, J.-W. Lee, H.-S. Joo, W.-S. Kim, H.-K. Kim, E.-M. Lee, S.-R. Kim, K.-H. Ma, D.-H. Jang, N.-S. Kim, M.-S. Choi, S.-J. Oh, J.-B. Lee, T.-K. Jung, J.-H. Yoo, and C. Kim, "8Gb 3-D DDR3 DRAM using through-silicon-via technology," in *Proc. ISSCC*, Feb. 2009, pp. 130–132.

[6] C. C. Liu, I. Ganusov, M. Burtscher, and S. Tiwari, "Bridging the processor-memory performance gap with 3-D IC technology," *IEEE Des. Test Comput.*, vol. 22, no. 6, pp. 556–564, Nov.–Dec. 2005.

[7] B. Black, D. W. Nelson, C. Webb, and N. Samra, "3-D processing technology and its impact on iA32 microprocessors," in *Proc. ICCD*, Oct. 2004, pp. 316–318.

[8] S. Das, A. Chandrakasan, and R. Reif, "Design tools for 3-D integrated circuits," in *Proc. IEEE ASP-DAC*, Jan. 2003, pp. 53–56.

[9] H.-H. S. Lee and K. Chakrabarty, "Test challenges for 3-D integrated circuits," *IEEE Des. Test Comput.*, vol. 26, no. 5, pp. 26–35, Sep.–Oct. 2009.

[10] E. J. Marinissen and Y. Zorian, "Testing 3-D chips containing through-silicon vias," in *Proc. Int. Test Conf.*, Nov. 2009, pp. 1–11.

[11] E. J. Marinissen, V. Iyengar, and K. Chakrabarty, "A set of benchmarks for modular testing of SoCs," in *Proc. Int. Test Conf.*, Oct. 2002, pp. 519–528.

[12] V. Iyengar, K. Chakrabarty, and E. J. Marinissen, "Test wrapper and test access mechanism co-optimization for system-on-chip," *J. Electron. Testing Theory Applicat.*, vol. 18, pp. 213–230, Apr. 2002.

[13] E. J. Marinissen, S. K. Goel, and M. Lousberg, "Wrapper design for embedded core test," in *Proc. Int. Test Conf.*, 2000, pp. 911–920.

[14] S. K. Goel and E. J. Marinissen, "SoC test architecture design for efficient utilization of test bandwidth," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 8, no. 4, pp. 399–429, 2003.

[15] E. Larsson, K. Arvidsson, H. Fujiwara, and Z. Peng, "Efficient test solutions for core-based designs," *IEEE Trans. Comput.-Aided Des.*, vol. 23, no. 5, pp. 758–775, May 2004.

[16] Y. Huang, S. M. Reddy, W.-T. Cheng, P. Reuter, N. Mukherjee, C.-C. Tsai, O. Samman, and Y. Zaidan, "Optimal core wrapper width selection and SoC test scheduling based on 3-D bin packing algorithm," in *Proc. Int. Test Conf. Comput.*, 2002, pp. 74–82.

[17] Q. Xu and N. Nicolici, "Resource-constrained system-on-a-chip test: A survey," *IEE Proc.: Comput. Digital Tech.*, vol. 152, no. 1, pp. 67–81, Jan. 2005.

[18] B. Noia, K. Chakrabarty, and Y. Xie, "Test-wrapper optimization for embedded cores in TSV-based 3-D SoCs," in *Proc. IEEE Int. Conf. Comput. Des.*, Oct. 2009, pp. 70–77.

[19] B. Noia, K. Chakrabarty, and E. J. Marinissen, "Optimization methods for post-bond die-internal/external testing in 3-D stacked ICs," in *Proc. IEEE Int. Test Conf.*, Nov. 2010, pp. 1–9.

[20] X. Wu, Y. Chen, K. Chakrabarty, and Y. Xie, "Test-access mechanism optimization for core-based 3-D SoCs," *Microelectron. J.*, vol. 41, no. 10, pp. 601–615, Oct. 2010.

[21] L. Jiang, L. Huang, and Q. Xu, "Test architecture design and optimization for 3-D SoCs," in *Proc. Des. Autom. Test Eur.*, 2009, pp. 220–225.

[22] P.-W. Chen, C.-W. Wu, and D.-M. Kwai, "On-chip TSV testing for 3-D IC before bonding using sense amplification," in *Proc. Asian Test Symp.*, 2009, pp. 450–455.

[23] C.-Y. Lo, Y.-T. Hsing, L.-M. Denq, and C.-W. Wu, "SoC test architecture and method for 3-D ICs," *IEEE Trans. Comput.-Aided Des.*, vol. 29, no. 10, pp. 1645–1649, Oct. 2010.

[24] P.-Y. Chen, W. C.-W. Wu, and D.-M. Kwai, "On-chip testing of blind and open-sleeve TSVs for 3-D IC before bonding," in *Proc. IEEE VLSI Test Symp.*, Apr. 2010, pp. 263–268.

[25] L. Jiang, Q. Xu, K. Chakrabarty, and T. M. Mak, "Layout-driven test-architecture design and optimization for 3-D SoCs under pre-bond test-

pin-count constraint," in *Proc. IEEE Int. Conf. Comput.-Aided Des.*, Nov. 2009, pp. 191–196.
[26] E. J. Marinissen, J. Verbree, and M. Konijnenburg, "A structured and scalable test access architecture for TSV-based 3-D stacked ICs," in *Proc. IEEE VLSI Test Symp.*, Apr. 2010, pp. 269–274.
[27] S. Panth and S. K. Lim, "Scan chain and power delivery network synthesis for pre-bond test of 3-D ICs," in *Proc. IEEE VLSI Test Symp.*, May 2011, pp. 26–31.
[28] D. Lewi and H.-H. Lee, "A scan-island based design enabling prebond testability in die-stacked microprocessors," in *Proc. Int. Test Conf.*, 2007, pp. 1–8.
[29] X. Zhao, D. Lewis, H.-H. Lee, and S. K. Lim, "Pre-bond testable low-power clock tree design for 3-D stacked ICs," in *Proc. Int. Conf. Comput.-Aided Des.*, 2009, pp. 184–190.
[30] L. Jiang, Y. Liu, L. Duan, Y. Xie, and Q. Xu, "Modeling TSV open defects in 3-D stacked DRAM," in *Proc. IEEE Int. Test Conf.*, Nov. 2010, pp. 1–9.
[31] *Elpida Completes Development of Cu-TSV (Through Silicon Via) Multi-Layer 8-Gigabit DRAM* [Online]. Available: http://pda.physorg.com/_news170952515.html
[32] K. Chakrabarty, "Optimal test access architectures for system-on-a-chip," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 6, pp. 26–49, Jan. 2001.
[33] M. L. Flottes, J. Pouget, and B. Rouzeyre, "Sessionless test scheme: Power-constrained test scheduling for system-on-a-chip," in *Proc. 11th IFIP VLSI-SoC*, 2001, pp. 105–110.
[34] E. G. Coffman, Jr., M. R. Garey, D. S. Johnson, and R. E. Tarjan, "Performance bounds for level-oriented 2-D packing algorithms," *SIAM J. Comput.*, vol. 9, no. 4, pp. 809–826, 1980.
[35] X. Dong and Y. Xie, "System-level cost analysis and design exploration for 3-D ICs," in *Proc. ASP-DAC*, Jan. 2009, pp. 234–241.
[36] S. K. Goel and E. J. Marinissen, "Control-aware test architecture design for modular SoC testing," in *Proc. Eur. Test Workshop*, 2003, pp. 57–62.
[37] FICO. *Xpress-MP* [Online]. Available: http://www.fico.com/en/Products/DMTools/Pages/FICO-Xpress-Optimization-Suite.aspx

Brandon Noia received the B.S.E. degree from Duke University, Durham, NC. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Duke University, with a Graduate Fellowship from Semiconductor Research Corporation, Research Triangle Park, NC.

Krishnendu Chakrabarty (M'95–F'08) received the B.Tech. degree from the Indian Institute of Technology, Kharagpur, India, in 1990, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, in 1992 and 1995, respectively.

He is currently a Professor of electrical and computer engineering with the Department of Electrical and Computer Engineering, Duke University, Durham, NC. He is also a Chair Professor in software theory with Tsinghua University, Beijing, China. His current research interests include testing and design-for-testability of integrated circuits, digital microfluidics, biochips, and cyberphysical systems, optimization of digital print, and production system infrastructure. He has authored 10 books on these topics (with two more books in press), published over 370 papers in journals and refereed conference proceedings, and given over 150 invited, keynote, and plenary talks.

Dr. Chakrabarty was a recipient of the National Science Foundation Early Faculty (CAREER) Award, the Office of Naval Research Young Investigator Award, the Humboldt Research Fellowship from the Alexander von Humboldt Foundation, Germany, and several best paper awards at IEEE conferences. He is a Golden Core Member of the IEEE Computer Society and a Distinguished Engineer of ACM. He was a 2009 Invitational Fellow of the Japan Society for the Promotion of Science. He was the recipient of the 2008 Duke University Graduate School Dean's Award for Excellence in Mentoring, and the 2010 Capers and Marion McDonald Award for Excellence in Mentoring and Advising, Pratt School of Engineering, Duke University. He served as a Distinguished Visitor of the IEEE Computer Society from 2005 to 2007, and as a Distinguished Lecturer of the IEEE Circuits and Systems Society from 2006 to 2007. Currently, he is an ACM Distinguished Speaker, and a Distinguished Visitor of the IEEE Computer Society from 2010 to 2012. He is the Editor-in-Chief of the IEEE DESIGN AND TEST OF COMPUTERS and the *ACM Journal on Emerging Technologies in Computing Systems*. He is an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, and the IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS. He is an Editor of the *Journal of Electronic Testing: Theory and Applications*.

Sandeep Kumar Goel (SM'05) received the B.Tech. degree from the Institute of Engineering and Technology, Lucknow, India, in 1998, the M.Tech. degree from the Indian Institute of Technology Delhi, New Delhi, India, in 1999, and the Ph.D. degree from the University of Twente, Enschede, The Netherlands, in 2005.

He is currently a DFT/Diagnosis Manager with TSMC Technology, Inc., San Jose, CA. Prior to TSMC, he was with LSI Corporation, Milpitas, CA, Magma Design Automation, San Jose, and Philips Research, Eindhoven, The Netherlands. He has co-authored two books, contributed three invited chapters to book volumes, published over 60 papers in journals and referred conference/workshop proceedings, and delivered several invited talks. He is a co-inventor on seven granted U.S. and European patents and has 12 other patents pending. His current research interests include all topics in the domain of testing and debugging digital very large scale integration 2-D/3-D systems.

Dr. Goel was a recipient of the Most Significant Paper Award at the IEEE International Test Conference in 2010. He serves on various conference and workshop committees, including Design, Automation and Test in Europe, the IEEE European Test Symposium, the IEEE Infrastructure IP Workshop, the IEEE Workshop on Design and Diagnostics of Electronic Circuits and Systems, and the IEEE International Workshop on Defect and Data Driven Testing.

Erik Jan Marinissen (F'11) received the M.Sc. degree in computing science and the PDEng. degree in software technology from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 1990 and 1992, respectively.

He is currently a Principal Scientist with the 3-D Integration Program, IMEC, Leuven, Belgium. Prior to IMEC, he was with NXP Semiconductors and Philips Research, both in Eindhoven. He is a co-author of more than 150 journal and conference papers and a co-inventor on nine granted U.S. and EU patent families. His current research interests include all topics in the domain of test and debug of integrated circuits.

Dr. Marinissen was a recipient of best paper awards at the Chrysler-Delco-Ford Automotive Electronics Reliability Workshop in 1995, the IEEE International Board Test Workshop in 2002, and the most significant paper awards at the IEEE International Test Conference in 2008 and 2010. He served as an Editor-in-Chief of IEEE Std 1500 and serves as the Working Group Chair of IEEE P1838. He serves on numerous conference committees, including the IEEE Asian Test Symposium, Design, Automation and Test in Europe, the IEEE European Test Symposium, the IEEE International Test Conference, and the IEEE VLSI Test Symposium. He is a founder of workshops on "diagnostic services in network-on-chips," "3-D integration," and "testing 3-D stacked integrated circuits." He serves on the editorial boards of the IEEE DESIGN AND TEST OF COMPUTERS, *IET Computers and Digital Techniques*, and *Springer's Journal of Electronic Testing: Theory and Applications*. He is a Golden Core Member of the Computer Society.

Jouke Verbree received the Bachelors degree in computer science and the Masters degree in computer engineering from the Delft University of Technology, Delft, The Netherlands. He is currently pursuing the Ph.D. degree in computer engineering from the Delft University of Technology. His Masters graduation project was performed at IMEC, Leuven, Belgium.

His current research interests include computer chip design, 3-D SICs, and chip testing. He is a co-author of four conference papers.